AAA Application Traffic
Contents

AAA Application Traffic...................................................................................... 4
  Authentication Authorization Auditing (AAA) ............................................... 5
  How AAA Works .......................................................................................... 9
  Enabling AAA ............................................................................................ 13
  Setting up AAA Virtual Servers and DNS ................................................... 15
    Configuring the Authentication Virtual Server ............................................ 16
    Configuring a Traffic Management Virtual Server ...................................... 20
    Configuring DNS ...................................................................................... 23
    Verifying Your Setup for AAA ................................................................. 24
  Configuring Users and Groups ..................................................................... 26
  Configuring AAA Policies .......................................................................... 31
    Authentication Policies .............................................................................. 32
    Authorization Policies ............................................................................... 39
    Auditing Policies ...................................................................................... 44
  Session Settings ......................................................................................... 51
    Session Profiles ....................................................................................... 52
    Session Policies ........................................................................................ 55
    Global Session Settings ............................................................................ 60
  Traffic Settings ............................................................................................ 63
    Form SSO Profiles .................................................................................... 64
    Traffic Profiles .......................................................................................... 67
    Traffic Policies .......................................................................................... 70
  Authenticating with Client Certificates ....................................................... 74
  Configuring AAA with Commonly Used Protocols ....................................... 77
    Handling Authentication, Authorization and Auditing with Kerberos/NTLM .... 78
      How NetScaler Implements Kerberos Authentication .............................. 80
      Kerberos Authentication - Configuration on the NetScaler Appliance ..... 83
        Enabling AAA on the NetScaler ............................................................ 84
        Adding a Keytab file ............................................................................ 86
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adding a DNS Server</td>
<td>88</td>
</tr>
<tr>
<td>Creating an Authentication Negotiation Policy</td>
<td>89</td>
</tr>
<tr>
<td>Creating an Authentication Virtual Server</td>
<td>91</td>
</tr>
<tr>
<td>Configuring a Traffic Management Virtual Server</td>
<td>93</td>
</tr>
<tr>
<td>Verifying the configuration for Kerberos Authentication</td>
<td>96</td>
</tr>
<tr>
<td>Configuration of Kerberos Authentication on a Client</td>
<td>97</td>
</tr>
<tr>
<td>Offloading Kerberos Authentication from Physical Servers</td>
<td>98</td>
</tr>
</tbody>
</table>
AAA Application Traffic

Many companies restrict web site access to valid users only, and control the level of access permitted to each user. The AAA feature allows a site administrator to manage access controls with the NetScaler appliance instead of managing these controls separately for each application. Doing authentication on the appliance also permits sharing this information across all web sites within the same domain that are protected by the appliance.

The AAA feature supports authentication, authorization, and auditing for all application traffic. To use AAA, you must configure authentication virtual servers to handle the authentication process and traffic management virtual servers to handle the traffic to web applications that require authentication. You also configure your DNS to assign FQDNs to each virtual server. After configuring the virtual servers, you configure a user account for each user that will authenticate via the NetScaler appliance, and optionally you create groups and assign user accounts to groups. After creating user accounts and groups, you configure policies that tell the appliance how to authenticate users, which resources to allow users to access, and how to log user sessions. To put the policies into effect, you bind each policy globally, to a specific virtual server, or to the appropriate user accounts or groups. After configuring your policies, you customize user sessions by configuring session settings and binding your session policies to the traffic management virtual server. Finally, if your intranet uses client certs, you set up the client certificate configuration.

Before configuring AAA, you should be familiar with and understand how to configure load balancing, content switching, and SSL on the NetScaler appliance. For more information about load balancing, content switching, and SSL, see the Citrix NetScaler Traffic Management Guide at http://support.citrix.com/article/CTX128670.
AAA provides security for a distributed Internet environment by allowing any client with the proper credentials to connect securely to protected application servers from anywhere on the Internet. This feature incorporates the three security features of authentication, authorization, and auditing. Authentication enables the NetScaler ADC to verify the client’s credentials, either locally or with a third-party authentication server, and allow only approved users to access protected servers. Authorization enables the ADC to verify which content on a protected server it should allow each user to access. Auditing enables the ADC to keep a record of each user’s activity on a protected server.

To understand how AAA works in a distributed environment, consider an organization with an intranet that its employees access in the office, at home, and when traveling. The content on the intranet is confidential and requires secure access. Any user who wants to access the intranet must have a valid user name and password. To meet these requirements, the ADC does the following:

- Redirects the user to the login page if the user accesses the intranet without having logged in.
- Collects the user’s credentials, delivers them to the authentication server, and caches them in a directory that is accessible through LDAP.
- Verifies that the user is authorized to access specific intranet content before delivering the user’s request to the application server.
- Maintains a session timeout after which users must authenticate again to regain access to the intranet. (You can configure the timeout.)
- Logs the user accesses, including invalid login attempts, in an audit log.

Authentication requires that several entities—the client, the NetScaler appliance, the external authentication server if one is used, and the application server—respond to each other when prompted by performing a complex series of tasks in the correct order. If you are using an external authentication server, this process can be broken down into the following fifteen steps.

- The client sends a GET request for a URL on the application server.
- The NetScaler appliance’s traffic management virtual server redirects the request to the application server.
- The application server determines that the client has not been authenticated, and therefore sends an HTTP 200 OK response via the TM vserver to the client. The response contains a hidden script that causes the client to issue a POST request for /cgi/tm.
- The client sends a POST request for /cgi/tm.
- The NetScaler appliance’s authentication virtual server redirects the request to the authentication server.
The authentication server creates an authentication session, sets and caches a cookie that consists of the initial URL and the domain of the traffic management virtual server, and then sends an HTTP 302 response via the authentication virtual server, redirecting the client to /vpn/index.html.

- The client sends a GET request for /vpn/index.html.

- The authentication virtual server redirects the client to the authentication server login page.

- The client sends a GET request for the login page, enters credentials, and then sends a POST request with the credentials back to the login page.

- The authentication virtual server redirects the POST request to the authentication server.

- If the credentials are correct, the authentication server tells the authentication virtual server to log the client in and redirect the client to the URL that was in the initial GET request.

- The authentication virtual server logs the client in and sends an HTTP 302 response that redirects the client to the initially requested URL.

- The client sends a GET request for their initial URL.

- The traffic management virtual server redirects the GET request to the application server.

- The application server responds via the traffic management virtual server with the initial URL.

If you use local authentication, the process is similar, but the authentication virtual server handles all authentication tasks instead of forwarding connections to an external authentication server. The following figure illustrates the authentication process.
When an authenticated client requests a resource, the ADC, before sending the request to the application server, checks the user and group policies associated with the client account, to verify that the client is authorized to access that resource. The ADC handles all authorization on protected application servers. You do not need to do any special configuration of your protected application servers.

AAA-TM handles password changes for users by using the protocol-specific method for the authentication server. For most protocols, neither the user nor the administrator needs to do anything different than they would without AAA-TM. Even when an LDAP authentication server is in use, and that server is part of a distributed network of LDAP servers with a single designated domain administration server, password changes are usually handled seamlessly. When an authenticated client of an LDAP server changes his or her password, the client sends a credential modify request to AAA-TM, which forwards it to the LDAP server. If the user's LDAP server is also the domain administration server, that server responds appropriately and AAA-TM then performs the requested password change. Otherwise, the LDAP server sends AAA-TM an LDAP_REFERRAL response to the domain administration server. AAA-TM follows the referral to the indicated domain administration server, authenticates to that server, and performs the password change on that server.

When configuring AAA-TM with an LDAP authentication server, the system administrator must keep the following conditions and limitations in mind:

- AAA-TM assumes that the domain administration server in the referral accepts the same bind credentials as the original server.

- AAA-TM only follows LDAP referrals for password change operations. In other cases AAA-TM refuses to follow the referral.

- AAA-TM only follows one level of LDAP referrals. If the second LDAP server also returns a referral, AAA-TM refuses to follow the second referral.
The ADC supports auditing of all states and status information, so you can see the details of what each user did while logged on, in chronological order. To provide this information, the appliance logs each event, as it occurs, either to a designated audit log file on the appliance or to a syslog server. Auditing requires configuring the appliance and any syslog server that you use.
AAA provides security for a distributed Internet environment by allowing any client with the proper credentials to connect securely to protected application servers from anywhere on the Internet. This feature incorporates the three security features of authentication, authorization, and auditing. Authentication enables the NetScaler ADC to verify the client’s credentials, either locally or with a third-party authentication server, and allow only approved users to access protected servers. Authorization enables the ADC to verify which content on a protected server it should allow each user to access. Auditing enables the ADC to keep a record of each user’s activity on a protected server.

To understand how AAA works in a distributed environment, consider an organization with an intranet that its employees access in the office, at home, and when traveling. The content on the intranet is confidential and requires secure access. Any user who wants to access the intranet must have a valid user name and password. To meet these requirements, the ADC does the following:

- Redirects the user to the login page if the user accesses the intranet without having logged in.
- Collects the user’s credentials, delivers them to the authentication server, and caches them in a directory that is accessible through LDAP.
- Verifies that the user is authorized to access specific intranet content before delivering the user’s request to the application server.
- Maintains a session timeout after which users must authenticate again to regain access to the intranet. (You can configure the timeout.)
- Logs the user accesses, including invalid login attempts, in an audit log.

Authentication requires that several entities—the client, the NetScaler appliance, the external authentication server if one is used, and the application server—respond to each other when prompted by performing a complex series of tasks in the correct order. If you are using an external authentication server, this process can be broken down into the following fifteen steps.

- The client sends a GET request for a URL on the application server.
- The NetScaler appliance’s traffic management virtual server redirects the request to the application server.
- The application server determines that the client has not been authenticated, and therefore sends an HTTP 200 OK response via the TM vserver to the client. The response contains a hidden script that causes the client to issue a POST request for /cgi/tm.
- The client sends a POST request for /cgi/tm.
- The NetScaler appliance’s authentication virtual server redirects the request to the authentication server.
How AAA Works

- The authentication server creates an authentication session, sets and caches a cookie that consists of the initial URL and the domain of the traffic management virtual server, and then sends an HTTP 302 response via the authentication virtual server, redirecting the client to /vpn/index.html.

- The client sends a GET request for /vpn/index.html.

- The authentication virtual server redirects the client to the authentication server login page.

- The client sends a GET request for the login page, enters credentials, and then sends a POST request with the credentials back to the login page.

- The authentication virtual server redirects the POST request to the authentication server.

- If the credentials are correct, the authentication server tells the authentication virtual server to log the client in and redirect the client to the URL that was in the initial GET request.

- The authentication virtual server logs the client in and sends an HTTP 302 response that redirects the client to the initially requested URL.

- The client sends a GET request for their initial URL.

- The traffic management virtual server redirects the GET request to the application server.

- The application server responds via the traffic management virtual server with the initial URL.

If you use local authentication, the process is similar, but the authentication virtual server handles all authentication tasks instead of forwarding connections to an external authentication server. The following figure illustrates the authentication process.
When an authenticated client requests a resource, the ADC, before sending the request to the application server, checks the user and group policies associated with the client account, to verify that the client is authorized to access that resource. The ADC handles all authorization on protected application servers. You do not need to do any special configuration of your protected application servers.

AAA-TM handles password changes for users by using the protocol-specific method for the authentication server. For most protocols, neither the user nor the administrator needs to do anything different than they would without AAA-TM. Even when an LDAP authentication server is in use, and that server is part of a distributed network of LDAP servers with a single designated domain administration server, password changes are usually handled seamlessly. When an authenticated client of an LDAP server changes his or her password, the client sends a credential modify request to AAA-TM, which forwards it to the LDAP server. If the user’s LDAP server is also the domain administration server, that server responds appropriately and AAA-TM then performs the requested password change. Otherwise, the LDAP server sends AAA-TM an LDAP_REFERRAL response to the domain administration server. AAA-TM follows the referral to the indicated domain administration server, authenticates to that server, and performs the password change on that server.

When configuring AAA-TM with an LDAP authentication server, the system administrator must keep the following conditions and limitations in mind:

- AAA-TM assumes that the domain administration server in the referral accepts the same bind credentials as the original server.
- AAA-TM only follows LDAP referrals for password change operations. In other cases AAA-TM refuses to follow the referral.
- AAA-TM only follows one level of LDAP referrals. If the second LDAP server also returns a referral, AAA-TM refuses to follow the second referral.
The ADC supports auditing of all states and status information, so you can see the details of what each user did while logged on, in chronological order. To provide this information, the appliance logs each event, as it occurs, either to a designated audit log file on the appliance or to a syslog server. Auditing requires configuring the appliance and any syslog server that you use.
Enabling AAA

To use the AAA - Application Traffic feature, you must enable it. You can configure AAA entities—such as the authentication and traffic management virtual servers—before you enable the AAA feature, but the entities will not function until the feature is enabled.

