Chapter 7

Authentication, Authorization, and Accounting (AAA) Services

This chapter covers the following topics:

- AAA protocols and services supported by the Cisco ASA
- Defining an authentication server
- Authenticating administrative sessions
- Configuring authorization
- Configuring downloadable ACLs
- Configuring accounting
- Troubleshooting AAA

This chapter provides a detailed explanation of the configuration and troubleshooting of authentication, authorization, and accounting (AAA) network security services that Cisco ASA supports. AAA offers different solutions that provide access control to network devices. The following services are included within its modular architectural framework:

- **Authentication:** The process of validating users based on their identity and predetermined credentials, such as passwords and other mechanisms like digital certificates.
- **Authorization:** The method by which a network device assembles a set of attributes that regulates what tasks the user is authorized to perform. These attributes are measured against a user database. The results are returned to the network device to determine the user's qualifications and restrictions. This database can be located locally on Cisco ASA or it can be hosted on a RADIUS or Terminal Access Controller Access Control System Plus (TACACS+) server.
- **Accounting:** The process of gathering and sending user information to an AAA server used to track login times (when the user logged in and logged out) and the services that users access. This information can be utilized for billing, auditing, and reporting purposes.

AAA Protocols and Services Supported by Cisco ASA

Cisco ASA can be configured to maintain a local user database or for authentication on an external server. The following are the AAA authentication underlying protocols and servers that are supported as external database repositories:

- RADIUS
- TACACS+
- RSA SecurID (SDI)
- Microsoft Windows server operating systems that support NTLM Version 1 (often referred to as "Windows NT" authentication)
- Kerberos
- Lightweight Directory Access Protocol (LDAP)

Table 7-1 shows the different methods and the functionality that each protocol supports.

Table 7-1 *AAA Support Matrix*

Method	Authentication	Authorization	Accounting
Internal server	Yes	Yes	No
RADIUS	Yes	Yes	Yes
TACACS+	Yes	Yes	Yes
SDI	Yes	No	No
Windows NTLM	Yes	No	No
Kerberos	Yes	No	No
LDAP	No	Yes	No

Using an external authentication server in medium and large deployments is recommended for better scalability and easier management.

Cisco ASA supports the authentication methods listed in Table 7-1 with the following services:

- Remote-access virtual private network (VPN) user authentication
- Administrative session authentication
- Firewall session authentication (cut-through proxy)

Table 7-2 outlines the support for the authentication methods in correlation to the specific services.

Table 7-2 *Authentication Support for Each Service*

Service	Local	RADIUS	TACACS+	SDI	Windows NTLM	Kerberos	LDAP
Remote-access VPN	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Administrative sessions	Yes	Yes	Yes	Yes (version 8.2(1) or later)	Yes	Yes	Yes
Firewall sessions	Yes	Yes	Yes	Yes	Yes	Yes	Yes

As previously mentioned, the authorization mechanism assembles a set of attributes that describes what the user is allowed to do within the network or service. Cisco ASA supports local and external authorization, depending on the service used. Table 7-3 shows the authorization support matrix.

Table 7-3 *Authorization Support for Each Service*

Service	Local	RADIUS	TACACS+	SDI	Windows NTLM	Kerberos	LDAP
Remote-access VPN	Yes	Yes	No	No	No	No	Yes
Administrative sessions	Yes	No	Yes	No	No	No	No
Firewall sessions	No	Yes (with user-specific access control lists only)	Yes	No	No	No	No

Note Local authorization for administrative sessions can be used only for command authorization.

Accounting is supported by RADIUS and TACACS+ servers only.

The following subsections introduce each of the authentication protocols and servers that Cisco ASA supports.

RADIUS

RADIUS is a widely implemented authentication standard protocol that is defined in RFC 2865, "Remote Authentication Dial In User Service (RADIUS)." RADIUS operates in a client/server model. A RADIUS client is usually referred to as a network access server (NAS). A NAS is responsible for passing user information to the RADIUS server. Cisco ASA acts as a NAS and authenticates users based on the RADIUS server's response.

Cisco ASA supports several RADIUS servers that are RFC-compliant, such as Cisco Secure Access Control Server (ACS), Cisco Identity Services Engine (ISE), and RSA RADIUS, among others. Support and testing with other servers is a continuous effort between vendors.

The RADIUS server receives user authentication requests and subsequently returns configuration information required for the client (in this case, the Cisco ASA) to support the specific service to the user. The RADIUS server accomplishes this by sending Internet Engineering Task Force (IETF) or vendor-specific attributes. (RADIUS authentication attributes are defined in RFC 2865.) Figure 7-1 illustrates how this process works.

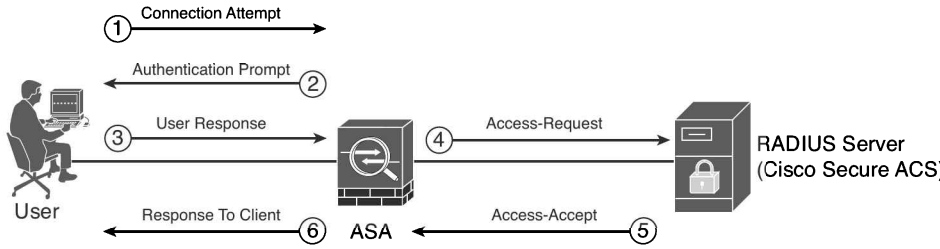

Figure 7-1 *RADIUS Authentication Process*

In Figure 7-1, a Cisco ASA acts as a NAS and the RADIUS server is a Cisco ISE server. The sequence of events is as follows:

1. A user attempts to connect to the Cisco ASA (via administrative session, remote-access VPN, or cut-through proxy authentication).
2. The Cisco ASA prompts the user, requesting a username and password.
3. The user sends his or her credentials to the Cisco ASA.
4. The Cisco ASA sends the authentication request (Access-Request) to the RADIUS server.
5. The RADIUS server sends an Access-Accept message (if the user is successfully authenticated) or an Access-Reject (if the user is not successfully authenticated).
6. The Cisco ASA responds to the user and allows access to the specific service.

The RADIUS server can also send IETF or vendor-specific attributes to the Cisco ASA, depending on the implementation and services used. These attributes can contain information such as an IP address to assign the client and authorization information. RADIUS

servers combine authentication and authorization phases into a single request-and-response communication cycle. The Cisco ASA authenticates itself to the RADIUS server by using a preconfigured shared secret. For security reasons, this shared secret is never sent over the network.

Note Passwords are sent as encrypted messages from the Cisco ASA to the RADIUS server. This is useful to protect this critical information from an intruder. The Cisco ASA hashes the password, using the shared secret that is defined on the Cisco ASA and the RADIUS server.

The RADIUS servers can also proxy authentication requests to other RADIUS servers or other types of authentication servers. Figure 7-2 illustrates this methodology.

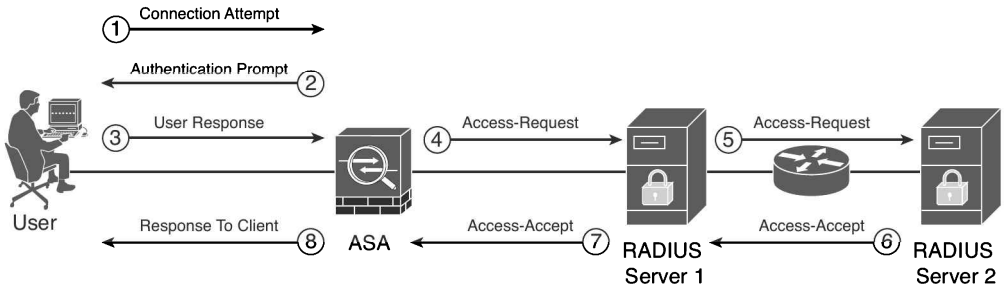

Figure 7-2 *RADIUS Server Acting as Proxy to Other Authentication Servers*

In Figure 7-2, RADIUS Server 1 acts as a proxy to RADIUS Server 2. It sends the authentication request from the Cisco ASA to RADIUS Server 2 and proxies the response back to the ASA.

TACACS+

TACACS+ is an AAA security protocol that provides centralized validation of users who are attempting to gain access to NASs. The TACACS+ protocol offers support for separate and modular AAA facilities. The TACACS+ protocol's primary goal is to supply complete AAA support for managing multiple network devices.

TACACS+ uses port 49 for communication and allows vendors to use either User Datagram Protocol (UDP) or TCP encoding. Cisco ASA uses the TCP version for its TACACS+ implementation.

The TACACS+ authentication concept is similar to RADIUS. The NAS sends an authentication request to the TACACS+ server (daemon). The server ultimately sends any of the following messages back to the NAS:

- ACCEPT: User has been successfully authenticated and the requested service is allowed. If authorization is required, the authorization process begins at this point.

- REJECT: User authentication is denied. The user may be prompted to retry authentication, depending on the TACACS+ server and NAS.
- ERROR: A certain error occurred during authentication. This can be experienced because of network connectivity problems or a configuration error.
- CONTINUE: User is prompted to provide further authentication information.

After the authentication process is complete, if authorization is required, the TACACS+ server proceeds with the authorization phase. The user must first successfully be authenticated before proceeding to TACACS+ authorization.

