



## Administration

---

This chapter describes management protocols and Network Access Server (NAS) security and control functionality with AAA and RADIUS servers.

- Remote Monitor (RMON), page 4-1
- Enabling Management Protocols: NTP, SNMP, and Syslog, page 4-2
  - Enabling the Network Time Protocol, page 4-3
  - Enabling Syslog, page 4-4
  - Enabling SNMP, page 4-7
  - Disabling the Logging of Access Interfaces, page 4-9
  - Confirming the Final Running Configuration, page 4-10
- Local and Remote Server Authentication, page 4-13
  - Configuring RADIUS, page 4-14
  - Configuring TACACS+, page 4-24

### Remote Monitor (RMON)

Remote Monitoring (RMON) is an Internet Engineering Task Force (IETF) monitoring standard (RFC 1757) that allows console systems and network monitors to exchange statistical and functional monitoring data, through RMON-compliant console managers and network probes. RMON provides network administrators with flexibility to satisfy networking demands through console and network monitoring probes to obtain fault diagnostics, planning, and performance information.

RMON delivers information in nine unique monitoring element groups that provide specific types of data, which satisfies common network-monitoring requirements. Some RMON groups are dependent upon others for support, but each is optional so that it is not necessary for vendors to support all groups within the management information base (MIB). See Table 4-1 for RMON group functions.

**Table 4-1** *RMON Groups*

<b>RMON Group</b>	<b>Description</b>
Alarm	Periodic statistical sampling from event generated variables in the probe that compares configured thresholds.
Events	Controls the generation and notification of events from this device.
Filters	Enables packet matching by equation filtering to form data streams that may be captured or generate events.
History	Records and stores periodic statistical samples, number of samples, and items sampled from a network.
Host	Contains statistics associated with each discovered network host.
HostTopN	Creates tables describing hosts that top a list ordered by one of their rate-based statistics.
Matrix	Stores new conversation statistics detected on source and destination device.
Packet Capture	Enables packet capturing after it flows through a channel.
Statistics	Contains probe calculated statistics for each interface monitored on device.

## Enabling Management Protocols: NTP, SNMP, and Syslog

This section describes how to enable basic management protocols on a Cisco AS5800 as part of a dial access service. It does not however, describe how to integrate the Cisco IOS software with NT or UNIX servers. Management protocols are described only from the perspective of the Cisco IOS software.

### Understanding Network Management Basics

Figure 4-1 shows a logical perspective of how management protocols interact between the Cisco IOS software (client) and a network element management server. Dashed lines represent different protocols and functions.

- NTP synchronizes time between network devices.
- The SNMP element manager (EM) receives SNMP traps from the Cisco IOS software. The SNMP manager uses SNMP to query variables and set configurations.
- The Cisco IOS software sends logging messages to a syslog daemon.

Figure 4-1 NTP, SNMP, and Syslog Interactions

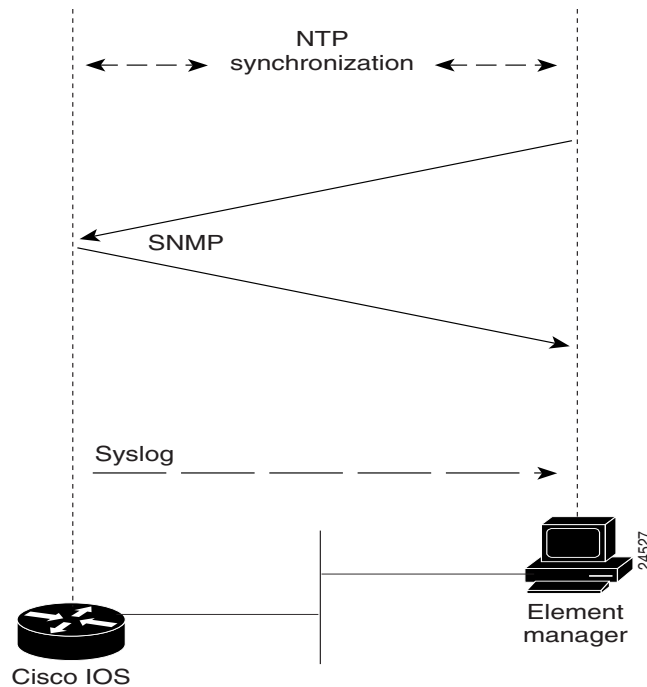


Table 4-2 provides the RFCs and URLs for the management protocols described in this section.