To enable AAA by using the command line interface

At the command prompt, type the following commands to enable AAA and verify the configuration:

- enable ns feature AAA
- show ns feature

Example

> enable feature AAA
  Done

> show ns feature

<table>
<thead>
<tr>
<th>Feature</th>
<th>Acronym</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1) Web Logging</td>
<td>WL</td>
<td>OFF</td>
</tr>
<tr>
<td>2) Surge Protection</td>
<td>SP</td>
<td>ON</td>
</tr>
<tr>
<td>15) AAA</td>
<td>AAA</td>
<td>ON</td>
</tr>
<tr>
<td>23) HTML Injection</td>
<td>HTMLInjection</td>
<td>ON</td>
</tr>
<tr>
<td>24) NetScaler Push</td>
<td>push</td>
<td>OFF</td>
</tr>
</tbody>
</table>

Done
To enable AAA by using the configuration utility

1. Navigate to System > Settings.

2. In the details pane, under Modes and Features, click Change basic features.

3. In the Configure Basic Features dialog box, select the Authentication, Authorization and Auditing check box.

4. Click OK.
Setting up AAA Virtual Servers and DNS

You can configure AAA by using the built-in wizard, or manually. To use the wizard, in the main AAA pane of the configuration utility, you click AAA - Application Traffic wizard and follow the prompts.

To configure AAA manually, you first configure an authentication virtual server, which involves binding an SSL certificate-key pair. You then associate the authentication virtual server with a new or existing traffic management virtual server. (Either a load balancing virtual server or a content switching virtual server can serve as a traffic management virtual server.) To complete the initial configuration, you configure DNS to assign hostnames to both the authentication virtual server and the traffic management virtual server, and verify that your virtual servers are UP and configured correctly.

**Caution:** Both virtual servers must have hostnames in the same domain, or the AAA configuration will not work.
Configuring the Authentication Virtual Server

To configure AAA, first configure an authentication virtual server to handle authentication traffic. Next, bind an SSL certificate-key pair to the virtual server to enable it to handle SSL connections. For additional information about configuring SSL and creating a certificate-key pair, see the Citrix NetScaler Traffic Management Guide at "http://support.citrix.com/article/CTX128670."

To configure an authentication virtual server by using the command line interface

To configure an authentication virtual server and verify the configuration, at the command prompt type the following commands in the order shown:

- `add authentication vserver <name> ssl <ipaddress>`
- `show authentication vserver <name>`
- `bind ssl certkey <certkeyName>`
- `show authentication vserver <name>`
- `set authentication vserver <name> -authenticationDomain <FQDN>`
- `show authentication vserver <name>`

Example

```
> add authentication vserver Auth-Vserver-2 SSL 10.102.29.77 443
Done
> show authentication vserver Auth-Vserver-2
  Auth-Vserver-2 (10.102.29.77:443) - SSL Type: CONTENT
  State: DOWN[Certkey not bound]
  Client Idle Timeout: 180 sec
  Down state flush: DISABLED
  Disable Primary Vserver On Down : DISABLED
  Authentication : ON
  Current AAA Users: 0
Done
> bind ssl certkey Auth-Vserver-2 Auth-Cert-1
Done
> show authentication vserver Auth-Vserver-2
  Auth-Vserver-2 (10.102.29.77:443) - SSL Type: CONTENT
  State: UP
  Client Idle Timeout: 180 sec
```
Parameters for configuring the authentication virtual server

**name**

A name for your new authentication virtual server, or the name of an existing authentication virtual server. The name can begin with a letter, number, or the underscore symbol, and can consist of from one to 127 letters, numbers, and the hyphen (-), period (.), pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) symbols. (Cannot be changed after the virtual server is created.)

**ipaddress**

IP address assigned to the authentication virtual server in DNS.

**sslkeyname**

Name of the SSL certificate-key pair to associate with the virtual server.

**authenticationDomain**

The fully qualified domain name to be assigned to the authentication virtual server in your DNS. Must match the domain name in the SSL certificate-key pair.
To configure an authentication virtual server by using the configuration utility


2. In the details pane, do one of the following:
   - To create a new authentication virtual server, click Add.
   - To modify an existing authentication virtual server, select the virtual server, and then click Edit.
     The Configuration dialog opens with the Basic Settings area expanded.

3. Specify values for the parameters as follows (asterisk indicates a required parameter):
   - Name*—name (Cannot be changed for a previously created virtual server)
   - IP Address*—ipaddress
   - Domain*—authenticationDomain
   - Failed login timeout—failedLoginTimeout (Seconds allowed before login fails and user must start login process again.)
   - Max login attempts—maxLoginAttempts (Number of login attempts allowed before user is locked out)

   **Note:** The authentication virtual server uses only the SSL protocol and port 443, so those options are greyed out. Any options that are not mentioned are not relevant and should be ignored.

4. Click Continue to display the Certificates area.

5. In the Certificates area, configure any SSL certificates you want to use with this virtual server.
   - To configure a CA certificate, click the arrow on the right of CA Certificate to display the CA Cert Key dialog box, select the certificate you want to bind to this virtual server, and click Save.
   - To configure a server certificate, click the arrow on the right of Server Certificate, and follow the same process as for CA certificate.

6. Click Continue to display the Advanced Authentication Policies area.

7. If you want to bind an advanced authentication policy to the virtual server, click the arrow on the right side of the line to display the Authentication Policy dialog box, choose the policy that you want to bind to the server, set the priority, and then click OK.

8. Click Continue to display the Basic Authentication Policies area.

9. If you want to create a basic authentication policy and bind it to the virtual server, click the plus sign to display the Policies dialog box, and follow the prompts to configure the policy and bind it to this virtual server.
10. Click Continue to display the 401-Based Virtual Servers area.

11. In the 401-Based Virtual Servers area, configure any load balancing or content switching virtual servers that you want to bind to this virtual server.

   - To bind a load balancing virtual server, click the arrow to the right of LB virtual server to display the LB Virtual Servers dialog box, and follow the prompts.

   - To bind a content switching virtual server, click the arrow to the right of CS virtual server to display the CS Virtual Servers dialog box, and follow the same process as to bind an LB virtual server.

12. If you want to create or configure a group, in the Groups area click the arrow to display the Groups dialog box, and follow the prompts.

13. Review your settings, and when you are finished, click Done. The dialog box closes. If you created a new authentication virtual server, it now appears in the Configuration window list.
Configuring a Traffic Management Virtual Server

After you have created and configured your authentication virtual server, you next create or configure a traffic management virtual server and associate your authentication virtual server with it. You can use either a load balancing or content switching virtual server for a traffic management virtual server. For more information about creating and configuring either type of virtual server, see the Citrix NetScaler Traffic Management Guide at http://support.citrix.com/article/CTX128670Traffic Management.

Note: The FQDN of the traffic management virtual server must be in the same domain as the FQDN of the authentication virtual server for the domain session cookie to function correctly.

You configure a traffic management virtual server for AAA by enabling authentication and then assigning the FQDN of the authentication server to the traffic management virtual server. You can also configure the authentication domain on the traffic management virtual server at this time. If you do not configure this option, the NetScaler appliance assigns the traffic management virtual server an FQDN that consists of the FQDN of the authentication virtual server without the hostname portion. For example, if domain name of the authentication vserver is tm.xyz.bar.com, the appliance assigns xyz.bar.com as the authentication domain.

To configure a TM virtual server for AAA by using the command line interface

At the command prompt, type one of the following sets of commands to configure a TM virtual server and verify the configuration:

- set lb vserver <name> -authentication ON -authenticationhost <FQDN> [-authenticationdomain <authdomain>]
- show lb vserver <name>
- set cs vserver <name> -authentication ON -authenticationhost <FQDN> [-authenticationdomain <authdomain>]
- show cs vserver <name>

Example

> set lb vserver vs-cont-sw -Authentication ON -AuthenticationHost mywiki.index.com
Done
> show lb vserver vs-cont-sw
    vs-cont-sw (0.0.0.0:0) - TCP   Type: ADDRESS
State: DOWN
Last state change was at Wed Aug 19 10:03:15 2009 (+410 ms)
Time since last state change: 5 days, 20:00:40.290
Effective State: DOWN
Client Idle Timeout: 9000 sec
Down state flush: ENABLED
Disable Primary Vserver On Down : DISABLED
No. of Bound Services : 0 (Total) 0 (Active)
Configured Method: LEASTCONNECTION
Mode: IP
Persistence: NONE
Connection Failover: DISABLED
Authentication: ON Host: mywiki.index.com

Done

Parameters for configuring a traffic management virtual server

name

The name of the load balancing or content switching virtual server being configured as a traffic management virtual server.

authentication

Toggle authentication of application traffic on the traffic management virtual server.
Possible values: ON, OFF. Default: OFF.

authenticationhost

FQDN assigned to the authentication virtual server.

authenticationdomain

The common domain in the FQDNs of both the authentication virtual server and the traffic management virtual server.
To configure a TM virtual server for AAA by using the configuration utility

1. In the navigation pane, do one of the following.
   - Expand Load Balancing, and then click Virtual Servers.
   - Expand Content Switching, and then click Virtual Servers.
   The AAA configuration process for either type of virtual server is identical.

2. In the details pane, select the virtual server on which you want to enable authentication, and then click Open.

3. In the Domain text box, type the authentication domain. See authenticationdomain in “Parameters for configuring a traffic management virtual server,” for information about this parameter.

4. On the Advanced tab, select the Authentication check box.

5. In the Authentication Host text box, type the fully qualified domain name of the authentication virtual server. See authenticationhost in the table above for information about this parameter.

6. Click OK. A message appears in the status bar, stating that the vserver has been configured successfully.
Configuring DNS

For the domain session cookie used in the authentication process to function correctly, you must configure DNS to assign both the authentication and the traffic management virtual servers to FQDNs in the same domain. For information about how to configure DNS address records, see the Citrix NetScaler Traffic Management Guide at "http://support.citrix.com/article/CTX128670."
Verifying Your Setup for AAA

After you configure authentication and traffic management virtual servers and before you create user accounts, you should verify that both virtual servers are configured correctly and are in the UP state.

To verify authentication virtual server setup by using the command line interface

At the command prompt, type the following command:

```shell
show authentication vserver <name>
```

Example

```shell
> show authentication vserver Auth-Vserver-2
  Auth-Vserver-2 (10.102.29.77:443) - SSL Type: CONTENT
  State: UP
  Client Idle Timeout: 180 sec
  Down state flush: DISABLED
  Disable Primary Vserver On Down : DISABLED
  Authentication : ON
  Current AAA Users: 0
  Authentication Domain: myCompany.employee.com
Done
```

Parameters for Verifying your Setup for AAA

State

Current state of the service or virtual server. Possible values:

- **UP** - The virtual server can respond to requests. Your authentication virtual server should be UP.

- **OUT OF SERVICE** - The virtual server has been manually disabled. New requests received by this virtual server are dropped unless a backup virtual server or HTTP redirection is configured.

- **DOWN** - The virtual server cannot respond to requests. An authentication virtual server is DOWN when a valid SSL certificate-key pair is not bound to it.

Client Idle Timeout
Idle time (in seconds) after which client connections are terminated. Default value for HTTP/SSL-based services: 180.

**Down state flush**

Perform delayed cleanup of connections on this virtual server. Possible values: ENABLED, DISABLED. Default: ENABLED.

**Disable Primary Vserver On Down**

Keep the primary virtual server secondary, when it comes back up, until manually forced to take over as primary. If enabled, preserves database updates on the backup, enabling you to synchronize the databases before restoring the primary. Possible values: ENABLED, DISABLED. Default: DISABLED.

**Authentication**

Authenticate application traffic for the traffic management virtual server. Possible values: ON, OFF. Default: OFF. This value must be ON for AAA to function.

**Current AAA Users**

Number of AAA users configured. This number should be zero if you have just started to create a AAA configuration.

**Authentication Domain**

Authentication domain configured for the authentication virtual server.

Beneath this information are listed any policies bound globally or to this authentication virtual server, and their priorities.

### To verify your AAA virtual server setup by using the configuration utility

1. In the navigation pane, expand AAA - Application Traffic, and then click Virtual Servers.

2. Review the information in the AAA Virtual Servers pane to verify that your configuration is correct and your authentication virtual server is accepting traffic. You can select a specific virtual server to view detailed information in the details pane.

   **Note:** For descriptions of what the information signifies, see the list above.
Configuring Users and Groups

After configuring the AAA basic setup, you create users and groups. You first create a user account for each person who will authenticate via the NetScaler appliance. If you are using local authentication controlled by the NetScaler appliance itself, you create local user accounts and assign passwords to each of those accounts.

You also create user accounts on the NetScaler appliance if you are using an external authentication server. In this case, however, each user account must exactly match an account for that user on the external authentication server, and you do not assign passwords to the user accounts that you create on the NetScaler. The external authentication server manages the passwords for users that authenticate with the external authentication server.

If you are using an external authentication server, you can still create local user accounts on the NetScaler appliance if, for example, you want to allow temporary users (such as visitors) to log in but do not want to create entries for those users on the authentication server. You assign a password to each local user account, just as you would if you were using local authentication for all user accounts.

Each user account must be bound to policies for authentication and authorization. To simplify this task, you can create one or more groups and assign user accounts to them. You can then bind policies to groups instead of individual user accounts.

To create a local AAA user account by using the command line interface

At the command prompt, type the following commands to create a local AAA user account and verify the configuration:

- `add aaa user <username> [-password <password>]`
- `show aaa user`

Example

```
> add aaa user user-2 -password emptybag
Done
> show aaa user
1)  UserName: user-1
2)  UserName: user-2
Done
```
To change the password for an existing AAA local user account by using the command line interface

At the command prompt, type the following command and, when prompted, type the new password:

```
set aaa user <username>
```

Example

```
> set aaa user user-2
Enter password:
Done
```

Parameters for Configuring AAA Local Users

**username**

A name for the user. The name can begin with a letter, number, or the underscore symbol, and can consist of from one to 127 letters, numbers, and the hyphen (-), period (.), pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) symbols. (Cannot be changed for an existing user.)