RSA SecurID

RSA SecurID (SDI) is a solution provided by RSA (now owned by EMC). The RSA Authentication Manager is the administrative component of the SDI solution. It enables the use of one-time passwords (OTP). Cisco ASA supports SDI authentication natively only for VPN user authentication. However, if the Cisco ASA is utilizing an authentication server, such as Cisco ISE for Windows NT, the server can use external authentication to an SDI server and proxy the authentication request for all other services supported by the Cisco ASA. The Cisco ASA and SDI use UDP port 5500 for communication.

The SDI solution employs small physical devices called *tokens* that provide users with an OTP that changes every 60 seconds. These OTPs are generated when a user enters a personal identification number (PIN) and are synchronized with the server to provide the authentication service. The SDI server can be configured to require the user to enter a new PIN when trying to authenticate. This process is called New PIN mode, which Cisco ASA supports. Figure 7-3 demonstrates how this solution works when a user attempts to connect to the Cisco ASA using the Cisco AnyConnect Secure Mobility client.

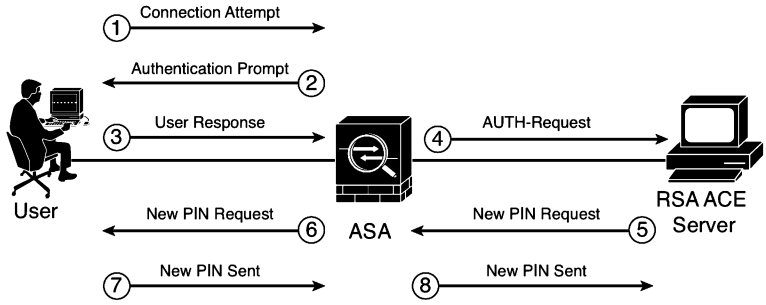

Figure 7-3 *SDI Authentication Using New PIN Mode*

The purpose of New PIN mode is to allow the user to change her PIN for authentication. The following sequence of events occurs when using SDI authentication with the New PIN mode feature, as shown in Figure 7-3:

1. The user attempts to establish a remote-access VPN connection with the Cisco AnyConnect Secure Mobility client employing SSL, and the SSL tunnel negotiation takes place.

2. The Cisco ASA prompts the user for authentication.
3. The user provides a username and passcode.
4. The Cisco ASA forwards the authentication request to the SDI server.
5. If New PIN mode is enabled, the SDI server authenticates the user and requests a new PIN to be used during the next authentication session for that user.
6. The Cisco ASA prompts the user for a new PIN.
7. The user enters a new PIN.
8. The Cisco ASA sends the new PIN information to the SDI server.

Note You can find more information about the RSA SDI server at http://www.emc.com/security/rsa-securid/rsa-authentication-manager.htm.

Microsoft Windows NTLM

Cisco ASA supports Windows NTLM native authentication only for VPN remote-access connections. It communicates with a Windows NTLM server via TCP port 139. Similarly to SDI, you can use a RADIUS/TACACS+ server, such as Cisco ISE and Cisco ACS, to proxy authentication to Windows NT for other services supported by Cisco ASA.

Active Directory and Kerberos

Cisco ASA can authenticate VPN users via an external Windows Active Directory, which uses Kerberos for authentication. Kerberos is an authentication protocol created by the Massachusetts Institute of Technology (MIT) that provides mutual authentication used by many vendors and applications. It can also communicate with a UNIX/Linux-based Kerberos server. Cisco ASA communicates with the Active Directory and/or a Kerberos server via UDP port 88.

Lightweight Directory Access Protocol

Cisco ASA supports LDAP authorization for remote-access VPN connections. The LDAP protocol is defined in RFC 4510, "Lightweight Directory Access Protocol (LDAP): Technical Specification Road Map," and RFC 4511, "Lightweight Directory Access Protocol (LDAP): The Protocol." LDAP provides authorization services when given access to a user database within a Directory Information Tree (DIT). This tree contains entities called *entries*, which consist of one or more attribute values called *distinguished names* (DN). The DN values must be unique within the DIT.

The Cisco ASA supports single sign-on (SSO) authentication of WebVPN users, employing the HTTP Form protocol. The SSO feature is designed to allow WebVPN users to

enter a username and password only once while accessing WebVPN services and any web servers behind the Cisco ASA. The Cisco ASA acts as a proxy for the user to the authenticating server. The Cisco ASA keeps a cookie and utilizes it to authenticate the user to any other protected web servers.

Defining an Authentication Server

Before configuring an authentication server on Cisco ASA, you must specify AAA server groups. A server group defines the attributes of one or more AAA servers. This information includes the AAA protocol used, IP address of the AAA servers, and other related information. Complete the following steps to by using Cisco ASDM:

1. Log in to ASDM and navigate to **Configuration > Device Management > Users/AAA > AAA Server Groups**.
2. By default, the LOCAL Server group is present in the configuration. To add an AAA server group, click **Add**.
3. In the Add AAA Server Group dialog box, enter a server group name in the AAA Server Group field, as illustrated in Figure 7-4. The AAA server group name used in this example is my-radius-group.

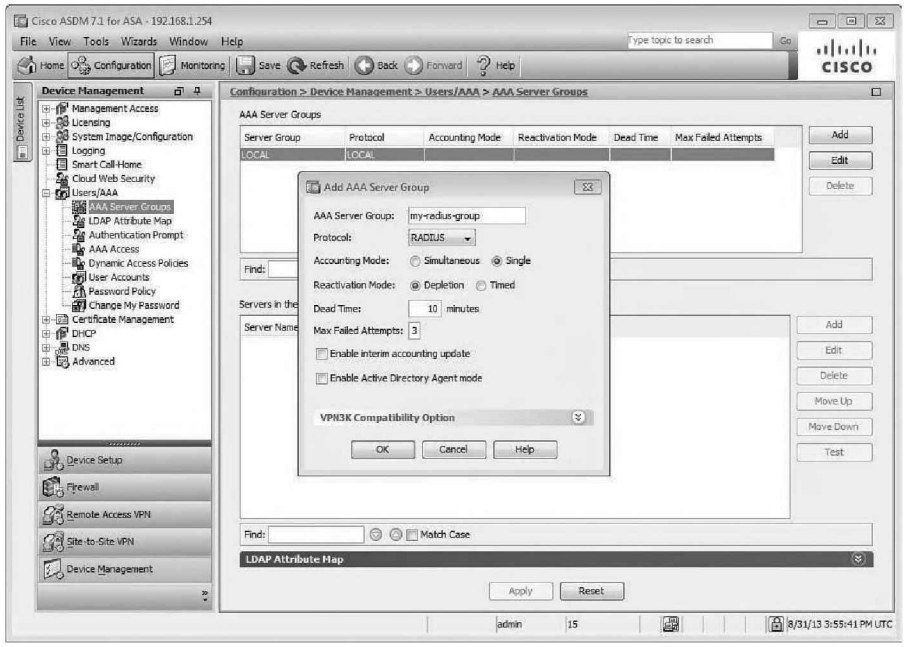

Figure 7-4 *Add AAA Server Group Dialog Box*

4. From the Protocol drop-down list, choose the AAA protocol to be used. RADIUS is used in this example; however, you can choose from any of the following server types:
   - RADIUS
   - TACACS+
   - SDI
   - NT Domain
   - Kerberos
   - LDAP
   - HTTP Form
5. Several of the parameters in this dialog box vary depending on the authentication protocol that is used. In this example, all the other fields are left with default values. The Accounting Mode field has two options: Simultaneous and Single. When single mode is selected, the Cisco ASA sends accounting data to only one accounting server. To send accounting data to all servers in the group, select **Simultaneous**.
6. Depletion is selected in the Reactivation Mode field. The reactivation mode is used to control the behavior when AAA servers fail. When depletion mode is selected in the Cisco ASA, failed servers are reactivated only after all the servers in the group are inactive. If this option is selected, you must add a time interval in the Dead Time field. In this example, the default value is configured (10 minutes).

   Alternatively, you can select Timed mode, where failed servers are reactivated after 30 seconds of down time.
7. The Max Failed Attempts field is used to limit the maximum number of failed authentication attempts. The default is 3 attempts.
8. Click **OK**.
9. Click **Apply** to apply the configuration changes.
10. Click **Save** to save the configuration in the Cisco ASA.

Complete the following steps to add the AAA server to the AAA server group that was previously configured:

1. Log in to ASDM and navigate to **Configuration > Device Management > Users/ AAA > AAA Server Groups**.
2. Click **Add** in the Servers in the Selected Group area (with **my-radius-group** selected in the AAA Server Groups area). The Add AAA Server dialog box shown in Figure 7-5 is displayed.
3. As you see in Figure 7-5, my-radius-group is already entered in the Server Group field. From the Interface Name drop-down list, choose the interface where the RADIUS server resides. In this example, the RADIUS server is reachable through the management interface.

