



# CHAPTER 15

## Applying AAA for Network Access

---

This chapter describes how to enable AAA (pronounced “triple A”) for network access.

For information about AAA for management access, see the [“AAA for System Administrators”](#) section on page 21-11.

This chapter contains the following sections:

- [AAA Performance, page 15-1](#)
- [Configuring Authentication for Network Access, page 15-1](#)
- [Configuring Authorization for Network Access, page 15-9](#)
- [Configuring Accounting for Network Access, page 15-13](#)
- [Using MAC Addresses to Exempt Traffic from Authentication and Authorization, page 15-14](#)

### AAA Performance

The FWSM uses “cut-through proxy” to significantly improve performance compared to a traditional proxy server. The performance of a traditional proxy server suffers because it analyzes every packet at the application layer of the OSI model. The FWSM cut-through proxy challenges a user initially at the application layer and then authenticates against standard RADIUS, TACACS+, or the local database. After the FWSM authenticates the user, it shifts the session flow, and all traffic flows directly and quickly between the source and destination while maintaining session state information.

### Configuring Authentication for Network Access

This section includes the following topics:

- [Authentication Overview, page 15-2](#)
- [Enabling Network Access Authentication, page 15-3](#)
- [Configuring Custom Login Prompts, page 15-5](#)
- [Enabling Secure Authentication of Web Clients, page 15-6](#)
- [Disabling Authentication Challenge per Protocol, page 15-8](#)

## Authentication Overview

The FWSM lets you configure network access authentication using AAA servers. This section includes the following topics:

- [One-Time Authentication, page 15-2](#)
- [Applications Required to Receive an Authentication Challenge, page 15-2](#)
- [Static PAT and HTTP, page 15-3](#)
- [Authenticating Directly with the FWSM, page 15-3](#)

### One-Time Authentication

A user at a given IP address only needs to authenticate one time for all rules and types, until the authentication session expires. (See the **timeout uauth** command in the *Catalyst 6500 Series Switch and Cisco 7600 Series Router Firewall Services Module Command Reference* for timeout values.) For example, if you configure the FWSM to authenticate Telnet and FTP, and a user first successfully authenticates for Telnet, then as long as the authentication session exists, the user does not also have to authenticate for FTP.

For HTTP or HTTPS authentication, once authenticated, a user never has to reauthenticate, no matter how low the **timeout uauth** command is set, because the browser caches the string “Basic=Uuhjksdkfhk==” in every subsequent connection to that particular site. This can be cleared only when the user exits *all* instances of the web browser and restarts. Flushing the cache is of no use.

### Applications Required to Receive an Authentication Challenge

Although you can configure the FWSM to require authentication for network access to any protocol or service, users can authenticate directly with HTTP, HTTPS, Telnet, or FTP only. A user must first authenticate with one of these services before the FWSM allows other traffic requiring authentication.

The authentication ports that the FWSM supports for AAA are fixed:

- Port 21 for FTP
- Port 23 for Telnet
- Port 80 for HTTP
- Port 443 for HTTPS

For Telnet and FTP, the FWSM generates an authentication prompt. After you authenticate correctly, the FWSM redirects you to your original destination. If the destination server also has its own authentication, you enter another username and password.

For HTTP, you log in using basic HTTP authentication supplied by the browser. For HTTPS, the FWSM generates custom login windows.

**Note**

If you use HTTP authentication without using the **aaa authentication secure-http-client** command, the username and password are sent from the client to the FWSM in clear text. We recommend that you use the **aaa authentication secure-http-client** command whenever you enable HTTP authentication. For more information about the **aaa authentication secure-http-client** command, see the “[Enabling Secure Authentication of Web Clients](#)” section on page 15-6.

For FTP, a user has the option of entering the FWSM username followed by an at sign (@) and then the FTP username (name1@name2). For the password, the user enters the FWSM password followed by an at sign (@) and then the FTP password (password1@password2). For example, enter the following text.

```
name> jamiec@jchrichton
password> letmein@he110
```

This feature is useful when you have cascaded firewalls that require multiple logins. You can separate several names and passwords by multiple at signs (@).

## Static PAT and HTTP

For HTTP authentication, the FWSM checks real ports when static PAT is configured. If it detects traffic destined for real port 80, regardless of the mapped port, the FWSM intercepts the HTTP connection and enforces authentication.