**password**

A password that the user uses to log in. This parameter is required for all user accounts if you are not using an external authentication server. If you are using an external authentication server, you provide a password only for local user accounts that do not exist on the authentication server.
To configure AAA local users by using the configuration utility

1. In the navigation pane, expand AAA - Application Traffic, and then click Users.

2. In the details pane, do one of the following:
   - To create a new user account, click Add.
   - To modify an existing user account, select the user account, and then click Open.

3. In the Create AAA User dialog box, in the User Name text box, type a name for the user. For rules for user names, see the list above.

4. If creating a locally authenticated user account, clear the External Authentication check box and provide a local password that the user will use to log on.

5. Click Create or OK, and then click Close. A message appears in the status bar, stating that the user has been configured successfully.

To create AAA local groups and add users to them by using the command line interface

At the command prompt, type the following commands. Type the first command one time, and type the second command once for each user:

- add aaa group <groupname>
- show aaa group

Example

> add aaa group group-2
Done
> show aaa group
1)          GroupName: group-1
2)          GroupName: group-2
Done

- bind aaa group <groupname> -username <username>

Example

> bind aaa group group-2 -username user-2
Done
> show aaa group group-2
  GroupName: group-2

    UserName: user-2
Done
To remove users from an AAA group by using the command line interface

At the command prompt, unbind users from the group by typing the following command once for each user account that is bound to the group:

```
unbind aaa group <groupname> -username <username>
```

Example

```bash
> unbind aaa group group-hr -username user-hr-1
Done
```

To remove an AAA group by using the command line interface

First remove all users from the group. Then, at the command prompt, type the following command to remove an AAA group and verify the configuration:

```
rm aaa group <groupname>
show aaa group
```

Example

```bash
> rm aaa group group-hr
Done
> show aaa group
1) GroupName: group-1
2) GroupName: group-finance
Done
```

Parameters for Configuring AAA Local Groups

**groupname**

A name for the group you are creating. The name can begin with a letter, number, or the underscore symbol, and can consist of from one to 127 letters, numbers, and the hyphen (-), period (.), pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) symbols. (Cannot be changed for existing groups.)

**username**

The name of a user account to be added to the new group.
To configure AAA local groups and add users to them by using the configuration utility

1. In the navigation pane, expand AAA - Application Traffic, and then click Groups.

2. In the details pane, do one of the following:
   - To create a new group, click Add.
   - To modify an existing group, select the group, and then click Open.

3. If you are creating a new group, in the Create AAA Group dialog box, in the Group Name text box, type a name for the group. For information about group names, see the list above, under groupname.

4. On the Users tab, configure the users assigned to the group.
   a. To add a user to the group, select the user, and then click Add.
   b. To remove a user from the group, select the user, and then click Remove.
   c. To create a new user account and add it to the group, click New, and then follow the instructions in “To configure AAA local users by using the configuration utility.”

5. Click Create or OK. The group that you created appears in the AAA Groups page.
Configuring AAA Policies

After you set up your users and groups, you next configure authentication policies, authorization policies, and audit policies to define which users are allowed to access your intranet, which resources each user or group is allowed to access, and what level of detail AAA will preserve in the audit logs. An authentication policy defines the type of authentication to apply when a user attempts to log on. If external authentication is used, the policy also specifies the external authentication server. Authorization policies specify the network resources that users and groups can access after they log on. Auditing policies define the audit log type and location.

You must bind each policy to put it into effect. You bind authentication policies to authentication virtual servers, authorization policies to one or more user accounts or groups, and auditing policies both globally and to one or more user accounts or groups.

When you bind a policy, you assign a priority to it. The priority determines the order in which the policies you define are evaluated. You can set the priority to any positive integer. In the NetScaler operating system, policy priorities work in reverse order: the higher the number, the lower the priority. For example, if you have three policies with priorities of 10, 100, and 1000, the policy assigned a priority of 10 is performed first, then the policy assigned a priority of 100, and finally the policy assigned an order of 1000. The AAA feature implements only the first of each type of policy that a request matches, not any additional policies of that type that a request might also match, so policy priority is important for getting the results you intend.

You can leave yourself plenty of room to add other policies in any order, and still set them to evaluate in the order you want, by setting priorities with intervals of 50 or 100 between each policy when you bind the policies. You can then add additional policies at any time without having to reassign the priority of an existing policy.

For additional information about binding policies on the NetScaler, see the Citrix NetScaler Traffic Management Guide at "http://support.citrix.com/article/CTX128670."
The NetScaler ADC can authenticate users with local user accounts or by using an external authentication server. The appliance supports the following authentication types:

**LOCAL**

Authenticates to the NetScaler by using a password, without reference to an external authentication server. User data is stored locally on the NetScaler appliance.

**RADIUS**

Authenticate to an external Radius server.

**LDAP**

Authenticates to an external LDAP authentication server.

**TACACS**

Authenticates to an external Terminal Access Controller Access-Control System (TACACS) authentication server.

After a user authenticates to a TACACS server, the NetScaler ADC connects to the same TACACS server for all subsequent authorizations. When a primary TACACS server is unavailable, this feature prevents delays while the ADC waits for the first TACACS server to time out before resending the authorization request to the second TACACS server.

*Note:* When authenticating through a TACACS server, AAA-TM logs only successfully executed TACACS commands, to prevent the logs from showing TACACS commands that were entered by users who were not authorized to execute them.

**CERT**

Authenticates to the NetScaler appliance by using a client certificate, without reference to an external authentication server.

**NEGOTIATE**

Authenticates to a Kerberos authentication server. If there is an error in Kerberos authentication, NetScaler uses NTLM authentication.

**SAML**

Authenticates to a server that supports the Security Assertion Markup Language (SAML).

**SAMLIDP**

Configures the NetScaler ADC to serve as a Security Assertion Markup Language (SAML) Identity Provider (IdP).
Authentication Policies

Authenticates to a web server, providing the credentials that the web server requires in an HTTP request and analyzing the web server response to determine that user authentication was successful.

An authentication policy is comprised of an expression and an action. Authentication policies use NetScaler expressions.

After creating an authentication action and an authentication policy, bind it to an authentication virtual server and assign a priority to it. When binding it, also designate it as either a primary or a secondary policy. Primary policies are evaluated before secondary policies. In configurations that use both types of policy, primary policies are normally more specific policies while secondary policies are normally more general policies intended to handle authentication for any user accounts that do not meet the more specific criteria.

To add an authentication action by using the command line interface

If you do not use LOCAL authentication, you need to add an explicit authentication action. To do this, at the command prompt, type the following command:

```
add authentication tacacsAction <name> -serverip <IP> [-serverPort <port>] [-authTimeout <positive_integer>] [...]
```

Example

```
Done
```

To configure an authentication action by using the command line interface

To configure an existing authentication action, at the command prompt, type the following command:

```
set authentication tacacsAction <name> -serverip <IP> [-serverPort <port>] [-authTimeout <positive_integer>] [...]
```

Example

```
Done
```
To remove an authentication action by using the command line interface

To remove an existing RADIUS action, at the command prompt, type the following command:

`rm authentication radiusAction <name>`

**Example**

```
> rm authentication tacacsaction Authn-Act-1
Done
```

Parameters for Configuring an Action

- **name**
  
The name of the action. Called by the `-reqAction` option when configuring the matching policy.

- **-serverip <IP>**
  
The IP the hosts the authentication server, in IPV4 or IPV6 format. The format is autodetected if configured at the command line.

- **-serverPort <port>**
  
The port on which the authentication server accepts connections.

- **-authtimeout <timeout>**
  
The timeout after which the authentication attempt will be deemed to have failed, and the user be notified.

In addition to the parameter listed above, each type of action has its own list of parameters that are specific to that action. For help with LDAP actions, see “LDAP Authentication Policies.” For help with RADIUS actions, see “RADIUS Authentication Policies.” For help with SAML actions, see “SAML Authentication Policies.”

To configure an authentication server by using the configuration utility

**Note:** In the configuration utility, the term server is used instead of action, but refers to the same task.

1. Navigate to AAA - Application Traffic > Policies > Authentication, and then click the type of authentication you want to configure.
2. In the details pane, on the Servers tab, do one of the following:
   - To create a new authentication server, click Add.
   - To modify an existing authentication server, select the server, and then click Open.
3. In the Create Authentication Server or Configure Authentication Server dialog box, type or select values for the parameters. The contents of the dialog box correspond to the parameters described in “Parameters for configuring an action” as follows (asterisk indicates a required parameter):
   - Name*—radiusActionName (Cannot be changed for a previously configured action)
   - Authentication Type*—authtype (Set to RADIUS, cannot be changed)
   - IP Address*—serverip <IP>
   - IPV6*—Select the checkbox if the server IP is an IPv6 IP. (No command line equivalent.)
   - Port*—serverPort
   - Time-out (seconds)*—authTimeout
4. Click Create or OK, and then click Close. The policy that you created appears in the Authentication Policies and Servers page.

To create and bind an authentication policy by using the command line interface

At the command prompt, type the following commands in the order shown to create and bind an authentication policy and verify the configuration:

- add authentication negotiatePolicy <name> <rule> <reqAction>
- show authentication localPolicy <name>
- bind authentication vserver <name> -policy <policyname> [-priority <priority>] [-secondary]
- show authentication vserver <name>

Example

```
> add authentication localPolicy Authn-Pol-1 ns_true
Done

> show authentication localPolicy
1) Name: Authn-Pol-1 Rule: ns_true
   Request action: LOCAL
   Done

> bind authentication vserver Auth-Vserver-2 -policy Authn-Pol-1
Done
```
To modify an existing authentication policy by using the command line interface

At the command prompt, type the following commands to modify an existing authentication policy:

```
set authentication localPolicy <name> <rule> [-reqaction <action>]
```

Example

```
> set authentication localPolicy Authn-Pol-1 'ns_true'
```

Done

To remove an authentication policy by using the command line interface

At the command prompt, type the following command to remove an authentication policy:

```
rm authentication localPolicy <name>
```

Example

```
> rm authentication localPolicy Authn-Pol-1
```

Done

Parameters for configuring authentication policies

authType
Type of authentication. Possible values: LOCAL, RADIUS, LDAP, TACACS, CERT, NEGOTIATE, SAML. Default value: LOCAL

**Note:** If you are creating an authentication policy for a RADIUS server, you must also create an authentication action. See [To create and bind an authentication policy by using the command line interface](http://support.citrix.com/article/CTX128673) for instructions.

**name**

A name for the policy you are creating. The name can begin with a letter, number, or the underscore symbol, and can consist of from one to 127 letters, numbers, and the hyphen (-), period (.), pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) symbols. (Cannot be changed for existing policies.)

**rule (expression)**

An expression that defines the requests to be authenticated. For a complete description of NetScaler expressions, see the *Citrix NetScaler Policy Configuration and Reference Guide* at "http://support.citrix.com/article/CTX128673."

**reqAction**

The action associated with your policy. Leave this blank unless you are using an external authentication server that requires an action. If you are enabling LDAP referral support, see “Authentication Policies” for instructions. If you are creating an authentication policy for a RADIUS server, see “RADIUS Authentication Policies” for instructions. If you are creating an authentication policy for a SAML server, see “SAML Authentication Policies” for instructions.

**authVsName**

The name of the authentication virtual server to which you are binding this policy.

**priority**

The priority assigned to this authentication policy.

**secondary**

Designate this policy as a secondary authentication policy.
To configure and bind authentication policies by using the configuration utility

1. Navigate to Security > AAA - Application Traffic > Policies > Authentication, and then select the type of policy that you want to create.

2. In the details pane, on the Policies tab, do one of the following:
   - To create a new policy, click Add.
   - To modify an existing policy, select the action, and then click Open.

3. In the Create Authentication Policy or Configure Authentication Policy dialog, type or select values for the parameters. The contents of the dialog box correspond to the parameters described in "Parameters for configuring authentication policies" as follows (asterisk indicates a required parameter):
   - Name*—policyname (Cannot be changed for a previously configured action)
   - Authentication Type*—authtype
   - Server*—authVsName
   - Expression*—rule (You enter expressions by first choosing the type of expression in the leftmost drop-down list beneath the Expression window, and then by typing your expression directly into the expression text area, or by clicking Add to open the Add Expression dialog box and using the drop-down lists in it to construct your expression.)

4. Click Create or OK. The policy that you created appears in the Authentication Policies and Servers page.

5. Click the Servers tab, and in the details pane do one of the following:
   - To use an existing server, select it, and then click Open Edit.
   - To create a new server, click Add, and follow the instructions.

6. If you want to designate this policy as a secondary authentication policy, on the Authentication tab, click Secondary. If you want to designate this policy as a primary authentication policy, skip this step.

7. Click Insert Policy.

8. Choose the policy you want to bind to the authentication virtual server from the drop-down list.

9. In the Priority column to the left, modify the default priority as needed to ensure that the policy is evaluated in the proper order.