Table 4-2 Management Protocol RFCs

Management Protocol	RFC	URL
NTP	RFC 1305	<a href="http://www.ietf.org/rfc/rfc1305.txt">http://www.ietf.org/rfc/rfc1305.txt</a>
SNMP	RFC 1157	<a href="http://www.ietf.org/rfc/rfc1157.txt">http://www.ietf.org/rfc/rfc1157.txt</a>

For more information about system management, refer to Cisco IOS Release 12.0 *Configuration Fundamentals Configuration Guide* and *Command Reference*, available online at <http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/12cgcr/index.htm>

## Enabling the Network Time Protocol

The Network Time Protocol (NTP) provides a common time base for networked routers, servers, and other devices. A synchronized time enables you to correlate syslog and Cisco IOS debug output to specific events. For example, you can find call records for specific users within one millisecond.

Comparing logs from various networks is essential for:

- Troubleshooting
- Fault analysis
- Security incident tracking

Without precise time synchronization between all the various logging, management, and AAA functions, time comparisons are not possible.

An NTP enabled network usually gets its time from an authoritative time source, such as a Cisco router, radio clock, or an atomic clock attached to a timeserver. NTP then distributes this time across the network. NTP is extremely efficient; no more than one packet per minute is necessary to synchronize two machines to within a millisecond of each another. NTP runs over UDP, which in turn runs over IP.

**Step 1** Locate an authoritative clock source. For example, you can use a Cisco router or an atomic clock that is attached to a time server.

**Step 2** Specify the primary NTP server IP address and automatic calendar updates as shown below:

```
!
ntp update-calendar
ntp server 172.22.66.18 prefer
!
```

**Step 3** Verify that the clock is synchronized to the NTP server. Inspect the status and time association. Clock sources are identified by their stratum levels. The following example shows a stratum level five clock.

```
5800-NAS# show ntp status
Clock is synchronized, stratum 5, reference is 172.22.66.18
nominal freq is 250.0000 Hz, actual freq is 250.0000 Hz, precision is 2**24
reference time is BB944312.4451C9E7 (23:11:30.266 PDT Wed Sep 22 1999)
clock offset is 0.5343 msec, root delay is 13.26 msec
root dispersion is 18.02 msec, peer dispersion is 0.09 msec
5800-NAS#
```

The following command identifies how often the NAS is polling and updating to the stratum clock. An asterisk (\*) next to the NTP servers IP address indicates successful synchronization with the stratum clock.

```
5800-NAS# show ntp association

address      ref clock      st when poll reach delay offset disp
*~172.22.66.18 172.60.8.1    16  46  64 377  1.0  0.53  0.1
 * master (synced), # master (unsynced), + selected, - candidate, ~ configured
5800-NAS#
```

## Enabling Syslog

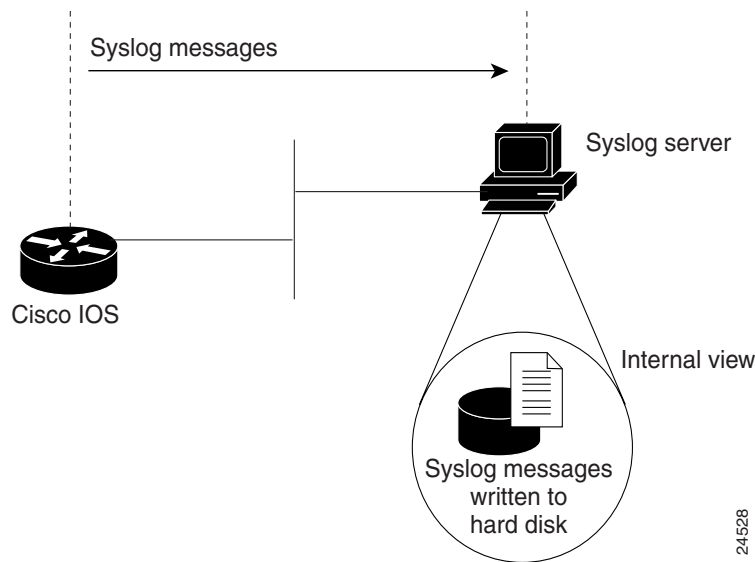
The Cisco IOS software can send syslog messages to one or more element manager servers. Syslog messages are then collected by a standard UNIX or NT type syslog daemon.

Syslog enables you to:

- Centrally log and analyze configuration events and system error messages such as interface status, security alerts, environmental conditions, and CPU process overloads.
- Capture client debug output sessions in a real-time scenario.
- Reserve Telnet sessions for making configurations changes and using **show** commands. This prevents Telnet sessions from getting cluttered up with debug output.

Figure 4-2 shows the Cisco IOS software sending syslog data to an element manager. Syslog data either stays in the Cisco IOS software buffer, or is pushed out and written to the element managers hard disk.