For example, assume that outside TCP port 889 is translated to port 80 (www) and that any relevant access lists permit the traffic:

```
static (inside,outside) tcp 10.48.66.155 889 192.168.123.10 www netmask 255.255.255.255
```

Then when users try to access 10.48.66.155 on port 889, the FWSM intercepts the traffic and enforces HTTP authentication. Users see the HTTP authentication page in their web browsers before the FWSM allows HTTP connection to complete.

If the local port is different than port 80, as in the following example:

```
static (inside,outside) tcp 10.48.66.155 889 192.168.123.10 111 netmask 255.255.255.255
```

Then users do not see the authentication page. Instead, the FWSM sends to the web browser an error message indicating that the user must be authenticated prior using the requested service.

## Authenticating Directly with the FWSM

If you do not want to allow HTTP(S), Telnet, or FTP through the FWSM but want to authenticate other types of traffic, you can configure virtual Telnet, virtual SSH, or virtual HTTP. With virtual Telnet, SSH, or HTTP, the user connects using Telnet, SSH, or HTTP to a given IP address configured on the FWSM, and the FWSM provides a prompt. For more information about the **virtual telnet**, **virtual ssh**, or **virtual http** commands, see the *Catalyst 6500 Series Switch and Cisco 7600 Series Router Firewall Services Module Command Reference*.

## Enabling Network Access Authentication

To enable network access authentication, perform the following steps:

- 
- Step 1** Using the **aaa-server** command, identify your AAA servers. If you have already identified your AAA servers, continue to the next step.
- For more information about identifying AAA servers, see the [“Identifying AAA Server Groups and Servers” section on page 14-9](#).
- Step 2** Using the **access-list** command, create an access list that identifies the source addresses and destination addresses of traffic you want to authenticate. For steps, see the [“Adding an Extended Access List” section on page 10-6](#).

The **permit** ACEs mark matching traffic for authentication, while **deny** entries exclude matching traffic from authentication. Be sure to include the destination ports for either HTTP(S), Telnet, or FTP in the access list because the user must authenticate with one of these services before other services are allowed through the FWSM.

**Step 3** To configure authentication, enter the following command:

```
hostname(config)# aaa authentication match acl_name interface_name server_group
```

where *acl\_name* is the name of the access list you created in [Step 2](#), *interface\_name* is the name of the interface as specified with the **nameif** command, and *server\_group* is the AAA server group you created in [Step 1](#).



**Note** You can alternatively use the **aaa authentication include** command (which identifies traffic within the command). However, you cannot use both methods in the same configuration. See the *Catalyst 6500 Series Switch and Cisco 7600 Series Router Firewall Services Module Command Reference* for more information.

**Step 4** (Optional) If you are using the local database for network access authentication and you want to limit the number of consecutive failed login attempts that the FWSM allows any given user account, use the **aaa local authentication attempts max-fail** command. For example:

```
hostname(config)# aaa local authentication attempts max-fail 7
```



**Tip**

To clear the lockout status of a specific user or all users, use the **clear aaa local user lockout** command.

**Step 5** (Optional) When a user authentication times out or you clear the authentication sessions using the **clear uauth** command, you can force any active connections to close immediately by entering the following command:

```
hostname(config)# aaa authentication clear-conn interface_name source_ip source_mask
```

Without this command, active connections are not terminated even though the user authentication session expired.

For example, the following commands authenticate all inside HTTP traffic and SMTP traffic:

```
hostname(config)# aaa-server AuthOutbound protocol tacacs+
hostname(config-aaa-server-group)# exit
hostname(config)# aaa-server AuthOutbound (inside) host 10.1.1.1
hostname(config-aaa-server-host)# key TACPlusUauthKey
hostname(config-aaa-server-host)# exit
hostname(config)# access-list MAIL_AUTH extended permit tcp any any eq smtp
hostname(config)# access-list MAIL_AUTH extended permit tcp any any eq www
hostname(config)# aaa authentication match MAIL_AUTH inside AuthOutbound
```

The following commands authenticate Telnet traffic from the outside interface to a particular server (209.165.201.5):

```
hostname(config)# aaa-server AuthInbound protocol tacacs+
hostname(config-aaa-server-group)# exit
hostname(config)# aaa-server AuthInbound (inside) host 10.1.1.1
hostname(config-aaa-server-host)# key TACPlusUauthKey
hostname(config-aaa-server-host)# exit
hostname(config)# access-list TELNET_AUTH extended permit tcp any host 209.165.201.5 eq telnet
```

```
hostname(config)# aaa authentication match TELNET_AUTH outside AuthInbound
```

## Configuring Custom Login Prompts

By default, when a user authenticates with the FWSM, they see the following prompt:

- For HTTP—HTTP Authentication.
- For FTP—FTP Authentication.
- For Telnet—no prompt.