10. Click OK. A message appears in the status bar, stating that the policy has been configured successfully.
Authorization Policies

After you create authentication policies, you next create any authorization policies you need. Authorization policies, like other policies, consist of an expression and action. There are only two actions for authorization policies: ALLOW and DENY. ALLOW permits users to access the specified resource; DENY blocks access. The default setting for authorization when no specific policy exists is to deny access to network resources. This means that a user or group can access a particular resource only if an authorization policy explicitly allows access. For optimum security, the best practice is not to change the default setting and to create specific authorization policies for users who need access to specific resources.

Authorization use both default syntax expressions and classic expressions. These expressions are described in detail in the Citrix NetScaler Policy Configuration and Reference Guide at "http://support.citrix.com/article/CTX128673."

After you create an authorization policy, you bind it to the appropriate user accounts or groups to put it into effect.

To create an authorization policy

At the NetScaler command prompt, type the following commands to create an authorization policy and verify the configuration:

- add authorization policy <name> <rule> <action>
- show authorization policy <name>

Example

> add authorization policy authz-pol-1 "HTTP.REQ.URL.SUFFIX.EQ(\"gif\")" DENY
Done
> show authorization policy authz-pol-1
1) Name: authz-pol-1 Rule: HTTP.REQ.URL.SUFFIX.EQ("gif")
   Action: DENY
Done
>

To modify an authorization policy

At the command prompt, type the following command to modify an authorization policy:

set authorization policy <name> [-rule <expression>] -action <action>

Example
Authorization Policies

> set authorization policy authz-pol-1 -rule "HTTP.REQ.URL.SUFFIX.EQ(\"gif\")" -action ALLOW
Done
> show authorization policy authz-pol-1
1) Name: authz-pol-1 Rule: HTTP.REQ.URL.SUFFIX.EQ("gif")
   Action: ALLOW
Done
>

To bind an authorization policy to a user account or group

At the command prompt, type one of the following commands to bind an authorization policy to a user account or group and verify the configuration:

- bind aaa user <userName> [-policy <policyname> [-priority <priority>]]
  [-intranetApplication <appname>] [-urlName <urlname>]
  [-intranetIP <intranetip> [netmask]]
- show aaa user <userName>
- bind aaa group <groupName> [-policy <policyname> [-priority <priority>]]
  [-intranetApplication <appname>] [-urlName <urlname>]
  [-intranetIP <intranetip> [netmask]]
- show aaa group <name>

Example

> bind aaa user user-hr-1 -policy authz-pol-1
Done
> show aaa user user-hr-1
   UserName: user-hr-1
   Policy: authz-pol-1, Priority: 0
Done
> bind aaa group group-1 -policy authz-pol-1
Done
> show aaa group group-1
   GroupName: group-1
   UserName: user-2
   UserName: user-1
   Policy: authz-pol-1, Priority: 0
Done
To unbind an authorization policy from a user account or group

At the command prompt, type one of the following commands to unbind an authorization policy from a user account or group:

- `unbind aaa user <userName> -policy <policyname>`
- `unbind aaa group <groupName> -policy <policyname>`

Example

```
> unbind aaa user aaa-user-1 -policy auth-pol-1
Done
```

To remove an authorization policy

First unbind the policy from all user accounts and groups, and then, at the NetScaler command prompt, type the following command to remove an authorization policy:

```
rm authorization policy <name>
```

Parameters for configuring authorization policies

`policyname`

A name for the authorization policy you are creating. The name can begin with a letter, number, or the underscore symbol, and can consist of from one to 127 letters, numbers, and the hyphen (-), period (.), pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) symbols. (Cannot be changed for existing policies.)

`rule`

A NetScaler default syntax or classic syntax expression that defines which requests to allow or deny. For a complete description of NetScaler default syntax and classic syntax expressions, see the *Citrix NetScaler Policy Configuration and Reference Guide* at "http://support.citrix.com/article/CTX128673."

`action`

The action to perform when a connection matches the policy. Possible values: ALLOW, DENY. Default: ALLOW.

`username or groupname`

The name of the user account or the group to which you are binding the authorization policy.

`priority`
Authorization Policies

The priority you are assigning to the policy.

**internetApplication**

The name of the intranet application to which you are binding the authorization policy.

**urlname**

The URL of the intranet application to which you are binding the authorization policy.

**intranetip**

The intranet IP of the intranet application to which you are binding the authorization policy.

**netmask**

If the intranet application to which you are binding the authorization policy resides on an IP range, the subnet mask of that intranet range.
To configure and bind authorization policies by using the configuration utility

1. In the navigation pane, expand AAA - Application Traffic, and then click Authorization.

2. In the details pane, do one of the following:
   
   • To create a new authorization policy, click Add.
   
   • To modify an existing authorization policy, select the policy, and then click Open.

3. In the Create Authorization Policy or Configure Authorization Policy dialog, type or select values for the parameters. The contents of the dialog box correspond to the parameters described in "Parameters for configuring authorization policies" as follows (asterisk indicates a required parameter):
   
   • Name*—policyname (Cannot be changed for a previously configured policy.)
   
   • Action*—action
   
   • Expression*—rule (By default, the Expression box accepts default syntax policies. To switch to the classic syntax view, click Switch to Classic Syntax.)

4. Click Create or OK. The policy that you created appears on the Authorization Policies page.

5. To bind an authorization policy to a user account or group, in the navigation pane, under AAA - Application Traffic, click either Users or Groups, as appropriate, and then add that policy to the user account list:
   
   a. In the details pane, select the appropriate user account, and then click Open.
   
   b. Click the Authorization tab.
   
   c. Click Insert Policy.
   
   d. Select the policy you want to bind to the user account or group.
   
   e. In the Priority column, modify the default priority as needed to ensure that the policy is evaluated in the proper order.
   
   f. Click OK.
   
   A message appears in the status bar, stating that the policy has been configured successfully.
Auditing Policies

After you create authentication policies, you next create any auditing policies you need. The NetScaler ADC allows auditing of all states and status information, so you can see the event history for any user in chronological order. When you configure auditing on the ADC, you can choose to store the log files locally on the ADC or to send them to a syslog server.

To put your auditing policies into effect, you bind them globally, to a specific authentication virtual server, or to specific user accounts or groups.

To create an auditing policy by using the command line interface

At the command prompt, type the following commands to create an auditing policy and verify the configuration:

- add audit nslogPolicy <name> [-rule <rule>] [-action <action>]
- show audit nslogPolicy

Example

> add audit nslogPolicy audit-1 ns_true audit_server
  Done
> show audit nslogPolicy
  1)  Name: audit-pol Rule: ns_true
      Action: audit_server
  2)  Name: audit-1  Rule: ns_true
      Action: audit_server
  Done

To modify an auditing policy by using the command line interface

At the command prompt, type the following commands to modify an auditing policy and verify the configuration:

- set audit nslogPolicy <name> [-rule <expression>] [-action <string>]
- show audit nslogPolicy

Example
To globally bind an auditing policy by using the command line interface

At the command prompt, type the following commands to globally bind an auditing policy:

bind tm global [-policyName <string> [-priority <positive_integer>]]

Example

> bind tm global -policyName Audit-Pol-1 -priority 1000
Done

To bind an auditing policy to an authentication virtual server by using the command line interface

At the command prompt, type the following commands to bind an auditing policy to an authentication virtual server and verify the configuration:

· bind authentication vserver <name> [-policy <string> [-priority <positive_integer>]] [-secondary] [-groupExtraction]]
· show authentication vserver [<name>]

Example

> bind authentication Vserver Auth-Vserver-2 -policy Authn-Pol-1
Done
> show authentication Vserver Auth-Vserver-2
  Auth-Vserver-2 (10.102.29.77:443) - SSL Type: CONTENT
  State: UP
  Client Idle Timeout: 180 sec
  Down state flush: DISABLED
  Disable Primary Vserver On Down : DISABLED
  Authentication : ON
  Current AAA Users: 0
  Authentication Domain: myCompany.employee.com
To bind an auditing policy to a user account or a group by using the command line interface

At the command prompt, type one of the following commands to bind an auditing policy to a user account or a group:

- `bind audit <logtype> user <userName> -policy <policyname> [-priority <priority>]`
- `bind audit <logtype> user <userName> -policy <policyname> [-priority <priority>]`

**Example**

```
> bind audit nslogPolicy user aaa-user-1 -policyName Audit-Pol-1 -priority 1000
Done
```

To unbind a globally bound auditing policy by using the command line interface

At the command prompt, type the following commands to unbind a globally-bound auditing policy:

```
unbind audit <logtype> global -policy <policyname>
```

**Example**

```
> unbind audit nslogPolicy global -policy Audit-Pol-1
Done
```

To unbind an auditing policy from an authentication virtual server by using the command line interface

At the command prompt, type the following commands to unbind an auditing policy from an authentication virtual server:

```
unbind authentication vserver <name> [-policy <string> [-secondary][[-groupExtraction]]]
```

**Example**
To unbind an auditing policy from a user account or a group by using the command line interface

At the command prompt, type one of the following commands to unbind an auditing policy from a user account or a group:

- `unbind audit <logtype> user <userName> -policy <policyname>`
- `unbind audit <logtype> group <groupName> -policy <policyname>`

Example

> unbind audit nslogPolicy group aaa-group-1 -policyName Audit-Pol-1
Done

To remove an auditing policy by using the command line interface

First unbind the policy from all users and groups, and then, at the command prompt, type the following command to remove an auditing policy:

`rm audit <logtype> <policyname>`

Parameters for configuring auditing policies

logtype

Which type of log your policy uses. If your policy logs to nslog, type nslogPolicy. If your policy logs to an external syslog server, type syslogPolicy.

policyname

A name for the policy you are creating. The name can begin with a letter, number, or the underscore symbol, and can consist of from one to 127 letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) symbols. (Cannot be changed for existing policies.)

rule

A NetScaler classic expression that defines which requests to audit. If you do not specify a rule, audit policies default to ns_true, which logs all connections. For a complete description of NetScaler classic expressions, see the Citrix NetScaler Policy Configuration and Reference Guide at "http://support.citrix.com/article/CTX128673."
action

The action associated with this policy. For auditing policies, the action is the name of the nslog or syslog server to which you want to direct connections that match the audit log policy. If you do not specify a server, the default nslog server or syslog server is used.

authvsnname

The name of the authentication virtual server to which you are binding the policy.

username

The name of the user account to which you are binding the policy.

grouppname

The name of the group to which you are binding the policy.

priority

The priority assigned to this auditing policy.
To configure and bind auditing policies by using the configuration utility

1. In the navigation pane, expand AAA - Application Traffic, expand Policies, and then click Auditing.

2. In the details pane, do one of the following:
   - To create a new auditing policy, click Add.
   - To modify an existing auditing policy, select the policy, and then click Edit.

3. In the Create Audit Policy or Configure Audit Policy dialog, type or select values for the parameters. The contents of the dialog box correspond to the parameters described in "Parameters for configuring auditing policies" as follows (asterisk indicates a required parameter):
   - Name*—policyname (Cannot be changed for a previously configured policy.)
   - Auditing type*—logtype (When creating auditing policies by using the configuration utility, you cannot specify a rule.)
   - Server*—action

4. Click Create or OK. The policy that you created appears in the Auditing Policies page.

5. Click OK.

6. To globally bind an auditing policy, in the details pane, click Global Bindings and fill in the Bind/Unbind Audit Policies to Global dialog box.
   a. Select the name of the audit policy you want to globally bind.
   b. Click OK.
   A message appears in the status bar, stating that the policy has been configured successfully.

7. To bind an auditing policy to an authentication virtual server, in the navigation pane, click Virtual Servers, and add that policy to the authentication policies list.
   a. In the details pane, select the appropriate virtual server, and then click Open.
   b. Click the Policies tab.
   c. Click Insert Policy.
   d. Choose the policy you want to bind to the authentication virtual server from the drop-down list.
   e. In the Priority column, modify the default priority as needed to ensure that the policy is evaluated in the proper order.
   f. Click OK.

8. To bind an auditing policy to a user account or group, in the navigation pane, click Users or Groups, and add that policy to the user account list.
a. In the details pane, select the appropriate user account, and then click Open.

b. Click the Policies tab.

c. Click Insert Policy.

d. Choose the policy you want to bind to the group from the drop-down list.

e. In the Priority column, modify the default priority as needed to ensure that the policy is evaluated in the proper order.

f. Click OK.

A message appears in the status bar, stating that the policy has been configured successfully.
Session Settings

After you configure your authentication, authorization, and auditing profiles, you configure session settings to customize your user sessions. The session settings are:

The session timeout.

Controls the period after which the user is automatically disconnected and must authenticate again to access your intranet.

The default authorization setting.

Determines whether the NetScaler appliance will by default allow or deny access to content for which there is no specific authorization policy.

The single sign-on setting.

Determines whether the NetScaler appliance will log users onto all web applications automatically after they authenticate, or will pass users to the web application logon page to authenticate for each application.

The credential index setting.

Determines whether the NetScaler appliance will use primary or the secondary authentication credentials for single signon.

To configure the session settings, you can take one of two approaches. If you want different settings for different user accounts or groups, you create a profile for each user account or group for which you want to configure custom sessions settings. You also create policies to select the connections to which to apply particular profiles, and you bind the policies to users or groups. You can also bind a policy to the authentication virtual server that handles the traffic to which you want to apply the profile.