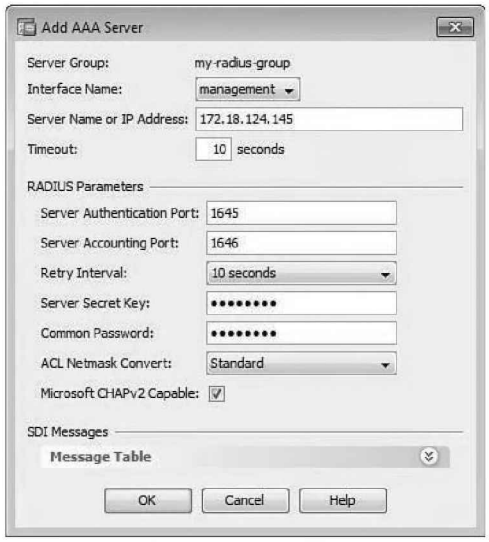

Figure 7-5 *Add AAA Server Dialog Box*

4. In the Server Name or IP Address field, enter the AAA server name or IP address. In Figure 7-5, the RADIUS server's IP address is 172.18.124.145 and the ASA has connectivity with this server using the management interface.
5. In the Timeout field, specify the amount of time (in seconds) that the Cisco ASA waits before timing out the authentication session. The default value of 10 seconds is used in this case.
6. In the Server Authentication Port field, you can specify the port used by the Cisco ASA to communicate to the RADIUS server for authentication purposes. In Figure 7-5, the default RADIUS authentication port 1645 is entered.
7. Similarly, in the Server Accounting Port field, you can specify the port used by the Cisco ASA to communicate to the RADIUS server for accounting. In this instance, the default RADIUS accounting port 1646 is entered.
8. From the Retry Interval drop-down list, specify the amount of time the Cisco ASA waits to retry an authentication attempt, in case the RADIUS server does not respond. The default value of 10 seconds is used.
9. In the Server Secret Key field, enter the secret key used by the Cisco ASA and the RADIUS server to authenticate each other. This can be a string of up to 64 characters.
10. In the Common Password field, enter a case-sensitive password that is common among users who access this RADIUS authorization server via the Cisco ASA. If you do not use a common password, the user's username is utilized as the password when accessing the RADIUS authorization server.
11. (Optional) Specify how the Cisco ASA handles netmasks received in downloadable ACLs (covered later in this chapter) by choosing any of the following from the ACL Netmask Convert drop-down list:

- **Detect automatically:** The Cisco ASA automatically detects a wildcard netmask expression and converts it to a standard netmask.
- **Standard:** The Cisco ASA honors the netmask received from the RADIUS server and does not perform any translation from wildcard netmask expressions.
- **Wildcard:** The Cisco ASA converts all netmasks to standard netmask expressions.

The default value (Standard) is used in this example.

12. Click **OK**.
13. Click **Apply** to apply the configuration changes.
14. Click **Save** to save the configuration in the Cisco ASA.

If you are using the command-line interface (CLI) to configure the Cisco ASA, specify AAA server groups with the **aaa-server** command. The syntax of the **aaa-server** command to specify a new AAA server group and the respective protocol is as follows:

```
aaa-server server-tag protocol server-protocol
```

The *server-tag* variable is the server group name that is referenced by the other AAA command, and *server-protocol* is the name of the supported AAA protocol. Example 7-1 shows the different authentication protocols that can be defined within an AAA server group.

Example 7-1 *AAA Server Group Authentication Protocols*

```
New York(config)# aaa-server my-radius-group protocol ?
  kerberos  Protocol Kerberos
  ldap      Protocol LDAP
  nt        Protocol NT
  radius    Protocol RADIUS
  sdi       Protocol SDI
  tacacs+   Protocol TACACS+
```

In Example 7-1, the AAA server group tag is named my-radius-group. After defining the AAA server group with the respective authentication protocol, you are shown the (config-aaa-server) prompt. Example 7-2 shows the commands that are used to accomplish the same tasks that were previously demonstrated for ASDM.

Example 7-2 *Configuring the AAA Server Using the CLI*

```
NewYork(config)# aaa-server my-radius-group protocol radius
NewYork(config-aaa-server-group)# aaa-server my-radius-group (management) host
172.18.124.145
NewYork(config-aaa-server-host)#  key myprivatekey
NewYork(config-aaa-server-host)#  radius-common-pw mycommonpassword
```

In Example 7-2, the AAA server group my-radius-group is defined to process authentication requests using the RADIUS protocol. In the second line the RADIUS server (172.18.124.145) is defined, as well as the interface (management) where the RADIUS server resides. The key used for authentication is myprivatekey. The RADIUS common password is set to mycommonpassword.

Note Accounting mode options are available only if you are configuring an AAA server group for RADIUS or TACACS+.

You can also use the **max-failed-attempts** subcommand, which specifies the maximum allowed number of communication failures for any server in the AAA server group before that server is disabled or deactivated. The maximum number of failures can be configured in a range from 1 to 5.

To review, the Cisco ASA supports two different AAA server reactivation policies or modes:

- **Timed mode:** The failed or deactivated servers are reactivated after 30 seconds of downtime.
- **Depletion mode:** The failed or deactivated servers remain inactive until all other servers within the configured group are inactive.

To view statistics about all AAA servers defined for a specific protocol, use the following command:

```
show aaa-server protocol server-protocol
```

Example 7-3 includes the output of this command for the RADIUS protocol.

Example 7-3 *Output of the* **show aaa-server protocol** *Command*

```
New-York# show aaa-server protocol radius
Server Group:    mygroup
Server Protocol: radius
Server Address:  172.18.124.145
Server port:     1645(authentication), 1646(accounting)
Server status:   ACTIVE, Last transaction at unknown
Number of pending requests              0
Average round trip time                 0ms
Number of authentication requests       55
Number of authorization requests        13
Number of accounting requests           45
Number of retransmissions               0
Number of accepts                       54
Number of rejects                       1
Number of challenges                    54
```

```
Number of malformed responses          0
Number of bad authenticators           0
Number of timeouts                     0
Number of unrecognized responses       0
```

Several counters can be helpful when troubleshooting AAA-related problems. For instance, you can compare the number of authentication requests versus the number of authentication rejects and accepts. Additionally, pay attention to any malformed authentication requests, unrecognized responses, or timeouts to determine whether there is a communication problem with the AAA server.

To show the configuration of a specific AAA server, use the following command:

```
show running-config aaa-server [server-group [(if_name) host ip_address]]
```

To show statistics about a specific AAA server, use the following command:

```
show aaa-server [server-tag [host hostname]]
```

Example 7-4 includes the output of this command for server 172.18.124.145.

Example 7-4 *Output of the* **show aaa-server** *Command for a Specific Host*

```
NewYork# show aaa-server my-radius-group host 172.18.124.145
Server Group:    my-radius-group
Server Protocol: radius
Server Address:  172.18.124.145
Server port:     1645(authentication), 1646(accounting)
Server status:   ACTIVE, Last transaction at unknown
Number of pending requests             0
Average round trip time                0ms
Number of authentication requests      55
Number of authorization requests       13
Number of accounting requests          45
Number of retransmissions              0
Number of accepts                      54
Number of rejects                      1
Number of challenges                   54
Number of malformed responses          0
Number of bad authenticators           0
Number of timeouts                     0
Number of unrecognized responses       0
```

To clear the AAA server statistics for a specific server, use this command:

```
clear aaa-server statistics [tag [host hostname]]
```

To clear the AAA server statistics for all servers providing services for a specific protocol, use this command:

```
clear aaa-server statistics protocol server-protocol
```

To erase a specific AAA server group from the configuration, use this command:

```
clear configure aaa-server [server-tag]
```

Configuring Authentication of Administrative Sessions

Cisco ASA supports authentication of administrative sessions by using a local user database, a RADIUS server, or a TACACS+ server. An administrator can connect to the Cisco ASA via any of the following:

- Telnet
- Secure Shell (SSH)
- Serial console connection
- Cisco ASDM

If connecting via Telnet or SSH, the user can retry authentication three times in case of user error. After the third time, the authentication session and connection to the Cisco ASA are closed. Authentication sessions via the console prompt the user continuously until the user enters the correct username and password.

Before you start the configuration, you must decide which user database you will employ (local or external AAA server). If you are using an external AAA server, configure the AAA server group and host as covered in the previous section. Use the **aaa authentication** command to require authentication verification when accessing Cisco ASA for administration. The following sections teach you how to configure external authentication for each type of connection.

Authenticating Telnet Connections

You can enable Telnet access to the Cisco ASA to any internal interface or to the outside if an IPsec connection is established. To configure authentication for Telnet connections to the Cisco ASA using ASDM, complete the following steps:

1. Log in to ASDM and navigate to **Configuration > Device Management > Users/AAA > AAA Access > Authentication**, as shown in Figure 7-6.