You can customize the login prompt, and also show prompts when a user is accepted or rejected. If you use a RADIUS server that communicates with a Windows Active Directory server, the reject prompt can be customized to show when a user was rejected due to invalid credentials (the wrong username or password) or because a password has expired. If a password expired, the user is prompted for a new password.



### Note

Customizing the login prompt causes the FWSM to use MSCHAPv2 for the user password. Please check for MSCHAPv2 compatibility with your RADIUS server and back-end database before enabling this feature.

To customize the login prompt, perform the following steps:

**Step 1** To customize the login prompt, enter the following command:

```
hostname(config)# auth-prompt prompt text
```

Where *text* is a string of up to 235 alphanumeric characters or 31 words, limited by whichever maximum is first reached. Special characters, spaces, and punctuation characters are permitted. Entering a question mark or pressing the **Enter** key ends the string. (The question mark appears in the string.)

**Step 2** To show text when a user is accepted, enter the following command:

```
hostname(config)# auth-prompt accept text
```

**Step 3** To show text when a user is rejected, enter the following command:

```
hostname(config)# auth-prompt reject text
```

When you enter the **reject** keyword without the **invalid-credentials** or **reject expired-pwd** keywords, then this generic prompt is displayed for all rejections that are not due to invalid credentials or expired passwords. For a rejection due to an invalid credential or an expired password, then the prompt you set for the **invalid-credentials** or **reject expired-pwd** keyword displays. If you do not set any prompts for invalid credentials or expired passwords, then the generic reject prompt is shown in all cases.

**Step 4** To show text when a user is rejected due to invalid credentials, enter the following command:

```
hostname(config)# auth-prompt reject invalid-credentials text
```

**Step 5** To show text when a user is rejected due to an expired password, enter the following command:

```
hostname(config)# auth-prompt reject expired-pwd text
```

This prompt is only used if the RADIUS server uses a Windows Active Directory server for the username and password. You must configure a prompt using the **expired-pwd** keyword for the user to be prompted for a new password.

---

The following example sets the authentication prompt to the string “Please enter your username and password.”:

```
hostname(config)# auth-prompt prompt Please enter your username and password
```

After this string is added to the configuration, users see the following:

```
Please enter your username and password
User Name:
Password:
```

You can also provide separate messages to display when the FWSM accepts or rejects the authentication attempt; for example:

```
hostname(config)# auth-prompt reject Authentication failed. Try again.
hostname(config)# auth-prompt accept Authentication succeeded.
```

To set rejection messages for invalid credentials, expired password, and for unknown rejection reasons, enter the following commands:

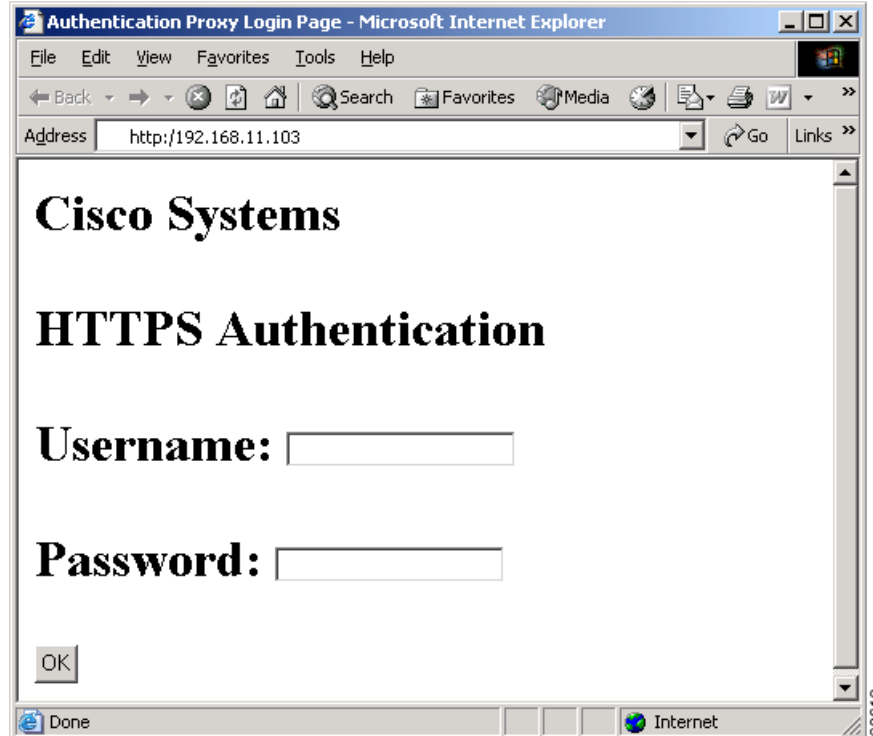
```
hostname(config)# auth-prompt reject Authentication failed. Try again.
hostname(config)# auth-prompt reject invalid-credentials Incorrect username or password
hostname(config)# auth-prompt reject expired-pwd Your password is expired. Reset your password and try again.
```