If you want the same settings for all sessions, or if you want to customize the default settings for sessions that do not have specific profiles and policies configured, you can simply configure the global session settings.
Session Profiles

To customize your user sessions, you first create a session profile. The session profile allows you to override global settings for any of the session parameters.

Note: The terms “session profile” and “session action” mean the same thing.

To create a session profile by using the command line interface

At the command prompt, type the following commands to create a session profile and verify the configuration:

- add tm sessionAction <name> [-sessTimeout <mins>] [-defaultAuthorizationAction (ALLOW | DENY)] [-SSO (ON | OFF)] [-ssoCredential (PRIMARY | SECONDARY)] [-ssoDomain <string>] [-httpOnlyCookie (YES | NO)] [-persistentCookie (ENABLED | DISABLED)] [-persistentCookieValidity <minutes>]

- show tm sessionAction <name>

Example

> add tm sessionAction session-profile -sessTimeout 30 -defaultAuthorization ALLOW
Done
> show tm sessionAction session-profile
  1) Name: session-profile
     Authorization action : ALLOW
     Session timeout: 30 minutes
     Done

To modify a session profile by using the command line interface

At the command prompt, type the following commands to modify a session profile and verify the configuration:

- set tm sessionAction <name> [-sessTimeout <mins>] [-defaultAuthorizationAction (ALLOW | DENY)] [-SSO (ON | OFF)] [-ssoCredential (PRIMARY | SECONDARY)] [-ssoDomain <string>] [-httpOnlyCookie (YES | NO)] [-persistentCookie (ENABLED | DISABLED)] [-persistentCookieValidity <minutes>]

- show tm sessionAction

Example
To remove a session profile by using the command line interface

At the command prompt, type the following command to remove a session profile:

```
rm tm sessionAction <name>
```

Parameters for configuring session profiles

**actionname**

A name for the session action, or the name of the existing session action you want to modify. The name can begin with a letter, number, or the underscore symbol, and can consist of from one to 127 letters, numbers, and the hyphen (-), period (.), pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) symbols. (Cannot be changed for existing session actions.)

**sesstimeout**


**defaultAuthorizationAction**

Whether to allow or deny access to resources by default when no specific policy overrides this setting. Possible values: ALLOW, DENY. Default: DENY

**sso**

Whether to use single sign-on or not for all web applications. Possible values: YES, NO. Default: NO.

**ssocredential**

Whether to use the primary or secondary authentication server for single sign-on credentials. Possible values: PRIMARY, SECONDARY. Default: PRIMARY.

**ssoDomain**

The domain for single sign-on, as a string.

**httpOnlyCookie**
Whether to set the HTTPOnly flag on the sso cookie. Possible values: YES, NO. Default: NO.

**persistentCookie**

Whether to use a persistent sso cookie. Possible values: ENABLED, DISABLED. Default: DISABLED.

**persistentCookieValidity**

The number of minutes, as an integer, that the persistent cookie remains valid.

**To configure session profiles by using the configuration utility**

1. In the navigation pane, expand AAA - Application Traffic, and then click Session.

2. In the details pane, click the Profiles tab.

3. On the Profiles tab, do one of the following:
   - To create a new session profile, click Add.

4. In the Create TM Session Profile or Configure TM Session Profile dialog, type or select values for the parameters. The contents of the dialog box correspond to the parameters described in "Parameters for configuring session profiles" as follows (asterisk indicates a required parameter):
   - **Name**—actionname (Cannot be changed for a previously configured session action.)
   - **Session Time-out**—sesstimeout
   - **Single Signon to Web Applications**—sso
   - **Credential Index**—ssocredential
   - **Single Sign-on Domain**—ssoDomain
   - **HTTPOnly Cookie**—httpOnlyCookie
   - **Enable Persistent Cookie**—persistentCookie
   - **Persistent Cookie Validity**—persistentCookieValidity

5. Click Create or OK. The session profile that you created appears in the Session Policies and Profiles pane.
Session Policies

After you create one or more session profiles, you create session policies and then bind the policies globally or to an authentication virtual server to put them into effect.

To create a session policy by using the command line interface

At the command prompt, type the following commands to create a session policy and verify the configuration:

- `add tm sessionPolicy <name> <rule> <action>
- `show tm sessionPolicy <name>

Example

```
> add tm sessionPolicy session-pol "URL == /*.gif" session-profile
Done
> show tm sessionPolicy session-pol
1)  Name: session-pol   Rule: URL == '/*.gif'
    Action: session-profile
Done
```

To modify a session policy by using the command line interface

At the command prompt, type the following commands to modify a session policy and verify the configuration:

- `set tm sessionPolicy <name> [-rule <expression>] [-action <action>]`
- `show tm sessionPolicy <name>

Example

```
> set tm sessionPolicy session-pol "URL == /*.gif" session-profile
Done
> show tm sessionPolicy session-pol
1)  Name: session-pol   Rule: URL == '/*.gif'
    Action: session-profile
Done
```
To globally bind a session policy by using the command line interface

At the command prompt, type the following commands to globally bind a session policy and verify the configuration:

`bind tm global -policyName <policyname> [-priority <priority>]`

**Example**

```
> bind tm global -policyName session-pol
Done

> show tm sessionPolicy session-pol
1) Name: session-pol   Rule: URL == '/*\.gif'
   Action: session-profile
   Policy is bound to following entities
   1) TM GLOBAL   PRIORITY : 0
Done
```

To bind a session policy to an authentication virtual server by using the command line interface

At the command prompt, type the following command to bind a session policy to an authentication virtual and verify the configuration:

`bind authentication vserver <name> -policy <policyname> [-priority <priority>]`

**Example**

```
> bind authentication vserver auth-vserver-1 -policyName Session-Pol-1 -priority 1000
Done
```

To unbind a session policy from an authentication virtual server by using the command line interface

At the command prompt, type the following commands to unbind a session policy from an authentication virtual server and verify the configuration:

`unbind authentication vserver <name> -policy <policyname>`

**Example**
To unbind a globally bound session policy by using the command line interface

At the command prompt, type the following commands to unbind a globally-bound session policy:

unbind tm global -policyName <policyname>

Example

> unbind tm global -policyName Session-Pol-1
Done

To remove a session policy by using the command line interface

First unbind the session policy from global, and then, at the command prompt, type the following commands to remove a session policy and verify the configuration:

rm tm sessionPolicy <name>

Example

> rm tm sessionPolicy Session-Pol-1
Done

Parameters for configuring session policies

policyname

A name for the session policy, or the name of the existing session policy you want to modify. The name can begin with a letter, number, or the underscore symbol, and can consist of from one to 127 letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) symbols. (Cannot be changed for existing session policies.)

rule
A NetScaler classic expression that defines the requests to select. If you do not specify a rule, session profiles default to ns_true, which selects all connections. For a complete description of NetScaler classic expressions, see the Citrix NetScaler Policy Configuration and Reference Guide at "http://support.citrix.com/article/CTX128673:"

**actionname**

The name of the session profile to apply to connections that match this policy.

**priority**

The priority assigned to this session policy.

**authvsname**

The name of the authentication virtual server to which you are binding this session policy.
To configure and bind session policies by using the configuration utility

1. In the navigation pane, expand AAA - Application Traffic, and then click Session.

2. In the details pane, on the Policies tab, do one of the following:
   - To create a new session policy, click Add.
   - To modify an existing session policy, select the policy, and then click Open.

3. In the Create Session Policy or Configure Session Policy dialog, type or select values for the parameters. The contents of the dialog box correspond to the parameters described in "Parameters for configuring session policies" as follows (asterisk indicates a required parameter):
   - Name*—policyname (Cannot be changed for a previously configured session policy.)
   - Request Profile*—actionname
   - Expression*—rule (You enter expressions by first choosing the type of expression in the leftmost drop-down list beneath the Expression text area and then typing your expression directly into the expression text area, or by clicking Add to open the Add Expression dialog box and using the drop-down lists in it to construct your expression.)

4. Click Create or OK. The policy that you created appears in the details pane of the Session Policies and Profiles page.

5. To globally bind a session policy, in the details pane, click Global Bindings, and fill in the dialog.
   a. Select the name of the session policy you want to globally bind.
   b. Click OK.

6. To bind a session policy to an authentication virtual server, in the navigation pane, click Virtual Servers, and add that policy to the policies list.
   a. In the details pane, select the virtual server, and then click Open.
   b. Click the Policies tab.
   c. Click Insert Policy.
   d. Choose the policy you want to bind to the authentication virtual server from the drop-down list.
   e. In the Priority column to the left, modify the default priority as needed to ensure that the policy is evaluated in the proper order.
   f. Click OK.
   A message appears in the status bar, stating that the policy has been configured successfully.
Global Session Settings

In addition to or instead of creating session profiles and policies, you can configure global session settings. These settings control the session configuration when there is no explicit policy overriding them.

To configure the session settings by using the command line interface

At the command prompt, type the following commands to configure the global session settings and verify the configuration:

```bash
set tm sessionParameter [-sessTimeout <mins>] [-defaultAuthorizationAction ( ALLOW | DENY )] [-SSO ( ON | OFF )] [-ssoCredential ( PRIMARY | SECONDARY )] [-ssoDomain <string>] [-httpOnlyCookie ( YES | NO )] [-persistentCookie ( ENABLED | DISABLED )] [-persistentCookieValidity <minutes>]
```

Example

```
> set tm sessionParameter -sessTimeout 30
Done
> set tm sessionParameter -defaultAuthorizationAction DENY
Done
> set tm sessionParameter -SSO ON
Done
> set tm sessionParameter -ssoCredential PRIMARY
Done
```

Parameters for configuring global session settings

**sessTimeout**

The timeout for user sessions, as an integer value representing a number of minutes.

**defaultAuthorizationAction**

The default authorization action for a user request, when no specific policy is available. Possible values: ALLOW, DENY. Default: DENY.

**sso**

Whether to allow a user to authenticate once and then have access to all web applications on your intranet, or to require authentication for each application. Possible values: ON, OFF. Default: OFF.
Global Session Settings

ssoCredential

If single signon is enabled, which group of credentials to use for authentication. Possible values: Primary, Secondary. Default: Primary.

ssoDomain

The domain for single sign-on, as a string.

httpOnlyCookie

Whether to set the HTTPOnly flag on the sso cookie. Possible values: YES, NO. Default: NO.

persistentCookie

Whether to use a persistent sso cookie. Possible values: ENABLED, DISABLED. Default: DISABLED.

persistentCookieValidity

The number of minutes, as an integer, that the persistent cookie remains valid.

homepage

The authentication home page to which users are redirected.
To configure the session settings by using the configuration utility

1. Navigate to Security > AAA - Application Traffic

2. In the details pane, under Settings, click Change global settings.

3. In the Global Session Settings dialog, type or select values for the parameters. The contents of the dialog box correspond to the parameters described in "Parameters for configuring global session settings" as follows

   · Session Time-out—sessTimeout
   · Single Sign-on to Web Applications—sso
   · Credential Index—ssoCredential
   · Single Sign-on Domain—ssoDomain
   · HTTPOnly Cookie—httpOnlyCookie
   · Enable Persistent Cookie—persistentCookie
   · Persistent Cookie Validity (minutes)—persistentCookieValidity
   · Home Page—homepage

4. Click OK.
Traffic Settings

If you use forms-based or SAML single sign-on (SSO) for your protected applications, you configure that feature in the Traffic settings. SSO enables your users to log on once to access all protected applications, rather than requiring them to log on separately to access each one.

Forms-based SSO allows you to use a web form of your own design as the sign-on method instead of a generic pop-up window. You can therefore put your company logo and other information you might want your users to see on the logon form. SAML SSO allows you to configure one NetScaler appliance or virtual appliance instance to authenticate to another NetScaler appliance on behalf of users who have authenticated with the first appliance.

To configure either type of SSO, you first create a forms or SAML SSO profile. Next, you create a traffic profile and link it to the SSO profile you created. Next, you create a policy, link it to the traffic profile. Finally, you bind the policy globally or to an authentication virtual server to put your configuration into effect.
Form SSO Profiles

To enable and configure forms-based SSO, you first create an SSO profile.

**Note:** In this feature, the terms “profile” and “action” mean the same thing.

**To create a form SSO profile by using the command line interface**

At the command prompt, type:

- `add tm formSSOAction <name> -actionURL <URL> -userField <string> -passwdField <string> -ssoSuccessRule <expression> [-nameValuePair <string>] [-responsesize <positive_integer>] [-nvtype (STATIC | DYNAMIC)] [-submitMethod (GET | POST)]`

- `show tm formSSOAction [<name>]`

**Example**

```
add tm formSSOAction SSO-Prof-1 -actionURL "/logon.php" -userField "loginID" -passwdField "passwd"
-nameValuePair "loginID passwd" -responsesize "9096"
-ssoSuccessRule "HTTP.RES.HEADER("Set-Cookie").CONTAINS("LogonID")"
-nvtype STATIC -submitMethod GET -sessTimeout 10 -defaultAuthorizationAction ALLOW
```

**To modify a form SSO by using the command line interface**

At the command prompt, type:

```
set tm formSSOAction <name> -actionURL <URL> -userField <string> -passwdField <string>
-ssoSuccessRule <expression> [-nameValuePair <string>] [-responsesize <positive_integer>]
[-nvtype (STATIC | DYNAMIC)] [-submitMethod (GET | POST)]
```

**Example**

```
set tm formSSOAction SSO-Prof-1 -actionURL "/logon.php" -userField "loginID" -passwdField "passwd"
-ssoSuccessRule "HTTP.RES.HEADER("Set-Cookie").CONTAINS("LogonID")"
-nameValuePair "loginID passwd" -responsesize "9096"
-nvtype STATIC -submitMethod GET
```
To remove a form SSO profile by using the command line interface

At the command prompt, type:

```
rm tm formSSOAction <name>
```

Example

```
rm tm sessionAction SSO-Prof-1
```

Parameters for configuring form SSO profiles

**name**

A name for the SSO action, or the name of the existing SSO action you want to modify. The name can begin with a letter, number, or the underscore symbol, and can consist of from one to 127 letters, numbers, and the hyphen (-), period (.), pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) symbols. (Cannot be changed for existing SSO actions.)