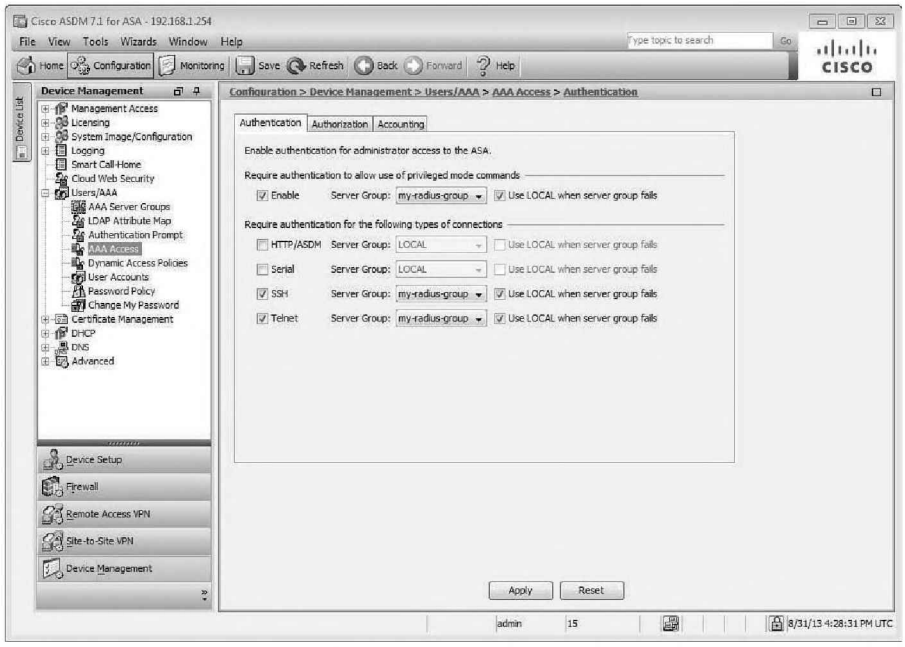

Figure 7-6 *Using ASDM to Configure Authentication for Telnet Connections*

2. Check the **Telnet** check box in the Require Authentication for the Following Types of Connections area.
3. In this case, the RADIUS server previously configured in the AAA server group is used for authentication. Choose the server group **my-radius-group** from the Server Group drop-down list to the right of the Telnet check box.
4. If you would like to fall back to the local user database in case the RADIUS server fails, check the **Use LOCAL When Server Group Fails** check box, as shown in Figure 7-6.
5. Click **OK**.
6. Click **Apply** to apply the configuration changes.
7. Click **Save** to save the configuration in the Cisco ASA.

You can also authenticate any users before they employ the **enable** command via the CLI. To accomplish this task, complete the following steps:

1. Log in to ASDM and navigate to **Configuration > Device Management > Users/AAA > AAA Access > Authentication**.
2. Check the **Enable** check box in the Require Authentication to Allow Use of Privilege Mode Commands area, as shown in Figure 7-6.

3. In Figure 7-6, the RADIUS server is used for authentication. Choose the server group **my-radius-group** from the Server Group drop-down list to the right of the Enable check box.
4. To allow the Cisco ASA to use the local database as a fallback method, check the **Use LOCAL When Server Group Fails** check box.
5. Click **OK**.
6. Click **Apply** to apply the configuration changes.
7. Click **Save** to save the configuration in the Cisco ASA.

Example 7-5 shows the CLI commands sent by ASDM to the Cisco ASA.

Example 7-5 *Using the CLI to Configure Authentication for Telnet Connections*

```
aaa authentication enable console my-radius-group LOCAL
aaa authentication telnet console my-radius-group LOCAL
telnet 0.0.0.0 0.0.0.0 inside
```

In Example 7-5, the **aaa authentication enable console** command is set to require authentication before any user can enter into enable mode. The **my-radius-group** AAA server group name is applied to this command. The keyword **LOCAL** is used to enable fallback to the local database if the configured authentication server is unavailable.

The second line in Example 7-5 enables authentication for Telnet connections by using the **my-radius-group** AAA server group, as well as the **LOCAL** keyword to enable fallback to the local database.

Tip Do not confuse the keyword **console** with the serial console on the Cisco ASA. This keyword is used to force the Cisco ASA to require AAA authentication for any client trying to connect to it via Telnet, serial console, HTTP, or SSH. Telnet is used in Example 7-5.

Authenticating SSH Connections

The steps for using ASDM to configure authentication for SSH administrative sessions to the Cisco ASA are very similar to the steps discussed in the previous section. Complete the following steps to configure authentication for SSH connections to the Cisco ASA:

1. Log in to ASDM and navigate to **Configuration > Device Management > Users/AAA > AAA Access > Authentication**. The same window illustrated in Figure 7-6 is displayed.
2. Check the **SSH** check box in the Require Authentication for the Following Types of Connections area.

3. In Figure 7-6, the RADIUS server previously configured in the AAA server group (my-radius-group) is used for authentication.
4. If you would like to fall back to the local user database in case the RADIUS server fails, check the **Use LOCAL When Server Group Fails** check box to the right of the SSH check box, as shown in Figure 7-6.
5. Click **OK**.
6. Click **Apply** to apply the configuration changes.
7. Click **Save** to save the configuration in the Cisco ASA.

To enable SSH on Cisco ASA via the CLI, you first configure a hostname and domain name before generating the RSA key pair used by SSH. Example 7-6 shows how to generate the RSA key pair and enable SSH version 2 connections from any systems on the inside interface.

Example 7-6 *Generating RSA Key Pair and Enabling SSH Version 2*

```
asa# configure terminal
asa (config)# hostname NewYork
New-York(config)# domain-name cisco.com
New-York(config)# crypto key generate rsa modulus 2048
INFO: The name for the keys will be: ASA.cisco.com
Keypair generation process begin.
New-York(config)# ssh 0.0.0.0 0.0.0.0 inside
New-York(config)# ssh version 2
```

After the RSA key pair has been generated and SSH has been enabled, complete your AAA server group and host configuration. The **my-radius-group** AAA server group is used in the **aaa authentication ssh console** command to enable SSH authentication, as shown in Example 7-7.

Example 7-7 *Configuring SSH Authentication to a TACACS+ Server*

```
New-York(config)# aaa authentication ssh console my-radius-group LOCAL
```

The **LOCAL** keyword is used in Example 7-7 to enable fallback to the local database. Make sure to issue the **write memory** command to save the configuration after the RSA key pair is generated.

Authenticating Serial Console Connections

Complete the following steps to configure authentication for serial console connections to the Cisco ASA, using ASDM:

1. Log in to ASDM and navigate to **Configuration > Device Management > Users/ AAA > AAA Access > Authentication.**

2. Check the **Serial** check box in the Require Authentication for the Following Types of Connections area.
3. In Example 7-7, the RADIUS server previously configured in the AAA server group (my-radius-group) is used for authentication.
4. If you would like to fall back to the local user database in case the RADIUS server fails, check the **Use LOCAL When Server Group Fails** check box.
5. Click **OK**.
6. Click **Apply** to apply the configuration changes.
7. Click **Save** to save the configuration in the Cisco ASA.

To configure authentication of serial console connections, use the **aaa authentication serial console** command. Be aware that you can get locked out of the Cisco ASA easily with any misconfiguration. Example 7-8 demonstrates how to configure serial console authentication, using the AAA server group previously configured.

Example 7-8 *Configuring Serial Console Authentication*

```
New-York(config)# aaa authentication serial console my-radius-group LOCAL
```

Note Establishing two separate sessions to the Cisco ASA is always recommended when configuring AAA authentication. The purpose of this procedure is to avoid getting locked out of the CLI. Open one session using a Telnet or SSH connection and connect to the serial console of the Cisco ASA. One of the sessions can be disconnected after the configuration is verified and tested. If the administrator is locked out of the security appliance, follow the password recovery procedure discussed in Chapter 5, "System Maintenance."

Authenticating Cisco ASDM Connections

Complete the following steps to configure authentication for ASDM administrative connections to the Cisco ASA using ASDM:

1. Log in to ASDM and navigate to **Configuration > Device Management > Users/ AAA > AAA Access > Authentication**.
2. Check the **HTTP/ASDM** check box in the Require Authentication for the Following Types of Connections area.
3. In Example 7-8, the RADIUS server previously configured in the AAA server group (my-radius-group) is used for authentication.
4. If you would like to fall back to the local user database in case the RADIUS server fails, check the **Use LOCAL When Server Group Fails** check box.

5. Click **OK**.
6. Click **Apply** to apply the configuration changes.
7. Click **Save** to save the configuration in the Cisco ASA.

Alternatively, you can configure the **aaa authentication http console** CLI command to require authentication for Cisco ASDM users. Example 7-9 demonstrates how to configure ASDM authentication, using the AAA server group previously configured.

Example 7-9 *Configuring HTTP Authentication for ASDM Users*

```
New-York(config)# aaa authentication http console my-radius-group LOCAL
```

If this command is not configured, Cisco ASDM users can gain access to the Cisco ASA by entering only the enable password, and no username, at the authentication prompt.

Authenticating Firewall Sessions (Cut-Through Proxy Feature)

The Cisco ASA firewall session authentication is similar to the former "cut-through proxy" feature on the legacy Cisco Secure PIX Firewall. The firewall cut-through proxy requires the user to authenticate before passing any traffic through the Cisco ASA. A common deployment is to authenticate users before accessing a web server behind the Cisco ASA. Figure 7-7 illustrates how firewall session authentication works.