Figure 4-2 Syslog Messages Written to Hard Disk



**Note** Cisco System's UNIX syslog format is compatible with 4.3 BSD UNIX.

**Step 1** Enable debug timestamps and include date, time, and milliseconds relative to the local time zone:

```
!
service timestamps debug datetime msec localtime show-timezone
service timestamps log datetime msec localtime show-timezone
!
```

**Step 2** Verify that console logging is disabled. If it is enabled, the NAS will intermittently freeze up as soon as the console port is overloaded with log messages. See the field "1 flushes." Increments on this number represents bad logging behavior.

```
5800-NAS# show logging
Syslog logging: enabled (0 messages dropped, 1 flushes, 0 overruns)
  Console logging: level debugging, 1523 messages logged
  Monitor logging: level debugging, 0 messages logged
  Buffer logging: level debugging, 911 messages logged
  Trap logging: level informational, 44 message lines logged

5800-NAS(config)# no logging console
5800-NAS(config)# ^Z
5800-NAS# show logging
Syslog logging: enabled (0 messages dropped, 1 flushes, 0 overruns)
  Console logging: disabled
  Monitor logging: level debugging, 0 messages logged
  Buffer logging: level debugging, 912 messages logged
  Trap logging: level informational, 45 message lines logged
```



**Caution** Not entering the **no logging console** command might cause CPU interrupts, dropped packets, denial of service events, and router lock up.

**Step 3** Specify the logging configuration:

```
!
logging 172.22.66.18
logging buffered 10000 debugging
logging trap debugging
!
```

Table 4-3 describes the commands in the previous configuration fragment.

**Table 4-3 Syslog Commands**

Command	Purpose
<code>logging 172.22.66.18</code>	Specifies the syslog servers IP address.
<code>logging buffered 10000 debugging</code>	Sets the internal log buffer to 10,000 bytes for debug output (newer messages overwrite older messages).
<code>logging trap debugging</code>	Allows logging up to the debug level (all 8 levels) for all messages sent to the syslog server.

If you are working with multiple network access servers, assign a different logging facility tag to each server. Syslog information can be collected and sorted into different files on the syslog server.

For example:

- Assign local1 to NAS1
- Assign local2 to NAS2
- Assign local3 to NAS3

Assigning a different tag to each device enables you to intelligently sort and view syslog messages:

```
!
logging facility local7
!
```

**Step 4** Verify that local buffered logging is working:

```
5800-NAS# show logging
Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
  Console logging: disabled
  Monitor logging: level debugging, 0 messages logged
  Buffer logging: level debugging, 2 messages logged
  Trap logging: level debugging, 53 message lines logged
    Logging to 172.22.66.18, 2 message lines logged
```

**Log Buffer (10000 bytes):**

```
Sep 26 16:32:02.848 PDT: %SYS-5-CONFIG_I: Configured from console by admin on console
Sep 26 16:33:16.069 PDT: %SYS-5-CONFIG_I: Configured from console by admin on console
5800-NAS#
```

## Enabling SNMP

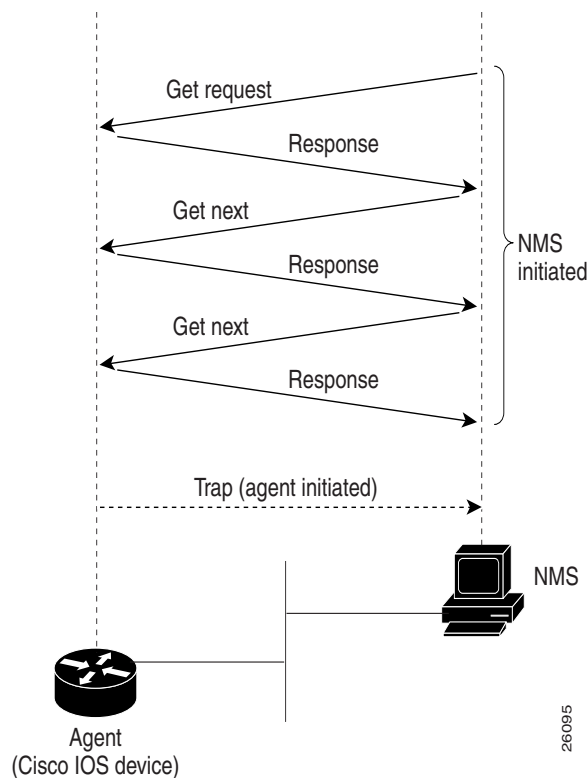
The SNMP traps generated by Cisco routers provide:

- Potentially harmful environmental conditions
- Processor status
- Port status
- Security issues

The Cisco IOS software generates SNMP traps based on the features that the Cisco IOS software supports.

Figure 4-3 shows the interactions and timing of the SNMP protocol between the EM (SNMP manager) and the NAS (SNMP agent). Traps are unsolicited messages sent from the NAS to the EM. Four functions of SNMP include: trap, get request, get next, and set request.