**actionURL**

The URL where the SSO logon form is located.

**userField**

The form field where the user types in the user ID or login.

**passwdField**

The form field where the user types in the password.

**ssoSuccessRule**

An expression that describes the action that this profile should take when invoked by a policy.

**responsesize**

The number of bytes to allow for the complete response size. Responses that exceed this value are blocked.

**nameValuePair**
To configure form SSO profiles by using the configuration utility

2. In the details pane, click the Form SSO Profiles tab.
3. On the Form SSO Profiles tab, do one of the following:
   - To create a new form SSO profile, click Add.
   - To modify an existing form SSO profile, select the profile, and then click Edit.
4. In the Create Form SSO Profile or Configure Form SSO Profile dialog, specify values for the parameters. The contents of the dialog box correspond to the parameters described in “Parameters for configuring form SSO profiles” as follows (an asterisk indicates a required parameter):
   - Name*—name (Cannot be changed for a previously configured session action.)
   - Action URL*—actionURL
   - User Name Field*—userField
   - Password Field*—passField
   - Expression*—ssoSuccessRule
   - Name Value Pair—nameValuePair
   - Response Size—responsesize
   - Extraction—nvtype
   - Submit Method—submitMethod
5. Click Create or OK, and then click Close. The form SSO profile that you created appears in the Traffic Policies, Profiles, and Form SSO Profiles pane.
Traffic Profiles

After creating at least one forms or SAML sso profile, you must next create a traffic profile.

**Note:** In this feature, the terms “profile” and “action” mean the same thing.

To create a traffic profile by using the command line interface

At the command prompt, type:

```
add tm trafficAction <name> [-appTimeout <mins>] [-SSO ( ON | OFF ) [-formSSOAction <string>]] [-persistentCookie ( ENABLED | DISABLED )] [-InitiateLogout ( ON | OFF )]
```

**Example**

```
add tm trafficAction Traffic-Prof-1 -appTimeout 10 -SSO ON -formSSOAction SSO-Prof-1
```

To modify a session profile by using the command line interface

At the command prompt, type:

```
set tm trafficAction <name> [-appTimeout <mins>] [-SSO ( ON | OFF ) [-formSSOAction <string>]] [-persistentCookie ( ENABLED | DISABLED )] [-InitiateLogout ( ON | OFF )]
```

**Example**

```
set tm trafficAction Traffic-Prof-1 -appTimeout 10 -SSO ON -formSSOAction SSO-Prof-1
```

To remove a session profile by using the command line interface

At the command prompt, type:

```
rm tm trafficAction <name>
```
Traffic Profiles

Example

ru tm trafficAction Traffic-Prof-1

Parameters for configuring traffic profiles

name

A name for the traffic action, or the name of the existing traffic action you want to modify. The name can begin with a letter, number, or the underscore symbol, and can consist of from one to 127 letters, numbers, and the hyphen (-), period (.) pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) symbols. (Cannot be changed for existing SSO actions.)

appTimeout


SSO

Whether to enable or disable SSO. Possible values: ON, OFF. Default: OFF.

formSSOAction

The name of the form SSO profile to use. (Use this only if you are configuring a traffic profile for forms SSO.)

samlSSOAction

The name of the SAML SSO profile to use. (Use this only if you are configuring a traffic profile for SAML SSO.)

persistentCookie

Whether to use a persistent SSO cookie. Possible values: ENABLED, DISABLED. Default: DISABLED.

InitiateLogout

Whether to initiate logout of the traffic management session. Possible values: ON, OFF. Default: OFF.
To configure traffic profiles by using the configuration utility


2. In the details pane, click the Profiles tab.

3. On the Profiles tab, do one of the following:
   
   · To create a new traffic profile, click Add.
   
   · To modify an existing traffic profile, select the profile, and then click Open.

4. In the Create Traffic Profile or Configure Traffic Profile dialog box, specify values for the parameters. The contents of the dialog box correspond to the parameters described in “Parameters for configuring form traffic profiles” as follows (an asterisk indicates a required parameter):
   
   · Name*—name (Cannot be changed for a previously configured session action.)
   
   · AppTimeout—appTimeout
   
   · Single Sign-On—SSO
   
   · Form SSO Action—formSSOAction
   
   · SAML SSO Action—samlSSOAction
   
   · Enable Persistent Cookie—persistentCookie
   
   · Initiate Logout—InitiateLogout

5. Click Create or OK. The traffic profile that you created appears in the Traffic Policies, Profiles, and either the Form SSO Profiles or SAML SSO Profiles pane, as appropriate.
Traffic Policies

After you create one or more form SSO and traffic profiles, you create traffic policies and then bind the policies, either globally or to a traffic management virtual server, to put them into effect.

To create a traffic policy by using the command line interface

At the command prompt, type:

add tm trafficPolicy <name> <rule> <action>

Example

add tm trafficPolicy Traffic-Pol-1 "HTTP.REQ.HEADER("Cookie").CONTAINS(\"login=true\")" Traffic-Prof-1

To modify a traffic policy by using the command line interface

At the command prompt, type:

set tm trafficPolicy <name> <rule> <action>

Example

set tm trafficPolicy Traffic-Pol-1 "HTTP.REQ.HEADER("Cookie").CONTAINS(\"login=true\")" Traffic-Prof-1

To globally bind a traffic policy by using the command line interface

At the command prompt, type:

bind tm global -policyName <string> [-priority <priority>]

Example
To bind a traffic policy to a load balancing or content switching virtual server by using the command line interface

At the command prompt, type one of the following commands:

- `bind lb vserver <name> -policy <policyName> [-priority <priority>]`
- `bind cs vserver <name> -policy <policyName> [-priority <priority>]`

Example

```
bind authentication vserver auth-vserver-1 -policyName Traffic-Pol-1 -priority 1000
```

To unbind a globally bound traffic policy by using the command line interface

At the command prompt, type:

```
unbind tm global -policyName <policyname>
```

Example

```
unbind tm global -policyName Traffic-Pol-1
```

To unbind a traffic policy from a load balancing or content switching virtual server by using the command line interface

At the command prompt, type one of the following commands:

- `unbind lb vserver <name> -policy <policyname>`
- `unbind cs vserver <name> -policy <policyname>`

Example
To remove a traffic policy by using the command line interface

First unbind the session policy from global, and then, at the command prompt, type:

```
rm tm trafficPolicy <name>
```

**Example**

```
rm tm trafficPolicy Traffic-Pol-1
```

**Parameters for configuring form traffic profiles**

**policyName**

A name for the traffic policy, or the name of the existing traffic policy you want to modify. The name can begin with a letter, number, or the underscore symbol, and can consist of from one to 127 letters, numbers, and the hyphen (-), period (.), pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) symbols. (Cannot be changed for existing SSO actions.)

**rule**

A NetScaler advanced expression that defines the requests to select. For a complete description of NetScaler advanced expressions, see the Citrix NetScaler Policy Configuration and Reference Guide at [http://support.citrix.com/article/CTX128673](http://support.citrix.com/article/CTX128673).

**actionname**

The name of the traffic profile to apply to connections that match this policy.

**priority**

The priority assigned to this traffic policy.

**authvsname**

The name of the authentication virtual server to which you are binding this traffic policy.
To configure and bind traffic policies by using the configuration utility

1. In the navigation pane, expand AAA - Application Traffic, then expand Policies, and then click Traffic.

2. In the details pane, do one of the following:
   - To create a new session policy, click Add.
   - To modify an existing session policy, select the policy, and then click Open.

3. In the Create Traffic Policy or Configure Traffic Policy dialog, specify values for the parameters. The contents of the dialog box correspond to the parameters described in “Parameters for configuring form traffic profiles” as follows (an asterisk indicates a required parameter):
   - Name*—policyName (Cannot be changed for a previously configured session policy.)
   - Profile*—actionName
   - Expression—rule (You enter expressions by first choosing the type of expression in the leftmost drop-down list beneath the Expression text area and then typing your expression directly into the expression text area, or by clicking Add to open the Add Expression dialog box and using the drop-down lists in it to construct your expression.)

4. Click Create or OK. The policy that you created appears in the details pane of the Session Policies and Profiles page.

5. To globally bind a traffic policy, in the details pane, click Global Bindings and fill in the Bind/Unbind Policies to Global dialog box.
   a. Select the name of the traffic policy you want to globally bind.
   b. Click OK.

6. To bind a traffic policy to an authentication virtual server, in the navigation pane, click Virtual Servers, and add that policy to the policies list.
   a. In the details pane, select the appropriate virtual server, and then click Open.
   b. Click the Policies tab
   c. Click Insert Policy.
   d. Choose the policy you want to bind to the authentication virtual server from the drop-down list.
   e. In the Priority column to the left, modify the default priority as needed to ensure that the policy is evaluated in the proper order.
   f. Click OK.
Web sites that contain sensitive content, such as online banking websites or websites with employee personal information, sometimes require client certificates for authentication. To configure AAA to authenticate users on the basis of client-side certificate attributes, you first enable client authentication on the traffic management virtual server and bind the root certificate to the authentication virtual server. Then, you implement one of two options. You can configure the default authentication type on the authentication virtual server as CERT, or you can create a certificate action that defines what the NetScaler ADC must do to authenticate users on the basis of a client certificate. In either case, your authentication server must support CRLs. You configure the ADC to extract the user name from the SubjectCN field or another specified field in the client certificate.

When the user tries to log on to an authentication virtual server for which an authentication policy is not configured, and a global cascade is not configured, the user name information is extracted from the specified field of the certificate. If the required field is extracted, the authentication succeeds. If the user does not provide a valid certificate during the SSL handshake, or if the user name extraction fails, authentication fails. After it validates the client certificate, the ADC presents a logon page to the user.

The following procedures assume that you have already created a functioning AAA configuration, and therefore they explain only how to enable authentication by using client certificates. These procedures also assume that you have obtained your root certificate and client certificates and have placed them on the ADC in the /nsconfig/ssl directory.

**To configure the AAA client certificate parameters by using the command line interface**

At the command prompt, type the following commands, in the order shown, to configure the certificate and verify the configuration:

- add ssl certKey <certkeyName> -cert <certFile> -key <keyFile> -password -inform <inform> -expiryMonitor <expiryMonitor> -notificationPeriod <notificationPeriod>

- bind ssl certKey <certkeyName> -vServer <certkeyName> -CA -crlCheck Mandatory

- show ssl certKey [<certkeyName>]

- set aaa parameter -defaultAuthType CERT

- show aaa parameter

- set aaa certParams -userNameField “Subject:CN”

- show aaa certParams
Parameters for authentication with client certificates

**certkeyName**

Name of the certificate-key pair.

**certFile**

Name of the file containing the certificate.

**keyFile**

Name of the file containing the private key.

**inform**

Format of the certificate-key pair. Possible values: DER, PEM. Default: PEM.

**password**

Password for the private key. If you are entering this at the NetScaler command line, you simply include the -password parameter, and then, after entering the command, enter the actual password when prompted. If you are using the configuration utility, you type the password in the appropriate text box.

**bundle**

Parse the certificate chain as a single file after linking the server certificate to its issuer’s certificate within the file. Possible values: YES, NO.

**expiryMonitor**

Monitor the certificate-key pair for expiration. Possible values: ENABLED, DISABLED. Default: ENABLED.

**notificationPeriod**

Period, in days before the certificate-key pair expires, during which the NetScaler ADC should notify the administrator of the impending expiration.

**vservername**

Name of the authentication virtual server to which to bind the root certificate.
To configure the AAA client certificate parameters by using the configuration utility

2. In the details pane, select the virtual server that you want to configure to handle client certificate authentication, and then click Edit.
3. On the Configuration page, under Certificates, click the right arrow (>) to open the CA Cert Key installation dialog.
4. In the CA Cert Key dialog box, click Insert.
5. In the CA Cert Key · SSL Certificates dialog box, click Install.
6. In the Install Certificate dialog box, set the following parameters, whose names correspond to the CLI parameter names as shown:
   - Certificate-Key Pair Name*—certkeyName
   - Certificate File Name—certFile
   - Key File Name—keyFile
   - Certificate Format—inform
   - Password—password
   - Certificate Bundle—bundle
   - Notify When Expires—expiryMonitor
   - Notification Period—notificationPeriod
7. Click Install, and then click Close.
8. In the CA Cert Key dialog box, in the Certificate list, select the root certificate.
9. Click Save.
10. Click Back to return to the main configuration screen.
12. In the details pane, select the policy you want to configure to handle client certificate authentication, and then click Edit.
13. In the Configure Authentication CERT Policy dialog, Server drop-down list, select the virtual server you just configured to handle client certificate authentication.
14. Click OK. A message appears in the status bar, stating that the configuration completed successfully.
Configuring AAA with Commonly Used Protocols

Configuring the NetScaler for Authentication, Authorization, and Auditing (AAA) needs a specific setup on the NetScaler and clients' browsers. The configuration varies with the protocol used for AAA.