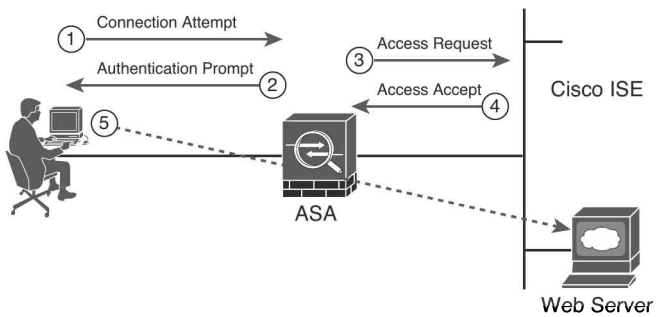

Figure 7-7 *Cut-Through Proxy Feature Example*

The following are the steps represented in Figure 7-7:

1. The user on the outside of the Cisco ASA attempts to create an HTTP connection to the web server behind the ASA.
2. The Cisco ASA intercepts this connection attempt and prompts the user for authentication.
3. The Cisco ASA receives the authentication information from the user and sends an AUTH Request to the Cisco Identity Services Engine (ISE) server.

4. The server authenticates the user and sends an AUTH Accept message to the Cisco ASA.
5. The Cisco ASA allows the user to access the web server and redirects the user's browser to the original destination.

Complete the following steps to enable network access authentication via the cut-through proxy feature, using ASDM:

1. Log in to ASDM and navigate to **Configuration > Firewall > AAA Rules.**
2. Click **Add** and choose **Add Authentication Rule.** The dialog box illustrated in Figure 7-8 is displayed.

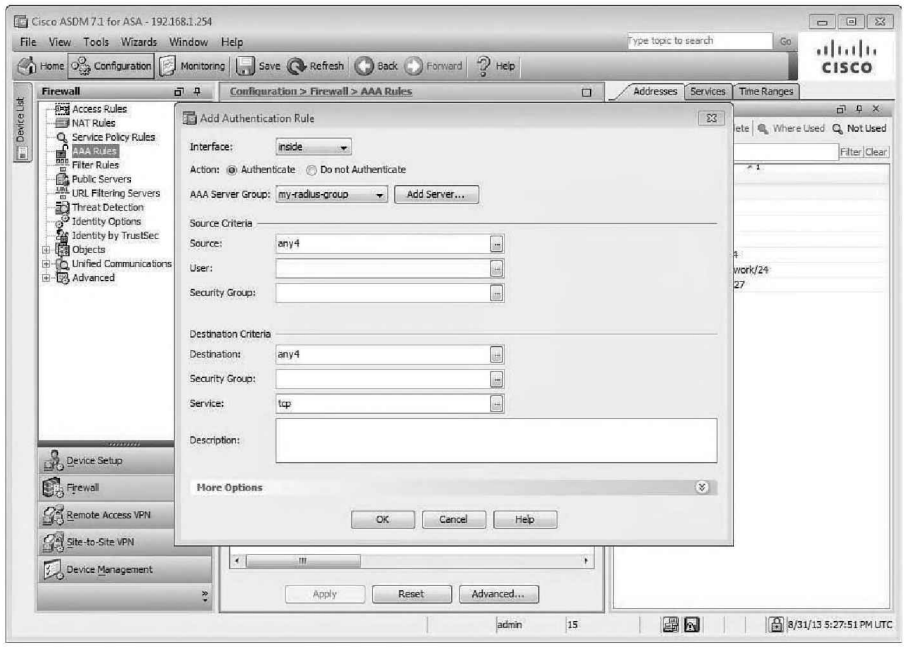

Figure 7-8 *Adding an Authentication Rule via ASDM*

3. From the Interface drop-down list, choose the interface where the authentication rule will be applied. The inside interface is selected in Figure 7-8.
4. Click **Authenticate** in the Action field to require user authentication.
5. Choose the AAA server group **my-radius-group** from the AAA Server Group drop-down list.

Note You can add an AAA server to the server group by clicking the Add Server button. In Figure 7-8, the preconfigured AAA server is used.

6. You must specify a source and a destination for traffic that requires authentication. Enter the source IP address, network address, or the **any** keyword in the Source field. Alternatively, you can click the ellipsis button (...) to select an address that has already been configured in ASDM. In this case, the **any4** keyword is entered to require authentication for any IPv4 source from the inside interface.
7. Enter the destination IP address, network address, or the **any** keyword in the Destination field. Alternatively, you can click the ellipsis button (...) to select an address that has already been configured in ASDM. In Figure 7-8, the **any4** keyword is entered to require authentication when a host tries to reach any IPv4 destination.
8. Enter an IP service name for the destination service in the Service field. Alternatively, click the ellipsis button (...) to open a separate dialog box where you can select from a list of available services. In Figure 7-8, authentication is required for any host trying to access any TCP-based applications.
9. (Optional) Enter a description for the authentication rule in the Description field.

Note Click More Options if you want to specify a source service for TCP or UDP applications or set a time range within which this rule is to be applied.

10. Click **OK**.
11. Click **Apply** to apply the configuration changes.
12. Click **Save** to save the configuration in the Cisco ASA.

Cut-through proxy can also be enabled with the **aaa authentication match** CLI command. It enables you to configure an ACL to classify what traffic is authenticated. Using the **aaa authentication match** command replaces the use of the **include** and **exclude** options and is now the preferred method to configure authentication through the Cisco ASA. The following is the command syntax:

```
aaa authentication match acl interface server-tag
```

The *acl* keyword refers to the name or number of the ACL configured to define what traffic is authenticated. The *interface* keyword defines the interface that receives the connection request. The *server-tag* is the AAA server group defined by the **aaa-server** command.

Example 7-10 shows the commands sent by ASDM to the Cisco ASA to enable cut-through proxy.

Example 7-10 *Configuring Cut-Through Proxy Using the CLI*

```
access-list inside_authentication extended permit tcp any any
aaa authentication match inside_authentication inside my-radius-group
```

In Example 7-10, an ACL named inside_authentication is configured to **permit** (or match) TCP traffic from any source to any destination. This ACL is then applied to the **aaa authentication match** command. The **inside** keyword specifies that this rule is applied to the inside interface. The AAA server group named my-radius-group is associated to the end of the command.

You can also add exceptions to not authenticate certain users based on IP address. Figure 7-9 illustrates an example of how the **aaa authentication match** command works. SecureMeInc.org has two users in the 10.10.1.0/24 network who need to access the web server in the 10.10.2.0/24 network. The Cisco ASA is configured to authenticate all users in the 10.10.1.0 network; however, User2 is allowed to connect to the web server without being authenticated.

The following are the steps represented in Figure 7-9:

1. User1 attempts to access the web server (10.10.2.88).
2. The Cisco ASA prompts the user to authenticate.
3. User1 replies with his credentials.
4. The Cisco ASA sends the authentication request (Access-Request) to the Cisco ISE RADIUS server (10.10.1.141).
5. The Cisco ISE server sends back its reply (Access-Accept) to the Cisco ASA.
6. User1 is able to access the web server.

User2 can access the web server without being required to authenticate.

The commands to achieve this configuration are included in Example 7-11.

Example 7-11 *Configuring Firewall Session Authentication Exceptions*

```
!An ACL is configured to require authentication of all traffic except for User2
(10.10.1.20)
access-list 150 extended deny ip host 10.10.1.20 any
access-list 150 extended deny ip host 172.18.124.20 any
!
!The aaa authentication match command is configured with the corresponding ACL.
aaa authentication match 150 inside my-radius-group
```

Cisco ASA is capable of excluding authentication for devices by using their MAC addresses. This feature is practical when bypassing authentication for devices such as printers and IP phones. Create a MAC address list by using the **mac-list** command. Subsequently, use the **aaa mac-exempt** command to bypass authentication for the specified MAC addresses on the list. Example 7-12 demonstrates how to configure the Cisco ASA to achieve this functionality.

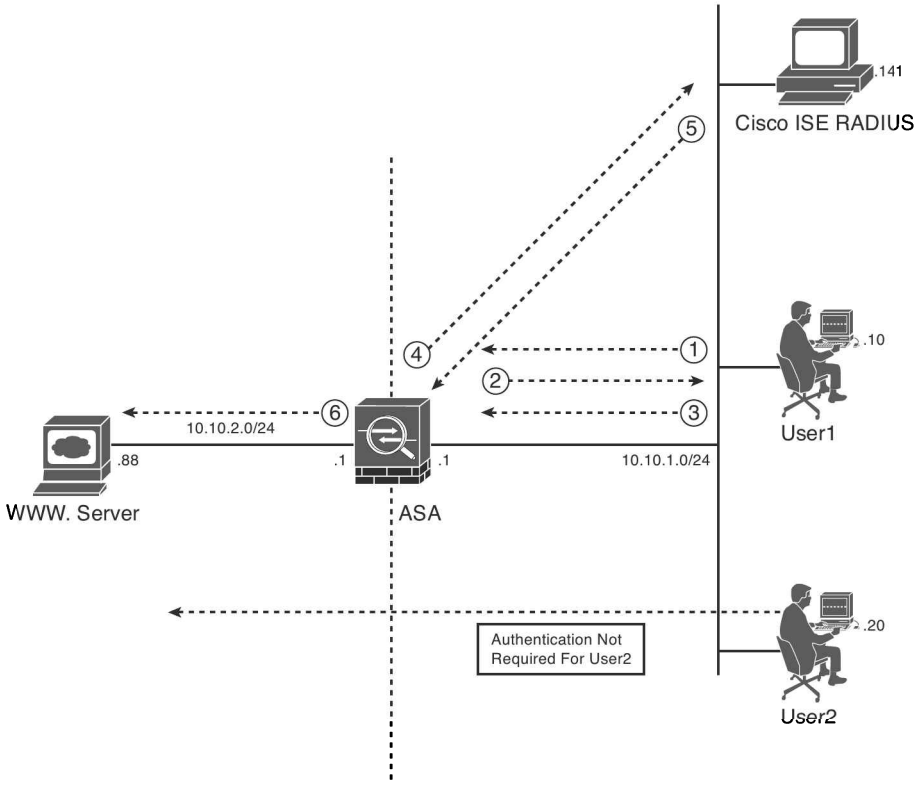

Figure 7-9 *Firewall Session Authentication Exceptions*

Example 7-12 *Configuring Authentication Exceptions by Using MAC Address Lists*

```
mac-list MACLIST permit 0003.470d.61aa ffff.ffff.ffff
mac-list MACLIST permit 0003.470d.61bb ffff.ffff.ffff
aaa mac-exempt match MACLIST
```