**Figure 4-3** *SNMP Event Interaction and Timing*



**Note**

For a listing of all SNMP traps supported by Cisco, refer to *Cisco IOS SNMP Traps Supported and How to Configure Them*, available online at [http://www.cisco.com/warp/public/477/SNMP/snmp\\_traps.html](http://www.cisco.com/warp/public/477/SNMP/snmp_traps.html)

**Step 1** Configure the Cisco IOS software to support basic SNMP functions. Access lists 5 and 8 are used for SNMP community strings:

- The read only (RO) community string is called “poptarts.” It uses access list 8 as a filter.
- The read write (RW) community string is called “pixysticks.” It uses access list 5 as a filter.

```

!
snmp-server contact admin user@the.doc
snmp-server location 5800-NAS-corporate
snmp-server community popstarts RO 8
snmp-server community pixysticks RW 5
snmp-server host 172.22.66.18 maddog
snmp-server trap-source Loopback0
snmp-server enable traps snmp
!
access-list 5 permit 172.22.67.1
access-list 5 permit 0.0.0.1 172.22.68.20
access-list 8 permit 172.22.67.1
access-list 8 permit 0.0.0.1 172.22.68.20
!

```

Table 4-4 describes commands in the previous configuration fragment.

**Table 4-4** *SNMP Commands*

Command	Purpose
<code>snmp-server contact admin user@the.doc</code>	Specifies a contact name to notify whenever a MIB problems occurs.
<code>snmp-server location 5800-NAS-corporate</code>	Specifies a geographic location name for the router.
<code>snmp-server community popstarts RO 8</code>	Assigns a read only (RO) community string. Only queries and get requests can be performed.  The community string (poptarts) allows polling but no configuration changes. Without the correct community string on both machines, SNMP will not let you do the authorization to get or set the request.
<code>snmp-server community pixysticks RW 5</code>	Assigns a read write (RW) community string.  This community string (pixysticks) enables configuration changes to be performed. For example, you can shut down an interface, download a configuration file, or change a password.
<code>snmp-server host 172.22.66.18 maddog</code>	Identifies the IP address of the SNMP host followed by a password.
<code>snmp-server trap-source Loopback0</code>	Associates SNMP traps with a loopback interface. In this way, an Ethernet shutdown will not disrupt SNMP management flow.
<code>snmp-server enable traps</code>	Enables traps for unsolicited notifications for configuration changes, environmental variables, and device conditions.
<code>access-list 5 permit 172.22.67.1</code> <code>access-list 8 permit 172.22.67.1</code>	Permits access from a single element management server.
<code>access-list 5 permit 0.0.0.1 172.22.68.20</code> <code>access-list 8 permit 0.0.0.1 172.22.68.20</code>	Permits access from a block of addresses at your network operations center.

**Caution**

If you are not using SNMP, make sure to turn it off. Never use a configuration that uses “public” or “private” as community strings—these strings are well known in the industry and are common defaults on hardware. These strings are open invitations to attacks, regardless if you use filters.

- Step 2** Monitor SNMP input and output statistics. For example, display a real-time view of who is polling the NAS for statistics and how often.

Excessive polling will:

- Consume much of the CPU resources
- Cause packets to be dropped
- Crash the NAS

```
5800-NAS# show snmp
Chassis: 11811596
Contact: admin user@the.doc
Location: 5800-NAS-corporate
0 SNMP packets input
  0 Bad SNMP version errors
  0 Unknown community name
  0 Illegal operation for community name supplied
  0 Encoding errors
  0 Number of requested variables
  0 Number of altered variables
  0 Get-request PDUs
  0 Get-next PDUs
  0 Set-request PDUs
0 SNMP packets output
  0 Too big errors (Maximum packet size 1500)
  0 No such name errors
  0 Bad values errors
  0 General errors
  0 Response PDUs
  0 Trap PDUs

SNMP logging: enabled
  Logging to 172.22.66.18.162, 0/10, 0 sent, 0 dropped.
5800-NAS#
```

## Disabling the Logging of Access Interfaces

Limit the amount of output logged from the group-async interface and ISDN D channels. Carefully choose the data sources for system management purposes. AAA accounting and the modem-call record terse feature provides the best data set for analyzing ISDN remote node device activity.

Link status up-down events and SNMP trap signals:

- Occur regularly on access interfaces. Dialer interfaces going up and down is normal behavior and does not indicate a problem.
- Should not be logged or sent to a management server.