For more information about configuring the NetScaler for Kerberos authentication, see Handling Authentication, Authorization and Auditing with Kerberos/NTLM.
Handling Authentication, Authorization and Auditing with Kerberos/NTLM

Kerberos, a computer network authentication protocol, provides secure communication over the Internet. Designed primarily for client-server applications, it provides for mutual authentication by which the client and server can each ensure the other’s authenticity. Kerberos uses a trusted third party, referred to as Key Distribution Center (KDC). A KDC consists of an Authentication Server (AS), which authenticates a user, and a Ticket Granting Server (TGS).

Each entity on the network (client or server) has a secret key that is known only to itself and the KDC. The knowledge of this key implies authenticity of the entity. For communication between two entities on the network, the KDC generates a session key, referred to as the Kerberos ticket or service ticket. The client makes a request to the AS for credentials for a specific server. The client then receives a ticket, referred to as Ticket Granting Ticket (TGT). The client then contacts the TGS, using the TGT it received from the AS to prove its identity, and asks for a service. If the client is eligible for the service, the TGS issues a Kerberos ticket to the client. The client then contacts the server hosting the service (referred to as the service server), using the Kerberos ticket to prove that it is authorized to receive the service. The Kerberos ticket has a configurable lifetime. The client authenticates itself with the AS only once. If it contacts the physical server multiple times, it reuses the AS ticket.

The following figure shows the basic functioning of the Kerberos protocol.

Figure 1. Functioning of Kerberos

Kerberos authentication has the following advantages:

- Faster authentication. When a physical server gets a Kerberos ticket from a client, the server has enough information to authenticate the client directly. It does not have to contact a domain controller for client authentication, and therefore the authentication process is faster.
• Mutual authentication. When the KDC issues a Kerberos ticket to a client and the client uses the ticket to access a service, only authenticated servers can decrypt the Kerberos ticket. If the virtual server on the NetScaler is able to decrypt the Kerberos ticket, you can conclude that both the virtual server and client are authenticated. Thus, the authentication of the server happens along with the authentication of the client.

• Single sign-on between Windows and other operating systems that support Kerberos.

Kerberos authentication may have the following disadvantages:

• Kerberos has strict time requirements; the clocks of the involved hosts must be synchronized with the Kerberos server clock to ensure that the authentication does not fail. You can mitigate this disadvantage by using the Network Time Protocol daemons to keep the host clocks synchronized. Kerberos tickets have an availability period, which you can configure.

• Kerberos needs the central server to be available continuously. When the Kerberos server is down, no one can log on. You can mitigate this risk by using multiple Kerberos servers and fallback authentication mechanisms.

• Because all the authentication is controlled by a centralized KDC, any compromise in this infrastructure, such as the user's password for a local workstation being stolen, can allow an attacker to impersonate any user. You can mitigate this risk to some extent by using only a desktop machine or laptop that you trust, or by enforcing preauthentication by means of a hardware-token.

To use Kerberos authentication, you must configure it on the NetScaler appliance and on each client.
How NetScaler Implements Kerberos Authentication

**Note:** Kerberos/NTLM authentication is supported only in the NetScaler 9.3 nCore release or later, and it can be used only for AAA traffic management (AAA-TM) virtual servers.

NetScaler handles the components involved in Kerberos authentication in the following way:

### Key Distribution Center (KDC)

In the Windows 2000 Server or later versions, the Domain Controller and KDC are part of the Windows Server. If the Windows Server is UP and running, it indicates that the Domain Controller and KDC are configured. The KDC is also the Active Directory server.

**Note:** All Kerberos interactions are validated with the Windows Kerberos Domain Controller.

### Authentication Service and Protocol Negotiation

A NetScaler appliance supports Kerberos authentication on the AAA-TM authentication virtual servers. If the Kerberos authentication fails, the NetScaler uses the NTLM authentication.

By default, Windows 2000 Server and later Windows Server versions use Kerberos for AAA. If you create an authentication policy with NEGOTIATE as the authentication type, the NetScaler attempts to use the Kerberos protocol for AAA and if the client's browser fails to receive a Kerberos ticket, the NetScaler uses the NTLM authentication. This process is referred to as *negotiation*.

The client may fail to receive a Kerberos ticket in any of the following cases:

- Kerberos is not supported on the client.
- Kerberos is not enabled on the client.
- The client is in a domain other than that of the KDC.
- The Access Directory on the KDC is not accessible to the client.

For Kerberos/NTLM authentication, the NetScaler does not use the data that is present locally on the NetScaler appliance.

### Authorization

The traffic management virtual server can be a load balancing virtual server or a content switching virtual server.

### Auditing
The NetScaler appliance supports auditing of Kerberos authentication with the following audit logging:

- Complete audit trail of the traffic management end-user activity
- SYSLOG and high performance TCP logging
- Complete audit trail of system administrators
- All system events
- Scriptable log format

**Supported Environment**

Kerberos authentication does not need any specific environment on the NetScaler. The client (browser) must provide support for Kerberos authentication.

**High Availability**

In a high availability setup, only the active NetScaler joins the domain. In case of a failover, the NetScaler lwagent daemon joins the secondary NetScaler appliance to the domain. No specific configuration is required for this functionality.

**Kerberos Authentication Process**

The following figure shows a typical process for Kerberos authentication in the NetScaler environment.

![Kerberos Authentication Process on NetScaler](image)

Figure 1. Kerberos Authentication Process on NetScaler
How NetScaler Implements Kerberos Authentication

The Kerberos authentication occurs in the following stages:

**Client authenticates itself to the KDC.**

1. The NetScaler appliance receives a request from a client.

2. The traffic management (load balancing or content switching) virtual server on the NetScaler sends a challenge to the client.

3. To respond to the challenge, the client gets a Kerberos ticket.
   - The client sends the Authentication Server of the KDC a request for a ticket-granting ticket (TGT) and receives the TGT. (See 3, 4 in the figure, Kerberos Authentication Process.)
   - The client sends the TGT to the Ticket Granting Server of the KDC and receives a Kerberos ticket. (See 5, 6 in the figure, Kerberos Authentication Process.)

**Note:** The above authentication process is not necessary if the client already has a Kerberos ticket whose lifetime has not expired. In addition, clients such as Web Services, .NET, or J2EE, which support SPNEGO, get a Kerberos ticket for the target server, create an SPNEGO token, and insert the token in the HTTP header when they send an HTTP request. They do not go through the client authentication process.

**Client requests a service.**

1. The client sends the Kerberos ticket containing the SPNEGO token and the HTTP request to the traffic management virtual server on the NetScaler. The SPNEGO token has the necessary GSSAPI data.

2. The NetScaler establishes a security context between the client and the NetScaler. If the NetScaler cannot accept the data provided in the Kerberos ticket, the client is asked to get a different ticket. This cycle repeats till the GSSAPI data is acceptable and the security context is established. The traffic management virtual server on the NetScaler acts as an HTTP proxy between the client and the physical server.

**NetScaler completes the authentication.**

1. After the security context is complete, the traffic management virtual server validates the SPNEGO token.

2. From the valid SPNEGO token, the virtual server extracts the user ID and GSS credentials, and passes them to the authentication daemon.

3. A successful authentication completes the Kerberos authentication.
Kerberos Authentication - Configuration on the NetScaler Appliance

To configure Kerberos authentication on the NetScaler appliance, perform the following tasks:

1. Enable the Authentication, Authorization, and Auditing (AAA) feature on the NetScaler appliance.

2. On the Active Directory, add a user for Kerberos authentication, map the HTTP service to this user, and generate a keytab file and import it to the NetScaler appliance. You can map more than one service if the Kerberos authentication is required for more than one service. The keytab file should contain entries for every service that is bound to the traffic management virtual server on the NetScaler. The keytab file is necessary for decrypting the secret received from the client during Kerberos authentication. The authentication details of all the services are stored in a single keytab file on the NetScaler.

3. Add a DNS server.

   **Note:** The NetScaler must obtain the IP address of the domain controller from the fully qualified domain name (FQDN). Therefore, Citrix recommends configuring the NetScaler with a DNS server. A less preferred alternative is to create a static DNS entry.

4. Create an authentication negotiation policy with a negotiation action.

5. Configure an authentication server and bind the authentication policy to the authentication virtual server.

6. Configure an authentication service and a traffic management virtual server, and bind the service to the virtual server. You can use either a load balancing or a content switching virtual server.

7. Verify the configuration.
Enabling AAA on the NetScaler

Enable authentication of the traffic on the NetScaler appliance.

To enable Authentication, Authorization, and Auditing (AAA) by using the command line interface

At the command prompt, type the following commands to enable AAA and verify the configuration:

- enable ns feature AAA
- show ns feature

Example

> enable feature aaa
Done
> show ns feature
Feature Acronym Status
------- ------- ------
1) Web Logging WL ON
... 
3) Load Balancing LB ON
4) Content Switching CS ON
5) Cache Redirection CR ON
...
14) SSL VPN SSLVPN ON
15) AAA AAA ON
...
26) CloudBridge CloudBridge OFF
Done
To enable Authentication, Authorization, and Auditing (AAA) on NetScaler by using the configuration utility

1. In the navigation pane, expand System, and then click Settings.

2. In the details pane, under Modes and Features, click Configure basic features.

3. In the Configure Basic Features dialog box, select the Authentication, Authorization and Auditing check box.

4. Click OK.

5. In the confirmation dialog box, click Yes. A message appears in the status bar to indicate that the feature is enabled.
Adding a Keytab file

The keytab file contains information about services necessary for Kerberos authentication. The keytab file is necessary for decrypting the secret received from the client during Kerberos authentication. You can map more than one service if the Kerberos authentication is required for more than one service. The keytab file should contain entries for every service that is bound to the traffic management virtual server on the NetScaler. The authentication details of all the services are stored in a single keytab file on the NetScaler.

To generate a keytab file and import it to the NetScaler appliance, follow the procedure described below:

**Note:** You can generate the keytab file and import it onto the NetScaler only from the command line.

1. Log onto the Active Directory server and create a user for Kerberos authentication.
   For example, type the following command:
   ```
   net user Kerb-SVC-Account freebsd!@#456 /add
   ```

2. In the User Properties section, ensure the following settings:
   - The Change password at next logon option is not selected.
   - The Password does not expire option is selected.

3. Map the HTTP service to the above user and export the keytab file. For example, run the following command on the Active Directory server:
   ```
   ktpass /out keytabfile /princ HTTP/owa.newacp.com@NEWACP.COM /pass freebsd!@#456 /mapuser newacp\dummy /ptype KRB5_NT_PRINCIPAL
   ```
   **Note:** If you want to map more services, repeat the above command for every service. You can give the same name or different names for the output file.

4. Transfer the keytab file to the NetScaler by using the unix ftp command or any other file transfer utility of your choice.

5. Log onto the NetScaler appliance, and run the ktutil utility to verify the keytab file. The keytab file has an entry for the HTTP service after it is imported.

**Example**

```
root@ns# ktutil
ktutil: rkt /var/keytabfile
ktutil: list
slot KVNO Principal
--------------------------------------------------------------------
```
Adding a Keytab file

ktutil: wkt /etc/ krb5.keytab
ktutil: list
slot KVNO Principal

1 2 HTTP/owa.newacp.com@NEWACP.COM
ktutil: quit
Adding a DNS Server

The NetScaler appliance should obtain the IP address of the domain controller from the fully qualified domain name (FQDN). Therefore, Citrix recommends configuring the NetScaler with a DNS server. A less preferred alternative is to create a static DNS entry.

To add a DNS server by using the command line interface

At the command prompt, type the following command:

add dns nameserver <IP>

Note: Alternatively, you can add static host entries or use any other means so that the NetScaler can resolve the FQDN name of the domain controller to an IP address.

Example

add dns nameserver 1.2.3.4

Parameters for configuring the DNS server

dnsIpAddress

The IP address of the name server that is used to resolve domain names to IP addresses.

To add a DNS server by using the NetScaler configuration utility

1. In the navigation pane, expand DNS, and then click Name Servers.
2. In the details pane, click Add.
3. In the IP Address box, type the IP address.
4. Click Create, and then Close.
5. Verify that the details pane shows the newly added DNS server.
Creating an Authentication Negotiation Policy

Create a negotiation policy with a negotiation action for Kerberos authentication of services.

To create an authentication negotiation policy by using the command line interface

At the command prompt, type the following commands:

- add authentication negotiateAction <name> -domain <domainName> -domainUser <domainUsername> -domainUserPasswd <domainUserPassword> -encrypted

- add authentication negotiatePolicy <name> <rule> <reqAction>

Example

add authentication negotiateAction negact -domain newacp.com -domainUser Administrator -domainUserPasswd skp5sep
add authentication negotiatePolicy negopol ns_true negact

Parameters for creating an authentication negotiation policy

negotiateActionName
The name of the negotiate action associated with the negotiate policy.

domainName
The fully qualified domain name in which the client and KDC are present.

domainUsername
The user name of the user who can access the domain.

domainUserPassword
The password of the user who can access the domain.
Creating an Authentication Negotiation Policy

negotiatePolicyName

A name for the policy you are creating. The name can begin with a letter, number, or the underscore symbol, and can consist of from one to 127 letters, numbers, and the hyphen (-), period (.), pound (#), space ( ), at sign (@), equals (=), colon (:) and underscore (_) symbols. (Cannot be changed for existing policies.)

negotiatePolicyExpression

A policy expression that defines the requests to be authenticated.