## Enabling Secure Authentication of Web Clients

The FWSM provides a method of securing HTTP authentication. Without securing HTTP authentication, usernames and passwords provided to the FWSM would be passed to the destination web server. By using the **aaa authentication secure-http-client** command, you enable the exchange of usernames and passwords between a web client and the FWSM with HTTPS. HTTPS encrypts the transmission, preventing the username and password from being passed to the external web server by HTTP.

After enabling this feature, when a user accesses a web page requiring authentication, the FWSM displays the Authentication Proxy Login Page shown in [Figure 15-1](#).

Figure 15-1 Authentication Proxy Login Page

**Note**

The Cisco Systems text field shown in this example was customized using the **auth-prompt** command. See the “[Configuring Custom Login Prompts](#)” section on page 15-5.

After the user enters a valid username and password, an “Authentication Successful” page appears and closes automatically. If the user fails to enter a valid username and password, an “Authentication Failed” page appears.

Secured web-client authentication has the following limitations:

- A maximum of 128 concurrent HTTPS authentication sessions are allowed. If all 128 HTTPS authentication processes are running, a new connection requiring authentication will not succeed.
- When **uauth timeout 0** is configured (the **uauth timeout** is set to 0), HTTPS authentication might not work. If a browser initiates multiple TCP connections to load a web page after HTTPS authentication, the first connection is let through, but the subsequent connections trigger authentication. As a result, users are continuously presented with an authentication page, even if the correct username and password are entered each time. To work around this, set the **uauth timeout** to 1 second with the **timeout uauth 0:0:1** command. However, this workaround opens a 1-second window of opportunity that might allow non-authenticated users to go through the firewall if they are coming from the same source IP address.
- Because HTTPS authentication occurs on the SSL port 443, users must not configure an **access-list** command statement to block traffic from the HTTP client to HTTP server on port 443. Furthermore, if static PAT is configured for web traffic on port 80, it must also be configured for the SSL port. In the following example, the first line configures static PAT for web traffic and the second line must be added to support the HTTPS authentication configuration.

```
static (inside,outside) tcp 10.132.16.200 www 10.130.16.10 www
```

```
static (inside,outside) tcp 10.132.16.200 443 10.130.16.10 443
```

- HTTP users see a pop-up window generated by the browser itself if **aaa authentication secure-http-client** is not configured. If **aaa authentication secure-http-client** is configured, a form loads in the browser to collect username and password. In either case, if a user enters an incorrect password, the user is prompted again. When the web server and the authentication server are on different hosts, use the **virtual http** command to get the correct authentication behavior.

To enable secure authentication of web clients, perform the following steps:

---

**Step 1** Enable HTTP authentication. For more information about enabling authentication, see the [“Enabling Network Access Authentication” section on page 15-3](#).

**Step 2** To enable secure authentication of web clients, enter this command:

```
aaa authentication secure-http-client
```



**Note**

Use of the **aaa authentication secure-http-client** command is not dependent upon enabling HTTP authentication. If you prefer, you can enter this command before you enable HTTP authentication so that if you later enable HTTP authentication, usernames and passwords are already protected by secured web-client authentication.