The following configuration fragment disables logging on access interfaces:

```
!
interface Serial 0:23
  no logging event link-status
  no snmp trap link-status
!
interface Serial 1:23
  no logging event link-status
  no snmp trap link-status
!
interface Serial 2:23
  no logging event link-status
  no snmp trap link-status
!
interface Serial 3:23
  no logging event link-status
  no snmp trap link-status
!
interface Group-Async 1
  no logging event link-status
  no snmp trap link-status
!
```

## Confirming the Final Running Configuration

The following is an example of the Cisco AS5800 running configuration with Cisco IOS Release 12.0(4) XL1 installed.

```
5800-NAS# show running-config
Building configuration...

Current configuration:
!
version 12.x
service timestamps debug datetime msec localtime show-timezone
service timestamps log datetime msec localtime show-timezone
service password-encryption
!
hostname 5800-NAS
!
logging buffered 10000 debugging
no logging console
aaa new-model
aaa authentication login default local
aaa authentication ppp default if-needed local
enable secret 5 $1$LKgL$tgi19XvWn7fld7JGt55p01
!
username dude password 7 045802150C2E
username admin password 7 044E1F050024
!
!
!
shelf-id 0 router-shelf
shelf-id 1 dial-shelf
!
!
!
```

```
resource-pool disable
!
modem-pool Default
  pool-range 1/2/0-1/10/143
!
!
spe 1/2/0 1/10/11
  firmware ios-bundled default
modem recovery action none
ip subnet-zero
no ip source-route
ip host guessme 172.22.100.9
ip domain-name the.net
ip name-server 172.22.11.10
ip name-server 172.22.12.11
!
async-bootp dns-server 172.30.10.1 172.30.10.2
isdn switch-type primary-ni
isdn voice-call-failure 0
!
!
controller T3 1/0/0
  framing m23
  cablelength 0
  t1 4 controller
!
controller T1 1/0/0:4
  framing esf
  pri-group timeslots 1-24
!
!
voice-port 1/0/0:4:D
!
!
process-max-time 200
!
interface Loopback0
  ip address 172.22.99.1 255.255.255.255
  no ip directed-broadcast
!
interface Loopback1
  ip address 172.22.90.1 255.255.255.0
  no ip directed-broadcast
!
interface FastEthernet0/1/0
  ip address 172.22.66.23 255.255.255.0
  no ip directed-broadcast
!
interface Serial1/0/0:4:23
  no ip address
  no ip directed-broadcast
  no snmp trap link-status
  isdn switch-type primary-ni
  isdn incoming-voice modem
  no cdp enable
!
interface Group-Async0
  ip unnumbered FastEthernet0/1/0
  no ip directed-broadcast
  encapsulation ppp
  async mode interactive
  no snmp trap link-status
  peer default ip address pool addr-pool
  no cdp enable
```

```
    ppp authentication chap pap
    group-range 1/2/00 1/10/143
    !
ip local pool addr-pool 172.22.90.2 172.22.90.254
ip classless
ip route 0.0.0.0 0.0.0.0 172.22.66.1
no ip http server
!
logging trap debugging
logging 172.22.66.18
access-list 5 permit 172.22.67.1
access-list 5 permit 0.0.0.1 172.22.68.20
access-list 8 permit 172.22.67.1
access-list 8 permit 0.0.0.1 172.22.68.20
snmp-server engineID local 00000009020000D0D3424C1C
snmp-server community poptarts RO 8
snmp-server community pixysticks RW 5
snmp-server community maddog view vifdefault RO
snmp-server trap-source Loopback0
snmp-server location 5800-NAS-Austin
snmp-server contact admin dude@the.net
snmp-server enable traps snmp
snmp-server enable traps isdn call-information
snmp-server enable traps hsrp
snmp-server enable traps config
snmp-server enable traps entity
snmp-server enable traps envmon
snmp-server enable traps syslog
snmp-server enable traps rsvp
snmp-server enable traps frame-relay
snmp-server enable traps rtr
snmp-server enable traps dial
snmp-server enable traps dsp card-status
snmp-server enable traps bgp
snmp-server enable traps voice poor-qov
snmp-server host 172.22.66.18 maddog
!
banner login ^C
This is a secured device.
Unauthorized use is prohibited by law.
^C
!
line con 0
  transport input none
line aux 0
  transport input telnet
line vty 0 4
line 1/2/00 1/10/143
  autoselect during-login
  autoselect ppp
  modem InOut
  no modem log rs232
!
ntp update-calendar
ntp server 172.22.66.18 prefer
end
```

# Access Service Security

The Cisco AS5800 is designed to support a security paradigm providing authentication, authorization, and accounting (AAA) security measures using RADIUS and TACACS+.

- Authentication—requires dial-in users to identify themselves and prove their identity, thus preventing wrongful access to lines on your Cisco AS5800, or connecting through the lines directly to network resources.
- Authorization—prevents users from gaining access to particular services and devices on the network.
- Accounting—provides records for billing and other needs to determine who is connected to the network and how long they have been connected. It does not describe how to configure accounting.