In Example 7-12, a MAC list named MACLIST is defined with two host MAC addresses and is associated with the **aaa mac-except** command.

Note Only one MAC list can be associated with **aaa mac-exempt.**

Both authentication and authorization are bypassed if this feature is turned on.

Authentication Timeouts

Authentication timeouts specify how long the Cisco ASA should wait before requiring the user to reauthenticate after a period of inactivity or absolute duration. Customize authentication timeouts in ASDM by navigating to **Configuration > Firewall > Advanced > Global Timeouts** and editing the Authentication Inactivity Timeout field. Alternatively, you can configure authentication timeouts via the CLI by using the **timeout uauth** command. The following is the command syntax:

```
timeout uauth hh:mm:ss [absolute | inactivity]
```

The inactivity timer begins after a user connection becomes idle. The absolute timer runs continuously. If you use the inactivity and absolute timeouts at the same time, the absolute timeout duration is longer than the inactivity timeout. If you set the timeouts the opposite way, the inactivity timeout does not work because the absolute timeout always expires sooner.

Tip It is recommended to configure the **absolute timeout** command value for at least 2 minutes. Never configure the **timeout uauth duration** to 0, because the authentication session would then never time out.

Additionally, you can use the **clear uauth** command to delete all cached credentials and make all users reauthenticate when attempting to create a new connection through the Cisco ASA. You can append a username at the end of the command to make a specific user reauthenticate. For example, use **clear uauth joe** to force a user called "joe" to reauthenticate.

Customizing Authentication Prompts

Cisco ASA enables you to customize the authentication prompts by navigating to the **Configuration > Device Management > Users/AAA > Authentication Prompt** in ASDM and entering an authentication prompt in the Prompt area. Similarly, you can use the **auth-prompt** command in the CLI to customize the authentication prompt. This customization is available only for Telnet, HTTP, or FTP authentication. The following is the usage and syntax of this command:

```
auth-prompt [prompt | accept | reject] prompt text
```

Table 7-4 lists all the options of the auth-prompt command.

Table 7-4 auth-prompt *Command Options*

Option	Description
prompt text	The actual text that will be printed at challenge, accept, or reject time.
prompt	Specifies that text following this keyword is printed as the authentication prompt.
accept	The text following this keyword is printed at authentication acceptance.
reject	The text following this keyword is printed at authentication rejection.

Note The **accept** and **reject** options apply only for Telnet connections.

Configuring Authorization

Cisco ASA supports authorization services over TACACS+ for firewall cut-through proxy sessions. It also supports authorization services over TACACS+ and its internal user database for administrative sessions. RADIUS-downloadable ACLs are also supported by Cisco ASA. Command access is authorized by privilege level only when authorization is performed against the local database.

Additionally, authorization over RADIUS, LDAP, and internal user databases is available for VPN user connections. This is used for *mode-config* attributes for remote-access VPN clients. Information about *mode-config* and its attributes is provided in Chapter 22, "Clientless Remote-Access SSL VPNs."

Complete the following steps to configure authorization with ASDM:

1. Log in to ASDM and navigate to **Configuration > Firewall > AAA Rules**.
2. Click **Add** and choose **Add Authorization Rule**. The dialog box shown in Figure 7-10 is displayed.

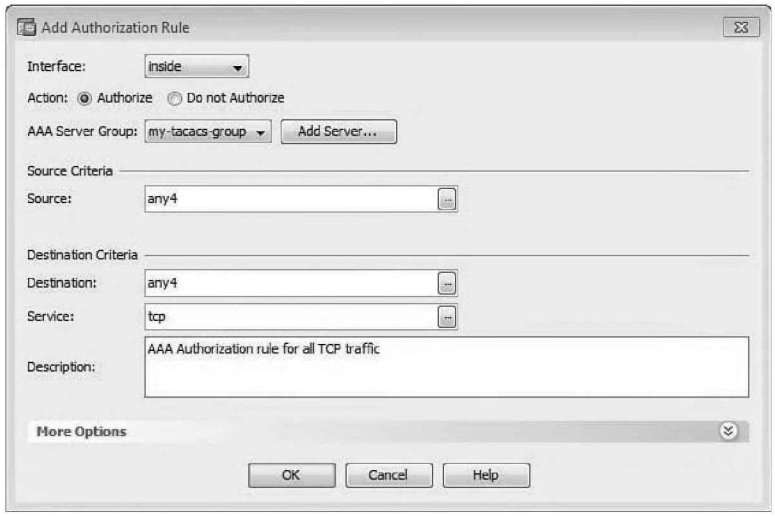

Figure 7-10 *Add Authorization Rule Dialog Box*

3. From the Interface drop-down list, choose the interface where the authentication rule will be applied. The inside interface is selected in this example.
4. Click **Authorize** in the Action field to require user authentication.
5. Choose the AAA server group **my-tacacs-group** from the AAA Server Group drop-down menu.

Note Add an AAA server to the server group by clicking the **Add Server** button. In Figure 7-10, the preconfigured AAA server is used. The TACACS+ server was previously added to the configuration via navigation to Configuration > Device Management > Users/AAA > AAA Server Groups.

6. You must specify a source and a destination for traffic that requires authorization. Enter the source IP address, network address, or the **any** keyword in the Source field. Alternatively, you can click the ellipsis button (...) to select an address that has already been configured in ASDM. In this instance, the **any4** keyword is entered to require authentication for any IPv4 source from the inside interface.
7. Enter the destination IP address, network address, or the **any** keyword in the Destination field. Alternatively, you can click the ellipsis button (...) to select an address that has already been configured in ASDM. In Figure 7-10, the **any4** keyword is entered to require authorization when a host tries to reach any IPv4 destination
8. Enter an IP service name for the destination service in the Service field. Alternatively, click the ellipsis button (...) to open a separate dialog box where you can select from a list of available services. In this case, authentication is required for any host trying to access any TCP-based applications.
9. (Optional) Enter a description for the authentication rule in the Description field.

Note Click More Options if you want to specify a source service for TCP or UDP applications or set a time range within which this rule is to be applied.

10. Click OK.
11. Click **Apply** to apply the configuration changes.
12. Click **Save** to save the configuration in the Cisco ASA.

Alternatively, in the CLI, you can use the **aaa authorization match** command to enable authorization for firewall cut-through proxy and administrative sessions. The following is the syntax for this command to enable authorization for firewall cut-through proxy sessions:

```
aaa authorization match access_list_name if_name server_tag
```

The *access_list_name* option specifies the ACL name used to categorize which traffic requires authorization.

Command Authorization

You enable command authorization in ASDM by following these steps:

1. Log in to ASDM and navigate to **Configuration > Device Management > Users/ AAA > AAA Access > Authorization.**
2. Check the **Enable** check box to enable authorization.
3. Choose the AAA server group from the Server Group drop-down list.

Note TACACS+ server commands can be configured as a shared profile component, for a group, or for individual users. If you enable TACACS+ command authorization, and a user enters a command at the CLI, the Cisco ASA sends the command and username to the TACACS+ server to determine whether the command is authorized.

4. (Optional) Check the **Use LOCAL When Server Group Fails** check box as a fall-back method in case the TACACS+ server is unreachable.
5. To perform authorization for exec shell access, check the **Enable** check box in the Perform Authorization for Exec Shell Access area. Specify whether authorization is performed by using the remote server parameters or the local authentication server.
6. Click **Apply** to apply the configuration changes.
7. Click **Save** to save the configuration in the Cisco ASA.

To configure command authorization via the CLI, use the following command:

```
aaa authorization command {LOCAL | tacacs_server_tag [LOCAL]}
```

The server tag **LOCAL** defines local command authorization. It can also be used as a fallback method in case the TACACS+ server is unreachable.