---

## Disabling Authentication Challenge per Protocol

You can configure whether the FWSM challenges users for a username and password. By default, the FWSM prompts the user when a AAA rule enforces authentication for traffic in a new session and the protocol of the traffic is FTP, Telnet, HTTP, or HTTPS. In some cases, you may want to disable the authentication challenge for one or more of these protocols, using the following command:

```
hostname(config)# aaa authentication protocol challenge disable
```

For example, to disable the username and password challenge for new connections using FTP, enter the following command:

```
hostname(config)# aaa authentication ftp challenge disable
```

If you disable challenge authentication for a particular protocol, traffic using that protocol is allowed only if the traffic belongs to a session previously authenticated. This authentication can be accomplished by traffic using a protocol whose authentication challenge remains enabled. For example, if you disable challenge authentication for FTP, the FWSM denies new session using FTP if the traffic is included in an authentication rule. If the user establishes the session with a protocol whose authentication challenge is enabled (such as HTTP), FTP traffic is allowed.

# Configuring Authorization for Network Access

After a user authenticates for a given connection, the FWSM can use authorization to further control traffic from the user.

This section includes the following topics:

- [Configuring TACACS+ Authorization, page 15-9](#)
- [Configuring RADIUS Authorization, page 15-10](#)

## Configuring TACACS+ Authorization

You can configure the FWSM to perform network access authorization with TACACS+.

After a user authenticates, the FWSM checks the authorization rules for matching traffic. If the traffic matches the authorization statement, the FWSM sends the username to the TACACS+ server. The TACACS+ server responds to the FWSM with information that the FWSM treats as a user-specific, dynamic access list for that traffic, based on the user profile.

**Note**

If you have used the **access-group** command to apply access lists to interfaces, be aware of the following effects of the **per-user-override** keyword on authorization by dynamic access lists:

- Without the **per-user-override** keyword, traffic for a user session must be permitted by both the interface access list and the dynamic access list.
- With the **per-user-override** keyword, the dynamic access list determines what is permitted.

For more information, see the **access-group** command entry in the *Catalyst 6500 Series Switch and Cisco 7600 Series Router Firewall Services Module Command Reference*.

Authentication and authorization statements are independent; however, any unauthenticated traffic matched by an authorization statement will be denied. For authorization to succeed, a user must first authenticate with the FWSM.

**Note**

We suggest that you identify the same traffic for authentication as for authorization. Due to the way the FWSM uses the dynamic access list, if you have a more restrictive authorization statement than authentication, then some connections are unexpectedly denied. When a user first authenticates, if the connection matches the authentication statement and not the authorization statement, then later connections for that user that match the authorization statement are denied (for as long as the uauth session exists). Conversely, if the first connection matches the authorization statement, then later connections that do not match the authorization statement but that match the authentication statement are denied. Therefore, you need to match the authentication and authorization configurations.

See the documentation for your TACACS+ server for information about configuring network access authorizations for a user.

To configure TACACS+ authorization, perform the following steps:

- Step 1** Enable authentication. For more information, see the [“Enabling Network Access Authentication” section on page 15-3](#). If you have already enabled authentication, continue to the next step.
- Step 2** To enable authorization, enter the following command:

```
hostname(config)# aaa authorization match acl_name interface_name server_group
```

where *acl\_name* is the name of the access list you created for the authentication configuration, *interface\_name* is the name of the interface as specified with the **nameif** command or by default, and *server\_group* is the AAA server group you created when you enabled authentication.

The following commands authenticate and authorize inside Telnet traffic:

```
hostname(config)# access-list TELNET_AUTH extended permit tcp any any eq telnet
hostname(config)# aaa-server AuthOutbound protocol tacacs+
hostname(config-aaa-server-group)# exit
hostname(config)# aaa-server AuthOutbound (inside) host 10.1.1.1
hostname(config-aaa-server-host)# key TACPlusUauthKey
hostname(config-aaa-server-host)# exit
hostname(config)# aaa authentication match TELNET_AUTH inside AuthOutbound
hostname(config)# aaa authorization match TELNET_AUTH inside AuthOutbound
```

## Configuring RADIUS Authorization

When authentication succeeds, the RADIUS protocol returns user authorizations in the access-accept packet sent by a RADIUS server. For more information about configuring authentication, see the “Configuring Authentication for Network Access” section on page 15-1.