This section describes how to configure security using a local database resident on your Cisco AS5800 or using a remote security database for Terminal Access Controller Access Control System with Cisco proprietary enhancements (TACACS+) and Remote Authentication Dial-In User Service (RADIUS). Refer to the “Local and Remote Server Authentication” section on page 4-13 for local and remote authentication definitions.

**Note**

---

This section does not provide a comprehensive security overview. It does not describe how to completely configure TACACS, Extended TACACS, access lists or RADIUS. It presents the most commonly used security mechanisms to prevent unauthenticated and unauthorized access to network resources through a Cisco AS5800. For a comprehensive overview of Cisco security tools, refer to the security configuration guide in the Cisco IOS configuration guides and command references documentation.

---

This section describes the following topics:

- Local and Remote Server Authentication
- Configuring RADIUS
- Configuring TACACS+

## Local and Remote Server Authentication

This section describes the differences between local and remote security databases and the basic authentication process for each. Remote security databases described in this section include Terminal Access Controller Access Control System with Cisco proprietary enhancements (TACACS+) and Remote Authentication Dial-In User Service (RADIUS).

Generally the size of the network and type of corporate security policies and control determine whether you use a local or remote security database.

### Local Security Database

If you have one or two Cisco AS5800 providing access to your network, store username and password security information on your Cisco AS5800. This is referred to as local authentication.

## Remote Security Database

As your network expands, you need a centralized security database that provides username and password information each access server in the network. This centralized security database resides in a security server.

A centralized security database helps establish consistent remote access policies throughout a corporation. An example of a remote security database server is the CiscoSecure product from Cisco Systems. CiscoSecure is a UNIX security daemon, with which the administrator creates a database that defines the network users and their privileges. CiscoSecure uses a central database that stores user and group profiles with authentication and authorization information.

The Cisco AS5800 exchanges user authentication information with a TACACS+ or RADIUS database on the security server by transmitting encrypted TACACS+ or RADIUS packets across the network.

For specific information about the interaction between the security server and the Cisco AS5800, refer to the security configuration guide in the Cisco IOS configuration guides and command references documentation.

## Configuring RADIUS

This section describes the Remote Authentication Dial-In User (RADIUS) security system, defines its operation, and identifies appropriate and inappropriate network environments for using RADIUS technology. RADIUS Configuration Task List, page 4-16 describes how to configure RADIUS with the authentication, authorization, and accounting (AAA) command set. RADIUS Configuration Examples, page 4-20 offers two possible implementation scenarios.

This section includes the following topics:

- RADIUS Overview, page 4-14
- RADIUS Operation, page 4-15
- RADIUS Configuration Task List, page 4-16

For a complete description of the commands used in this section, refer to information on RADIUS commands in the security command reference for your Cisco IOS release. To locate documentation of other commands that appear in this section, use the command reference master index or search online.

## RADIUS Overview

RADIUS is a distributed client/server system that secures networks against unauthorized access. In the Cisco implementation, RADIUS clients run on Cisco routers and send authentication requests to a central RADIUS server. The server contains all user authentication and network service access information.

RADIUS is a fully open protocol, distributed in source code format, that can be modified to work with any security system currently available on the market.

Cisco supports RADIUS under its AAA security paradigm. RADIUS can be used with other AAA security protocols, such as TACACS+, Kerberos, or local username lookup. RADIUS is supported on all Cisco platforms.

RADIUS has been implemented in a variety of network environments that require high levels of security while maintaining network access for remote users.

Use RADIUS in the following network environments that require access security:

- Networks with multiple-vendor access servers, each supporting RADIUS. For example, access servers from several vendors use a single RADIUS server-based security database. In an IP-based network with multiple vendors access servers, dial-in users are authenticated through a RADIUS server that has been customized to work with the Kerberos security system.
- Turnkey network security environments in which applications support the RADIUS protocol, such as in an access environment that uses a “smart card” access control system. In one case, RADIUS has been used with Enigmas security cards to validate users and grant access to network resources.
- Networks already using RADIUS. You can add a Cisco router with RADIUS to the network. This might be the first step when you make a transition to a Terminal Access Controller Access Control System (TACACS+) server.
- Networks in which a user must only access a single service. Using RADIUS, you can control user access to a single host, to a single utility such as Telnet, or to a single protocol such as Point-to-Point Protocol (PPP). For example, when a user logs in, RADIUS identifies this user as having authorization to run PPP using IP address 10.2.3.4 and the defined access list is started.
- Networks that require resource accounting. You can use RADIUS accounting independent of RADIUS authentication or authorization. The RADIUS accounting functions allow data to be sent at the start and end of services, indicating the amount of resources (such as time, packets, and bytes used during the session).
- An Internet service provider (ISP) might use a freeware-based version of RADIUS access control and accounting software to meet special security and billing needs.