To create an authentication negotiation policy by using the NetScaler configuration utility

1. In the navigation pane, expand AAA-Application Traffic, expand Policies, and then click Authentication.

2. In the details pane, on the Policies tab, click Add.

3. In the Create Authentication Policy dialog box, set the following parameters:

   - Name
   - Authentication Type - Select NEGOTIATE.
   - Server - Select an existing server from the dropdown list. To add a new authentication server, click New..., and in the Create Authentication Server dialog box, set the following parameters:
     - Domain Name
     - User Name
     - Password
     - Confirm Password - Retype the password.
   - Expression - In the Named Expression list, select General and select True Value from the dropdown list, and then click Add Expression.

4. Click Create, and then click Close.

5. Verify that the policy you created appears in the Authentication Policies and Servers pane.
Creating an Authentication Virtual Server

Configure an authentication virtual server and bind the authentication negotiation policy to the authentication virtual server.

To create an authentication virtual server and bind the negotiation policy by using the command line interface

At the command prompt, type the following commands:

- `add authentication vserver <name> SSL <ipAuthVserver> 443 -authenticationDomain <domainName>`
- `bind authentication vserver <name> -policy <negotiatePolicyName>`

Example

```
add authentication vserver authen1 SSL 10.102.113.166 443 -authenticationDomain newacp.com
add ssl certKey cert1 -cert "/nsconfig/ssl/complete/server/server_rsa_2048.pem" -key "/nsconfig/ssl/complete/server/server_rsa_2048.ky"
bind ssl vserver authen1 -certkeyName cert1
bind authentication vserver authen1 -policy negopol
```

Parameters for configuring an authentication virtual server and binding the negotiation policy

**authVserverName**

A name for the new authentication virtual server. The name can begin with a letter, number, or the underscore symbol, and can consist of from one to 127 letters, numbers, and the hyphen (-), period (.), pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) symbols. (Cannot be changed after the virtual server is created.)

**ipAuthVserver**

The IP address of the authentication virtual server.

**domainName**

The fully qualified domain name in which the client and KDC are present. This domain is assigned to the authentication virtual server.

**negotiatePolicyName**
A name for the policy you are creating. The name can begin with a letter, number, or the underscore symbol, and can consist of from one to 127 letters, numbers, and the hyphen (-), period (.), pound (#), space ( ), at sign (@), equals (=), colon (:), and underscore (_) symbols. (Cannot be changed for existing policies.)

To create an authentication virtual server and bind the negotiation policy by using the NetScaler configuration utility

1. In the navigation pane, expand AAA - Application Traffic and click Virtual Servers.

2. In the details pane, click Add.

3. In the Create Virtual Server (Authentication) dialog box, set the following parameters:
   
   - Name
   - IP Address
   - Protocol - Select SSL
   - Domain - Type the fully qualified domain name added while creating the keytab file.

   **Note:** For AAA, the protocol must be SSL protocol and port must be 443. Therefore, these options are not provided.

4. On the Authentication tab, click Insert Policy. In the Authentication Policies group, from the Policy Name dropdown list, select the negotiate authentication policy you added for Kerberos authentication.

5. On the Certificates tab, select an SSL certificate from the list of available certificates, and then click Add. If the certificate you want to bind is not displayed in the Available Certificates list, click Install..., and then select the certificate file.

6. Click Create, and then click Close. The new authentication virtual server appears in the Authentication Virtual Servers pane.
Configuring a Traffic Management Virtual Server

Configure an authentication service and a traffic management virtual server, and bind the service to the virtual server. You can use either a load balancing or a content switching virtual server.

To create a traffic management virtual server and service, and bind the service by using the command line interface

At the command prompt, type the following commands:

- `add service <name>@<ipBackendWebserver> HTTP 80`
- `add lb vserver <name>@SSL <ipAddressLbVserver> 443 -authn401 ON -authnVsName <authVserverName>`
- `bind lb vserver <name>@<serviceName>`

**Note:** Use a similar procedure for using a content switching virtual server as the traffic management virtual server.

Example

```
add service svc1 10.217.28.92 HTTP 80
add lb vserver v2 HTTP 10.102.113.164 80 -persistenceType NONE -cltTimeout 180 -authn401 ON -authnVsName
bind lb vserver v2 svc1
```

Parameters for configuring the traffic management virtual server

**serviceName**

Name of the service. The name must not exceed 127 characters, and the leading character must be a number or letter. The following characters are also allowed: @ _ . (period) : (colon) # and space ( ).

**ipBackendWebServer**
To create a traffic management virtual server and service, and bind the service by using the NetScaler configuration utility

1. In the navigation pane, expand Load Balancing and click Services.

2. In the details pane, click Add.

3. In the Create Service dialog box, set the following parameters:
   - Service Name
   - Server
   - Protocol - Select HTTP.
   - Port - Select 80.

4. In the navigation pane, expand Load Balancing and click Virtual Servers.

5. In the details pane, click Add.

6. In the Create Virtual Server (Load balancing) dialog box, set values for the following parameters:
   - Name
   - IP Address
Configuring a Traffic Management Virtual Server

- Protocol
- Port

7. In the Create Virtual Server (Load balancing) dialog box, on the Services tab, select the service you created in Step 3 to Step 5.

8. In the Create Virtual Server (Load balancing) dialog box, on the Advanced tab, expand Authentication Settings, and then select the 401 Based Authentication check box.

9. Click Create, and then click Close. The new load balancing virtual server appears in the Load Balancing Virtual Servers pane.

10. In the details pane, verify the settings of the virtual server.

   **Note:** Use a similar procedure to create a content switching virtual server.

   **Note:** For more information, see Setting up basic load balancing.
Verifying the configuration for Kerberos Authentication

Ensure that you completed the following tasks and verify whether the configuration is complete and correct.

- Enable the AAA feature
- Import the keytab file
- Configure the DNS server
- Configure negotiation policies and actions
- Configure authentication virtual server
- Configure traffic management virtual server

To verify the configuration:

2. View the AAA session on the NetScaler. show aaa session

Example

ClientIp (ClientPort) ->ServerIp(ServerPort)
----------------------------------------------
PE id : 4
User name: john.smith@NEWACP.COM Session Type: TM
Done
Configuration of Kerberos Authentication on a Client

Kerberos support must be configured on the browser to use Kerberos for authentication. You can use any Kerberos-compliant browser. Instructions for configuring Kerberos support on Internet Explorer and Mozilla Firefox follow. For other browsers, see the documentation of the browser.

To configure Internet Explorer for Kerberos authentication

1. In the Tools menu select Internet Options.

2. On the Security tab, click Local Intranet, and then click Sites.

3. In the Local Intranet dialog box, make sure that the Automatically detect intranet network option is selected, and then click Advanced.

4. In the Local Intranet dialog box, add the web sites of the domains of the traffic management virtual server on the NetScaler. The specified sites become local intranet sites.

5. Click Close or OK to close the dialog boxes.

To configure Mozilla Firefox for Kerberos authentication

1. Make sure that you have Kerberos properly configured on your computer.

2. Type `about:config` in the URL bar.

3. In the filter text box, type `network.negotiate`.

4. Change `network.negotiate-auth.delegation-uris` to the domain that you want to add.

5. Change `network.negotiate-auth.trusted-uris` to the domain that you want to add.

   **Note:** If you are running Windows, you also need to enter `sspi` in the filter text box and change the `network.auth.use-sspi` option to `False`. 
Offloading Kerberos Authentication from Physical Servers

The NetScaler appliance can offload authentication tasks from servers. Instead of the physical servers authenticating the requests from clients, the NetScaler authenticates all the client requests before it forwards them to any of the physical servers bound to it. The user authentication is based on Active Directory tokens.

There is no authentication between the NetScaler and the physical server, and the authentication offload is transparent to the end users. After the initial logon to a Windows computer, the end user does not have to enter any additional authentication information in a pop-up or on a logon page.

In the current NetScaler release, Kerberos authentication is available only for Authentication, Authorization, and Auditing (AAA) Traffic Management Virtual Servers. Kerberos authentication is not supported for SSL VPN in the Access Gateway Enterprise Edition appliance or for NetScaler appliance management.

Kerberos authentication requires configuration on the NetScaler appliance and on client browsers.
To configure Kerberos authentication on the NetScaler appliance

1. Create a user account on Active Directory. When creating a user account, verify the following options in the User Properties section:

   - Make sure that you do not select the Change password at next logon option.
   - Be sure to select the Password does not expire option.

2. On the NetScaler appliance, at the CLI command prompt, type:

   ktpass -princ HTTP/kerberos.crete.lab.net@crete.lab.net -ptype KRB5_NT_PRINCIPAL -mapuser kerbuser@crete.lab.net -mapop set -pass Citrix1 -out C:\kerbtabfile.txt

   **Note:** Be sure to type the above command on a single line. The output of the above command is written into the C:\kerbtabfile.txt file.

3. Upload the kerbtabfile.txt file to the /etc directory of the NetScaler appliance by using a Secure Copy (SCP) client.

4. Run the following command to add a DNS server to the NetScaler appliance.

   add dns nameserver 1.2.3.4

   The NetScaler appliance cannot process Kerberos requests without the DNS server. Be sure to use the same DNS server that is used in the Microsoft Windows domain.

5. Switch to the shell prompt and run the following commands from the shell prompt:

   - ktutil # rkt /etc/kerbtabfile.txt
   - # wkt /etc/krb5.keytab
   - # list

   The list command displays the user account details that you created in the Active Directory. A sample screen of the output of the list command is shown below.
6. Switch to the command line interface of NetScaler.

7. Run the following command to create a Kerberos authentication server:
   ```plaintext
   add authentication negotiateAction KerberosServer -domain "crete.lab.net"
   -domainUser kerbuser -domainUserPasswd Citrix1
   ```

8. Run the following command to create a negotiation policy:
   ```plaintext
   add authentication negotiatePolicy Kerberos-Policy "REQ.IP.DESTIP ==
   192.168.17.200" KerberosServer
   ```

9. Run the following command to create an authentication virtual server:
   ```plaintext
   add authentication vserver Kerb-Auth SSL 192.168.17.201 443
   -AuthenticationDomain crete.example.com
   ```

10. Run the following command to bind the Kerberos policy to the authentication virtual server:
    ```plaintext
        bind authentication vserver Kerb-Auth -policy Kerberos-Policy -priority 100
    ```

11. Run the following command to bind an SSL certificate to the authentication virtual server. You can use one of the test certificates, which you can install from the GUI NetScaler appliance. Run the following command to use the ServerTestCert sample certificate.
    ```plaintext
        bind ssl vserver Kerb-Auth -certKeyName ServerTestCert
    ```

12. Create an HTTP load balancing virtual server with the IP address, 192.168.17.200.
    ```plaintext
        Ensure that you create a virtual server from the command line interface for NetScaler 9.3 releases if they are older than 9.3.47.8.
    ```

13. Run the following command to configure an authentication virtual server:
    ```plaintext
        set lb vserver <name>-authn401 ON -authnVsName Kerb-Auth
    ```

14. Enter the host name `http://www.crete.example.com` in the address bar of the Web browser.

    The Web browser displays an authentication dialog box because the Kerberos authentication is not set up in the browser.
**Offloading Kerberos Authentication from Physical Servers**

**Note:** Kerberos authentication requires a specific configuration on the client. Ensure that the client can resolve the hostname, which results in the Web browser connecting to an HTTP virtual server.

15. Configure Kerberos on the Web browser of the client computer.

1. For configuring on Internet Explorer, see "Configuring Internet Explorer for Kerberos authentication."

2. For configuring on Mozilla Firefox, see "Configuring Mozilla Firefox for Kerberos authentication."

16. Verify whether you can access the backend physical server without authentication.

---

**To configure Internet Explorer for Kerberos authentication**

1. Select Internet Options from the Tools menu.


3. Select Local Intranet from the Select a zone to view change security settings section.

4. Click Sites.

5. Click Advanced.


7. Restart Internet Explorer.
To configure Mozilla Firefox for Kerberos authentication

1. Enter `about:config` in the address bar of the browser.

2. Click the warning disclaimer.

3. Type `Network.Negotiate-auth.trusted-uris` in the Filter box.


   ![about:config](image)

<table>
<thead>
<tr>
<th>Preference Name</th>
<th>Status</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>network.negotiate-auth.allow-proxies</td>
<td>default</td>
<td>boolean</td>
</tr>
<tr>
<td>network.negotiate-auth.delegation-uris</td>
<td>default</td>
<td>string</td>
</tr>
<tr>
<td>network.negotiate-auth.gsslib</td>
<td>default</td>
<td>string</td>
</tr>
<tr>
<td>network.negotiate-auth.trusted-uris</td>
<td>default</td>
<td>string</td>
</tr>
</tbody>
</table>

5. In the Enter String Value dialog box, specify `www.crete.example.com`.

6. Restart Firefox.