When you configure the FWSM to authenticate users for network access, you are also implicitly enabling RADIUS authorizations; therefore, this section contains no information about configuring RADIUS authorization on the FWSM. It does provide information about how the FWSM handles access list information received from RADIUS servers.

You can configure a RADIUS server to download an access list to the FWSM or an access list name at the time of authentication. The user is authorized to do only what is permitted in the user-specific, dynamic access list.



### Note

If you have used the **access-group** command to apply access lists to interfaces, be aware of the following effects of the **per-user-override** keyword on authorization by dynamic access lists:

- Without the **per-user-override** keyword, traffic for a user session must be permitted by both the interface access list and the dynamic access list.
- With the **per-user-override** keyword, the dynamic access list determines what is permitted.

For more information, see the **access-group** command entry in the *Catalyst 6500 Series Switch and Cisco 7600 Series Router Firewall Services Module Command Reference*.

This section includes the following topics:

- [Configuring a RADIUS Server to Download Per-User Access Control Lists, page 15-10](#)
- [Configuring a RADIUS Server to Download Per-User Access Control List Names, page 15-12](#)

## Configuring a RADIUS Server to Download Per-User Access Control Lists

This section describes how to configure Cisco Secure ACS or a third-party RADIUS server, and includes the following topics:

- [Configuring Cisco Secure ACS for Downloadable Access Lists, page 15-11](#)
- [Configuring Any RADIUS Server for Downloadable Access Lists, page 15-12](#)

## Configuring Cisco Secure ACS for Downloadable Access Lists

You can configure downloadable access lists on Cisco Secure ACS as a shared profile component and then assign the access list to a group or to an individual user.

The access list definition consists of one or more FWSM commands that are similar to the extended **access-list** command, except without the following prefix:

```
access-list acl_name extended
```

The following example is a downloadable access list definition on Cisco Secure ACS Version 3.3:

```
+-----+
| Shared profile Components                               |
|                                                       |
|     Downloadable IP ACLs Content                     |
| Name:      acs_ten_acl                               |
|                                                       |
|     ACL Definitions                                  |
|                                                       |
| permit tcp any host 10.0.0.254                      |
| permit udp any host 10.0.0.254                      |
| permit icmp any host 10.0.0.254                    |
| permit tcp any host 10.0.0.253                     |
| permit udp any host 10.0.0.253                     |
| permit icmp any host 10.0.0.253                    |
| permit tcp any host 10.0.0.252                     |
| permit udp any host 10.0.0.252                     |
| permit icmp any host 10.0.0.252                    |
| permit ip any any                                   |
+-----+
```

For more information about creating downloadable access lists and associating them with users, see the user guide for your version of Cisco Secure ACS.

On the FWSM, the downloaded access list has the following name:

```
#ACSACL#-ip-acl_name-number
```

The *acl\_name* argument is the name that is defined on Cisco Secure ACS (*acs\_ten\_acl* in the preceding example), and *number* is a unique version ID generated by Cisco Secure ACS.

The downloaded access list on the FWSM consists of the following lines:

```
access-list #ACSACL#-ip-xxx-acs_ten_acl-3b5385f7 permit tcp any host 10.0.0.254
access-list #ACSACL#-ip-xxx-acs_ten_acl-3b5385f7 permit udp any host 10.0.0.254
access-list #ACSACL#-ip-xxx-acs_ten_acl-3b5385f7 permit icmp any host 10.0.0.254
access-list #ACSACL#-ip-xxx-acs_ten_acl-3b5385f7 permit tcp any host 10.0.0.253
access-list #ACSACL#-ip-xxx-acs_ten_acl-3b5385f7 permit udp any host 10.0.0.253
access-list #ACSACL#-ip-xxx-acs_ten_acl-3b5385f7 permit icmp any host 10.0.0.253
access-list #ACSACL#-ip-xxx-acs_ten_acl-3b5385f7 permit tcp any host 10.0.0.252
access-list #ACSACL#-ip-xxx-acs_ten_acl-3b5385f7 permit udp any host 10.0.0.252
access-list #ACSACL#-ip-xxx-acs_ten_acl-3b5385f7 permit icmp any host 10.0.0.252
access-list #ACSACL#-ip-xxx-acs_ten_acl-3b5385f7 permit ip any any
```

## Configuring Any RADIUS Server for Downloadable Access Lists

You can configure any RADIUS server that supports Cisco IOS RADIUS VSAs to send dynamic access lists to the FWSM in a Cisco IOS RADIUS cisco-av-pair VSA (VSA number 1). Cisco IOS RADIUS VSAs are identified by RADIUS vendor ID 9.