RADIUS is not suitable in the following network security situations:

- Multiprotocol access environments. RADIUS does not support the following protocols:
  - AppleTalk Remote Access Protocol (ARAP)
  - NetBIOS Frame Protocol Control Protocol (NBFCP)
  - NetWare Asynchronous Services Interface (NASI)
  - X.25 PAD connections
- Router-to-router situations. RADIUS does not provide two-way authentication. RADIUS can be used to authenticate from one Cisco router to a third party router if, other company’s router requires RADIUS authentication.
- Networks using a variety of services. RADIUS generally binds a user to one service model.

## RADIUS Operation

When attempting to log in and authenticate to Cisco AS5800 using RADIUS, the following steps occur:

1. The user enters a username and password at the corresponding prompts.
2. The username and encrypted password are sent over the network to the RADIUS server.
3. The user receives one of the following responses from the RADIUS server:
  - ACCEPT - The user is authenticated.
  - REJECT - The user is not authenticated and is prompted to reenter the username and password, or access is denied.
  - CHALLENGE - A challenge is issued by the RADIUS server. The challenge collects additional data from the user.

- **CHANGE PASSWORD** - A request is issued by the RADIUS server, asking the user to select a new password.

The **ACCEPT** or **REJECT** response is bundled with additional data that is used for **EXEC** or network authorization. You must first complete RADIUS authentication before using RADIUS authorization. The additional data included with the **ACCEPT** or **REJECT** packets consists of the following:

- Services that the user can access, including Telnet, rlogin, local-area transport (LAT) connections, and PPP, Serial Line Internet Protocol (SLIP), or **EXEC** services.
- Connection parameters, including the host or client IP address, access list, and user timeouts.

## RADIUS Configuration Task List

To configure RADIUS on your Cisco AS5800, you must perform the following tasks:

- Use the **aaa new-model** global configuration command to enable AAA. AAA must be configured if you plan to use RADIUS. For more information about using the **aaa new-model** command, refer to the “AAA Commands” section on page 4-23.
- Use the **aaa authentication** global configuration command to define method lists for RADIUS authentication. For more information about using the **aaa authentication** command, refer to the “Specify RADIUS Authentication” section on page 4-20.
- Use **line** and **interface** commands to enable the defined method lists to be used. For more information, refer to the “Specify RADIUS Authentication” section on page 4-20.

The following configuration tasks are optional:

- Use the **aaa authorization** global command to authorize specific user functions. For more information about using the **aaa authorization** command, refer to the “Specify RADIUS Authorization” section on page 4-20.
- Use the **aaa accounting** command to enable accounting for RADIUS connections. For more information about using the **aaa accounting** command, refer to the “Specify RADIUS Accounting” section on page 4-20.

This section describes how to set up RADIUS for authentication, authorization, and accounting on your network, and includes the following sections:

- Configure Router to RADIUS Server Communication, page 4-17
- Configure Router to Use Vendor-Specific RADIUS Attributes, page 4-17
- Configure Router for Vendor-Proprietary RADIUS Server Communication, page 4-18
- Configure Router to Query RADIUS Server for Static Routes and IP Addresses, page 4-19
- Configure Router to Expand Network Cisco AS5800 Port Information, page 4-19
- Specify RADIUS Authentication, page 4-20
- Specify RADIUS Authorization, page 4-20
- Specify RADIUS Accounting, page 4-20

## Configure Router to RADIUS Server Communication

The RADIUS host is normally a multi-user system running RADIUS server software from Livingston, Merit, Microsoft, or another software provider. A RADIUS server and a Cisco router use a shared secret text string to encrypt passwords and exchange responses.

To configure RADIUS to use the AAA security commands, you must specify the host running the RADIUS server daemon, and a secret text string that it shares with the router. Use the `radius-server` commands to specify the RADIUS server host and a secret text string.

To specify a RADIUS server host and shared secret text string, perform the following tasks in global configuration mode:

- Specify the IP address or host name of the remote RADIUS server host, and assign authentication and accounting destination port numbers.

```
radius-server host {hostname | ip-address}
[auth-port port-number] [acct-port port-number]
```

- Specify the shared secret text string used between the router and the RADIUS server.

```
radius-server key string
```

To customize communication between the router and the RADIUS server, use the following optional `radius-server` global configuration commands:

- Specify the number of times the router transmits each RADIUS request to the server before giving up (default is three).

```
radius-server retransmit retries
```

- Specify the number of seconds a router waits for a reply to a RADIUS request before retransmitting the request.

```
radius-server timeout seconds
```

- Specify the number of minutes a RADIUS server, which is not responding to authentication requests, is passed over by requests for RADIUS authentication.

```
radius-server deadtime minutes
```

## Configure Router to Use Vendor-Specific RADIUS Attributes

The Internet Engineering Task Force (IETF) draft standard specifies a method for communicating vendor-specific information between the network Cisco AS5800 and the RADIUS server, by using the vendor-specific attribute (Attribute 26). Vendor-specific attributes (VSAs) allow vendors to support their own extended attributes not suitable for general use. The Cisco RADIUS implementation supports one vendor-specific option using the format recommended in the specification. The Cisco vendor-ID is 9, and the supported option has vendor-type 1, which is named “cisco-avpair.” The value is a string of the format:

```
protocol : attribute sep value *
```

- “Protocol” is a value of the Cisco “protocol” attribute for a particular type of authorization.
- “Attribute” and “value” are an appropriate attribute/value (AV) pair defined in the Cisco TACACS+ specification
- “sep” is “=” for mandatory attributes and “\*” for optional attributes.