When using authorization, the following attributes are passed to the TACACS+ server in the attribute payload of the authorization request message:

- **cmd:** The command string to be authorized (used for authorization for administrative sessions only)
- **cmd-arg:** The command arguments to be sent (used for authorization for administrative sessions only)
- **service:** The type of service for which authorization is requested

The following attributes may be received from a TACACS+ server in an authorization response message:

- **idletime:** Idle timeout value for firewall cut-through proxy sessions
- **timeout:** Absolute timeout value for firewall cut-through proxy sessions
- **acl:** The identifier of an ACL to be applied to a specific user

Configuring Downloadable ACLs

Cisco ASA provides support for a per-user ACL authorization by enabling you to download an ACL from a RADIUS or TACACS+ server. This feature allows you to push an ACL to the Cisco ASA from a Cisco Secure ACS server or a Cisco ISE server. The downloadable ACL works in combination with the ACLs configured in the ASA. The user traffic needs to be permitted by both ACLs for it to flow through the ASA. However, the **per-user-override** option can be configured at the end of the **access-group** command to bypass this requirement. The following is an example of applying the **per-user-override** option on an **access-group** command applied to the inside interface:

```
access-group inside_access_ in interface inside per-user-override
```

In ASDM, configure this by navigating to **Configuration > Firewall > Access Rules** and clicking the **Advanced** button. The Access Rules Advanced Options dialog box is displayed, enabling you to select the **Per User Override** option on each access list entry.

All downloadable ACLs are applied to the interface from which the user is authenticated.

Figure 7-11 and these steps illustrate how downloadable ACLs work:

1. A user initiates a web connection to Cisco.com. The Cisco ASA is configured to perform authentication (cut-through proxy) and prompts the user for authentication credentials.
2. The user replies with his credentials.
3. The Cisco ASA sends the RADIUS authentication request (Access-Request) to the Cisco ISE server.

4. The Cisco ISE server authenticates the user and sends a RADIUS response (Access-Accept), including an ACL name associated with the user.
5. The Cisco ASA verifies whether it has an ACL named the same as the one downloaded from the Cisco ISE server. There is no need to download a new ACL if an ACL is identified with the same name.
6. The user is able to access Cisco.com.

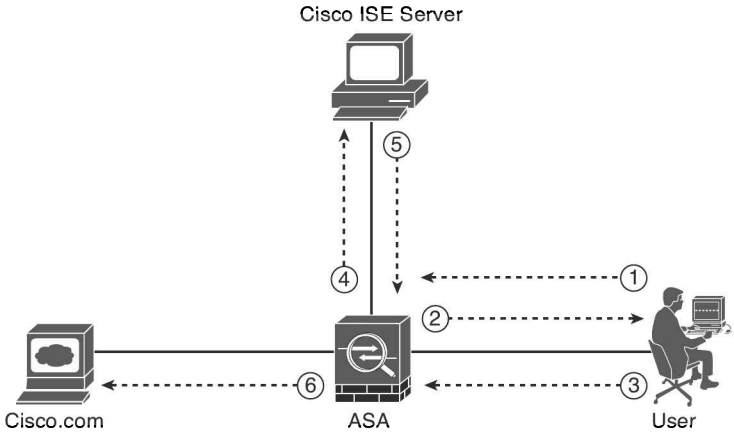

Figure 7-11 *Downloadable ACL Example*

You can configure downloadable ACLs in Cisco ISE in a few different ways:

- Configure a Shared Profile Component (SPC), including both the ACL name and the actual ACL. This enables you to apply the ACL to any number of users within Cisco ISE.
- Configure each ACL entry within a specific user profile.
- Configure the ACLs to be applied to a specific group.

These options are supported with Cisco ASA to better fit your security policies.

Configuring Accounting

To configure accounting on the Cisco ASA via ASDM, complete the following steps. The goal is to enable accounting for all IP traffic sourced from the 10.10.1.0/24 network and destined to the 10.10.2.0/24 network.

1. Log in to ASDM and navigate to **Configuration > Firewall > AAA Rules**.
2. Click **Add** and choose **Add Accounting Rule**.
3. Select the interface where the accounting rule is to be applied. In this case, the inside interface is used.

4. Under Action, choose **Account** to enable accounting.
5. Choose the AAA server group from the AAA Server Group drop-down list.
6. Configure the source and destination to define specific traffic traversing the Cisco ASA that is to be used for accounting. Design the specific source IP address or network under the Source field. By default, the **any** keyword is displayed to enable accounting for all sources. In this instance, the source network 10.10.1.0/24 is used.
7. Configure the specific destination IP address or network under the Destination field. By default, the **any** keyword is displayed to enable accounting for all sources. In this case, the destination network 10.10.2.0/24 is used.
8. Select the specific service or protocol in the Service field.
9. (Optional) Enter a description for this accounting rule in the Description field.
10. Click **Apply** to apply the configuration changes.
11. Click **Save** to save the configuration in the Cisco ASA.

To enable accounting via the CLI, use the **aaa accounting** command:

```
aaa accounting match access_list_name if_name server_tag
```

Example 7-13 demonstrates how to configure accounting on the Cisco ASA via the CLI. In Example 7-13, the previously configured AAA server group called my-radius-group is used. The **ip** keyword is used to enable accounting for all IP traffic sourced from the 10.10.1.0/24 network and destined to the 10.10.2.0/24 network.

Example 7-13 *Enabling Accounting by Using an ACL to Define Interesting Traffic*

```
NewYork (config)# access-list 100 permit ip 10.10.1.0 255.255.255.0 10.10.2.0
255.255.255.0
NewYork (config)# aaa accounting match 100 inside my-radius-group
```

In Example 7-13, an ACL is configured to enable accounting for all connections initiated from 10.10.1.0/24 to 10.10.2.0/24. The ACL is then applied to the **aaa accounting match** command. A previously defined AAA server group named my-radius-group is used with this command.

You can also use the **aaa accounting include | exclude** command options, as demonstrated for the **aaa authentication** command. The **aaa accounting match** command makes the **include** and **exclude** options obsolete and is the preferred method.

RADIUS Accounting

Table 7-5 lists all the RADIUS accounting messages supported by Cisco ASA.

The **accounting-on** message marks the start of accounting services. Subsequently, to mark the end of accounting services, use the **accounting-off** message. The **start** and **stop accounting records** messages are used to label when a user started and stopped a connection to a specific service. These sessions are labeled with their own accounting session IDs.

Table 7-5 *RADIUS Accounting Messages Supported in the Cisco ASA*

Attribute	Applicable Messages
acct-authentic	on, off, start, stop
acct-delay-time	on, off, start, stop
acct-status-type	on, off, start, stop
acct-session-id	start, stop
nas-ip-address	on, off, start, stop
nas-port	on, off, start, stop
user-name	on, off, start, stop
class	start, stop
service type	start, stop
framed-protocol	start, stop
framed-ip-address	start, stop
tunnel-client-endpoint	start, stop
acct-session-time	stop
acct-input-packets	stop
acct-output-packets	stop
acct-input-octets	stop
acct-output-octets	stop
acct-terminate-cause	stop
login-ip-host	on, off, start, stop
login-port	on, off, start, stop
cisco–av-pair (used to send source addr/port and dest addr/port)	on, off, start, stop
isakmp-initiator-ip	on, off, start, stop
isakmp-phase1-id	on, off, start, stop
isakmp-group-id	on, off, start, stop
acct-input-gigawords	stop
acct-output-gigawords	stop

TACACS+ Accounting

Table 7-6 lists all the TACACS+ accounting messages that Cisco ASA supports.

Table 7-6 *TACACS+ Accounting Messages Supported by Cisco ASA*

Attribute	Applicable Messages
username (fixed field)	start, stop
port (NAS) (fixed field)	start, stop
remote_address (fixed field)	start, stop
task_id	start, stop
foreign_IP	start, stop
local_IP	start, stop
cmd	start, stop
elapsed_time	stop
bytes_in	stop
bytes_out	stop

Cisco ASA also enables you to configure command accounting based on the user's privilege level. Use the following command to enable this feature:

```
aaa accounting command {privilege level}  tacacs_server_tag
```

Example 7-14 demonstrates how to configure command accounting on the Cisco ASA based on the user's privilege level.

Example 7-14 *Enabling Command Accounting*

```
New-York(config)# aaa accounting command privilege 15 my-tacacs-group
```

In Example 7-14, the **accounting** command is enabled for users that execute a **privilege level 15** command.

Alternatively, you can configure command accounting via ASDM by navigating to **Configuration > Device Management > Users/AAA > AAA Access > Accounting** and checking **Enable** in the Require Command Accounting for ASA area.

Troubleshooting Administrative Connections to Cisco ASA

Authenticate administrative connections by using RADIUS, TACACS+, or the Cisco ASA local user database. The following **debug** commands are available to troubleshoot AAA problems when you are trying to connect to the Cisco ASA for administration:

- **debug aaa:** Provides information about the authentication, authorization, or accounting messages generated and received by the Cisco ASA.

- **debug radius:** To troubleshoot RADIUS transactions, use this command, which has several options:
  - **all:** Enables all debug options.
  - **decode:** Shows decoded RADIUS transaction messages.
  - **session:** Provides information about all RADIUS sessions.
  - **user:** Enables you to capture RADIUS transaction information for a specific user connection.
- **debug tacacs:** To troubleshoot TACACS+ transactions, use this command with either of the following options:
  - **session:** Provides detailed information about all TACACS+ transactions.
  - **user:** Allows you to capture TACACS+ transaction information for a specific user connection.