In the cisco-av-pair VSA, configure one or more ACEs that are similar to the **access-list extended** command, except that you replace the following command prefix:

```
access-list acl_name extended
```

with the following text:

```
ip:inacl#nnn=
```

The *nnn* argument is a number in the range from 0 to 999999999 that identifies the order of the command statement to be configured on the FWSM. If this parameter is omitted, the sequence value is 0, and the order of the ACEs inside the cisco-av-pair RADIUS VSA is used.

The following example is an access list definition as it should be configured for a cisco-av-pair VSA on a RADIUS server:

```
ip:inacl#1=permit tcp 10.1.0.0 255.0.0.0 10.0.0.0 255.0.0.0
ip:inacl#99=deny tcp any any
ip:inacl#2=permit udp 10.1.0.0 255.0.0.0 10.0.0.0 255.0.0.0
ip:inacl#100=deny udp any any
ip:inacl#3=permit icmp 10.1.0.0 255.0.0.0 10.0.0.0 255.0.0.0
```

For information about making unique per user the access lists that are sent in the cisco-av-pair attribute, see the documentation for your RADIUS server.

On the FWSM, the downloaded access list name has the following format:

```
AAA-user-username
```

The *username* argument is the name of the user that is being authenticated.

The downloaded access list on the FWSM consists of the following lines. Notice the order based on the numbers identified on the RADIUS server.

```
access-list AAA-user-bcham34-79AD4A08 permit tcp 10.1.0.0 255.0.0.0 10.0.0.0 255.0.0.0
access-list AAA-user-bcham34-79AD4A08 permit udp 10.1.0.0 255.0.0.0 10.0.0.0 255.0.0.0
access-list AAA-user-bcham34-79AD4A08 permit icmp 10.1.0.0 255.0.0.0 10.0.0.0 255.0.0.0
access-list AAA-user-bcham34-79AD4A08 deny tcp any any
access-list AAA-user-bcham34-79AD4A08 deny udp any any
```

Downloaded access lists have two spaces between the word “access-list” and the name. These spaces serve to differentiate a downloaded access list from a local access list. In this example, “79AD4A08” is a hash value generated by the FWSM to help determine when access list definitions have changed on the RADIUS server.

## Configuring a RADIUS Server to Download Per-User Access Control List Names

To download a name for an access list that you already created on the FWSM from the RADIUS server when a user authenticates, configure the IETF RADIUS filter-id attribute (attribute number 11) as follows:

```
filter-id=acl_name
```

**Note**

In Cisco Secure ACS, the value for filter-id attributes are specified in boxes in the HTML interface, omitting **filter-id=** and entering only *acl\_name*.

For information about making unique per user the filter-id attribute value, see the documentation for your RADIUS server.

See the [“Adding an Extended Access List” section on page 10-6](#) to create an access list on the FWSM.

## Configuring Accounting for Network Access

The FWSM can send accounting information to a RADIUS or TACACS+ server about any TCP or UDP traffic that passes through the FWSM. If that traffic is also authenticated, then the AAA server can maintain accounting information by username. If the traffic is not authenticated, the AAA server can maintain accounting information by IP address. Accounting information includes when sessions start and stop, username, the number of bytes that pass through the FWSM for the session, the service used, and the duration of each session.

To configure accounting, perform the following steps:

**Step 1** If you want the FWSM to provide accounting data per user, you must enable authentication. For more information, see the [“Enabling Network Access Authentication” section on page 15-3](#). If you want the FWSM to provide accounting data per IP address, enabling authentication is not necessary and you can continue to the next step.

**Step 2** Using the **access-list** command, create an access list that identifies the source addresses and destination addresses of traffic you want accounted. For steps, see the [“Adding an Extended Access List” section on page 10-6](#).

The **permit** ACEs mark matching traffic for authorization, while **deny** entries exclude matching traffic from authorization.

**Note**

If you have configured authentication and want accounting data for all the traffic being authenticated, you can use the same access list you created for use with the **aaa authentication match** command.

**Step 3** To enable accounting, enter the following command:

```
hostname(config)# aaa accounting match acl_name interface_name server_group
```

**Note**

Alternatively, you can use the **aaa accounting include** command (which identifies traffic within the command) but you cannot use both methods in the same configuration. See the *Catalyst 6500 Series Switch and Cisco 7600 Series Router Firewall Services Module Command Reference* for more information.

The following commands authenticate, authorize, and account for inside Telnet traffic. Telnet traffic to servers other than 209.165.201.5 can be authenticated alone, but traffic to 209.165.201.5 requires authorization and accounting.

```

hostname(config)# aaa-server AuthOutbound protocol tacacs+
hostname(config-aaa-server-group)# exit
hostname(config)# aaa-server AuthOutbound (inside) host 10.1.1.1
hostname(config-aaa-server-host)# key TACPlusUauthKey
hostname(config-aaa-server-host)# exit
hostname(config)# access-list TELNET_AUTH extended permit tcp any any eq telnet
hostname(config)# access-list SERVER_AUTH extended permit tcp any host 209.165.201.5 eq
telnet
hostname(config)# aaa authentication match TELNET_AUTH inside AuthOutbound
hostname(config)# aaa authorization match SERVER_AUTH inside AuthOutbound
hostname(config)# aaa accounting match SERVER_AUTH inside AuthOutbound

```

## Using MAC Addresses to Exempt Traffic from Authentication and Authorization

The FWSM can exempt traffic from specific MAC addresses from being authenticated or authorized. This feature is particularly useful to exempt devices such as IP phones that cannot respond to authentication prompts.



### Note

This feature exempts the list of MAC addresses for through-the-box connections only. For connections like Telnet to the FWSM, the authentication or authorization is not exempted even if the MAC address of the device is specified.

To identify MAC addresses for exemption, perform the following steps:

**Step 1** To configure a MAC list, enter the following command:

```
hostname(config)# mac-list id {deny | permit} mac macmask
```

Where the *id* argument is the hexadecimal number that you assign to the MAC list.

To exempt a MAC address, use the **permit** keyword. To allow a MAC address to be authenticated and authorized, use the **deny** keyword.

To group a set of MAC addresses, enter the **mac-list** command as many times as needed with the same ID value. Because you can only use one MAC list for AAA exemption, be sure that your MAC list includes all the MAC addresses you want to exempt. You can create multiple MAC lists, but you can only use one at a time.

The order of entries matters, because the packet uses the first entry it matches, as opposed to a best match scenario. If you have a **permit** entry, and you want to deny an address that is allowed by the **permit** entry, be sure to enter the **deny** entry before the **permit** entry.

The *mac* argument specifies the source MAC address in 12-digit hexadecimal form; that is, nnnn.nnnn.nnnn.

The *macmask* argument specifies the portion of the MAC address that should be used for matching. For example, ffff.ffff.ffff matches the MAC address exactly. ffff.ffff.0000 matches only the first 8 digits.

**Step 2** To exempt traffic for the MAC addresses specified in a particular MAC list, enter the following command:

```
hostname(config)# aaa mac-exempt match id
```

Where *id* is the string identifying the MAC list containing the MAC addresses whose traffic is to be exempt from authentication and authorization. You can only enter one instance of the **aaa mac-exempt** command.

---

The following example bypasses authentication for a single MAC address:

```
hostname(config)# mac-list abc permit 00a0.c95d.0282 ffff.ffff.ffff
hostname(config)# aaa mac-exempt match abc
```

The following entry bypasses authentication for all Cisco IP Phones, which have the hardware ID 0003.E3:

```
hostname(config)# mac-list acd permit 0003.E300.0000 FFFF.FF00.0000
hostname(config)# aaa mac-exempt match acd
```

The following example bypasses authentication for a group of MAC addresses except for 00a0.c95d.02b2. Enter the **deny** statement before the **permit** statement, because 00a0.c95d.02b2 matches the **permit** statement as well, and if it is first, the **deny** statement will never be matched.

```
hostname(config)# mac-list 1 deny 00a0.c95d.0282 ffff.ffff.ffff
hostname(config)# mac-list 1 permit 00a0.c95d.0000 ffff.ffff.0000
hostname(config)# aaa mac-exempt match 1
```