If you enter **debug tacacs** without any options, the **debug** command is enabled with the **session** option by default. Example 7-15 includes the output of **debug tacacs** during a successful Telnet authentication.

Example 7-15 *Output of* **debug tacacs** *During a Successful Telnet Authentication*

```
NewYork# debug tacacs
 mk_pkt - type: 0x1, session_id: 4
 user: user1
 Tacacs packet sent
Sending TACACS Start message. Session id: 4, seq no:1
Received TACACS packet. Session id:4  seq no:2
tacp_procpkt_authen: GETPASS
Authen Message: Password:
mk_pkt - type: 0x1, session_id: 4
mkpkt_continue - response: ***
 Tacacs packet sent
Sending TACACS Continue message. Session id: 4, seq no:3
Received TACACS packet. Session id:4  seq no:4
tacp_procpkt_authen: PASS
TACACS Session finished. Session id: 4, seq no: 3
```

In Example 7-15, User1 connected to the Cisco ASA via Telnet. The Cisco ASA was configured to perform authentication via an external TACACS+ server. The first highlighted line shows that User1 attempted a connection to the Cisco ASA. The second highlighted line shows the ASA requesting the user's password. The user information is sent to the TACACS+ server and is finally authenticated. The third highlighted line shows that the authentication was successful.

Example 7-16 includes the output of **debug tacacs** during an authentication failure; the incorrect password was entered by the user and the TACACS+ server failed its authentication.

Example 7-16 *Output of* **debug tacacs** *During a Failed Authentication Because of Wrong Password*

```
New York# debug tacacs
 mk_pkt - type: 0x1, session_id: 5
 user: user1
 Tacacs packet sent
Sending TACACS Start message. Session id: 5, seq no:1
Received TACACS packet. Session id:5  seq no:2
tacp_procpkt_authen: GETPASS
Authen Message: Password:
mk_pkt - type: 0x1, session_id: 5
mkpkt_continue - response: ***
 Tacacs packet sent
Sending TACACS Continue message. Session id: 5, seq no:3
Received TACACS packet. Session id:5  seq no:4
tacp_procpkt_authen: FAIL
TACACS Session finished. Session id: 5, seq no: 3
```

In Example 7-17, the TACACS+ server was offline or unreachable.

Example 7-17 *Output of* **debug tacacs** *While TACACS+ Server Is Unreachable*

```
NewYork# debug tacacs
mk_pkt - type: 0x1, session_id: 6
 user: user1
 Tacacs packet sent
Sending TACACS Start message. Session id: 6, seq no:1
Received TACACS packet. Session id:6  seq no:2
TACACS Request Timed out. Session id: 6, seq no:1
TACACS Session finished. Session id: 6, seq no: 1
mk_pkt - type: 0x1, session_id: 6
user: user1
 Tacacs packet sent
Sending TACACS Start message. Session id: 6, seq no:1
Received TACACS packet. Session id:6  seq no:2
TACACS Request Timed out. Session id: 6, seq no:1
TACACS Session finished. Session id: 6, seq no: 1
mk_pkt - type: 0x1, session_id: 6
 user: user1
 Tacacs packet sent
Sending TACACS Start message. Session id: 6, seq no:1
```

```
Received TACACS packet. Session id:6  seq no:2
TACACS Request Timed out. Session id: 6, seq no:1
TACACS Session finished. Session id: 6, seq no: 1
aaa server host machine not responding
```

The highlighted lines in Example 7-17 show how the Cisco ASA attempts to communicate with the TACACS+ server three times and finally times out all authentication transactions. The **show aaa-server** command is useful while troubleshooting and monitoring authentication transactions. Example 7-18 includes the output of the **show aaa-server** command for all TACACS+ transactions.

Example 7-18 *Monitoring and Troubleshooting TACACS+ Transactions with the* **show aaa-server** *Command*

```
NewYork# show aaa-server protocol tacacs+
Server Group:    mygroup
Server Protocol: tacacs+
Server Address:  172.18.124.145
Server port:     49
Server status:   ACTIVE, Last transaction at 21:05:43 UTC Fri March 20 2009
Number of pending requests              0
Average round trip time                 43ms
Number of authentication requests       4
Number of authorization requests        0
Number of accounting requests           0
Number of retransmissions               0
Number of accepts                       3
Number of rejects                       1
Number of challenges                    4
Number of malformed responses           0
Number of bad authenticators            0
Number of timeouts                      0
Number of unrecognized responses        0
```

In Example 7-18, the Cisco ASA processed a total of four authentication requests. Three of those requests were successfully authenticated and one was rejected by the TACACS+ server.

Troubleshooting Firewall Sessions (Cut-Through Proxy)

The techniques to troubleshoot cut-through proxy sessions on Cisco ASA are similar to those mentioned in the previous section. Additionally, the **show uauth** command can be used to display information about authenticated users and current transactions. Example 7-19 shows the output of this command.

Example 7-19 *Output of the* **show uauth** *Command*

```
NewYork# show uauth
                        Current    Most Seen
Authenticated Users       0          0
Authen In Progress        1          3
```

In Example 7-19, a total of three concurrent authentication requests were processed by the Cisco ASA and one is currently being processed.

ASDM and CLI AAA Test Utility

Cisco ASDM and the Cisco ASA CLI provide the capability for an administrator to test authentication or authorization of a given user. This is very helpful for troubleshooting because it allows an administrator to perform these tests without having to ask non-technical users to try a connection or authentication attempt. In ASDM, the AAA server test utility is accessed under **Configuration > Device Management > Users/AAA > AAA Server Groups**, as shown in Figure 7-12.

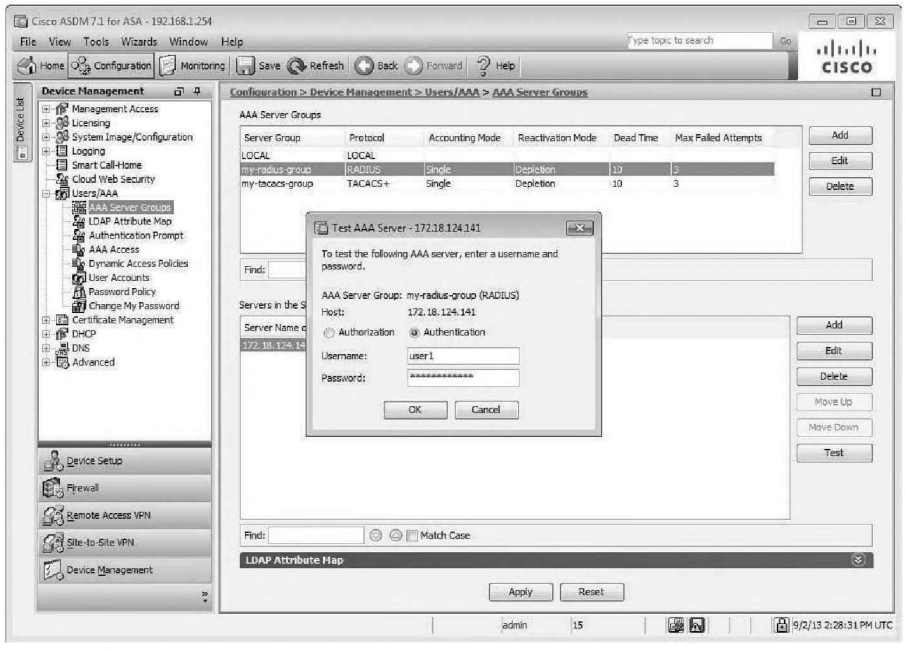

Figure 7-12 *ASDM AAA Authentication and Authorization Test Utility*

In Figure 7-12, an administrator is testing authentication for a user with the username user1. Alternatively, an administrator can do the same by using the **test aaa-server authentication** command, as shown in Example 7-20. In Example 7-20, the authentication attempt times out because there is a communication problem between the ASA and the AAA server (172.18.124.141).

Example 7-20 **test aaa-server** *authentication Command*

```
NewYork# test aaa-server authentication my-radius-group username user1 password
thisisthepassword
Server IP Address or name: 172.18.124.141
INFO: Attempting Authentication test to IP address <172.18.124.141> (timeout: 12
seconds)
ERROR: Authentication Server not responding: No response from server
```

Summary

Cisco ASA supports several AAA solutions for different services. It ensures the enforcement of assigned policies by allowing you to control who can log in to the Cisco ASA or in to the network. Additionally, it controls what each user is allowed to do. It can also record security audit information by using accounting services. This chapter covered how Cisco ASA can use authentication services to control pass-through access by requiring valid user credentials. It also demonstrated how Cisco ASA is configured to authenticate administrative sessions from Telnet, SSH, serial console, and ASDM.

This chapter demonstrated how authorization can enforce per-user access control after authentication is done. It guided you through configuring the Cisco ASA to authorize management and administrative commands and network access.

The Cisco ASA accounting services track traffic that passes through the security appliance, enabling you to have a record of user activity. This chapter also demonstrated how you can enable accounting to track and audit user activity.