This allows the full set of features available for TACACS+ authorization to also be used for RADIUS.

For example, the following AV pair causes the Cisco “multiple named ip address pools” feature to be activated during IP authorization (during PPP’s IPCP address assignment).

```
cisco-avpair= "ip:addr-pool=first"
```

The following example causes a “NAS Prompt” user to have immediate access to EXEC commands.

```
cisco-avpair= "shell:priv-lvl=15"
```

Other vendors have their own vendor-IDs, options, and associated VSAs. For more information about vendor-IDs and VSAs, refer to the RADIUS specification RFC 2138, “Remote Authentication Dial-In User Service (RADIUS),” described in *How Does RADIUS Work?*, available online at <http://www.cisco.com/warp/public/707/32.html>

To configure the NAS to recognize and use VSAs, perform the following task in global configuration mode:

Enable the network Cisco AS5800 to recognize and use VSAs as defined by RADIUS IETF attribute 26.

```
radius-server vsa send [accounting|authentication]
```

For a complete list of RADIUS attributes or more information about vendor-specific Attribute 26, refer to the RADIUS Attributes appendix.

## Configure Router for Vendor-Proprietary RADIUS Server Communication

Although the IETF draft standard for RADIUS specifies a method for communicating vendor-specific information between the network Cisco AS5800 and the RADIUS server, some vendors have extended the RADIUS attribute set in a unique way. Cisco IOS software supports a subset of vendor-proprietary RADIUS attributes.

To configure RADIUS (whether vendor-proprietary or IETF draft-compliant), you must specify the host running the RADIUS server daemon and the secret text string it shares with the Cisco device. You specify the RADIUS host and secret text string by using the radius-server commands. To identify that the RADIUS server is using a vendor-proprietary implementation of RADIUS, use the radius-server host nonstandard command.

Vendor-proprietary attributes will not be supported unless you use the radius-server host non-standard command.

To specify a vendor-proprietary RADIUS server host and a shared secret text string, perform the following tasks in global configuration mode.

Specify the IP address or host name of the remote RADIUS server host and identify that it is using a vendor-proprietary implementation of RADIUS.

```
radius-server host {hostname |ip-address} non-standard
```

Specify the shared secret text string used between the router and the vendor-proprietary RADIUS server. The router and the RADIUS server use this text string to encrypt passwords and exchange responses.

```
radius-server key string
```

## Configure Router to Query RADIUS Server for Static Routes and IP Addresses

Some vendor-proprietary implementations of RADIUS let the user define static routes and IP pool definitions on the RADIUS server, instead of on each individual Cisco AS5800 in the network. Each network Cisco AS5800 then queries the RADIUS server for static route and IP pool information.

To have the Cisco AS5800 query the RADIUS server for static routes and IP pool definitions when the device first starts up, use the following commands in global configuration mode:

```
radius-server configure-nas
```



### Note

Because the **radius-server configure-nas** command is performed when the Cisco router starts up, it will not take effect until you enter a **copy running-config startup-config** command.

## Configure Router to Expand Network Cisco AS5800 Port Information

In some situations, PPP or login authentication occurs on an interface different from the interface on which the call itself comes in. For example, in a V.120 ISDN call, login or PPP authentication occurs on a virtual asynchronous interface “ttt” but the call itself occurs on one of the channels of the ISDN interface.

The **radius-server attribute nas-port extended** command configures RADIUS to expand the size of the NAS-Port attribute (RADIUS IETF Attribute 5) field to 32 bits. The upper 16 bits of the NAS-Port attribute display the type and number of the controlling interface; the lower 16 bits indicate the interface undergoing authentication.

To display expanded interface information in the NAS-Port attribute field, perform the following task in global configuration mode.

Expand the NAS-Port attribute size from 16 to 32 bits to display extended interface information.

```
radius-server attribute nas-port extended
```



### Note

This command replaces the deprecated **radius-server extended-portnames** command.

On platforms with multiple interfaces (ports) per slot, the Cisco RADIUS implementation will not provide a unique NAS-Port attribute that permits distinguishing between the interfaces. For example, if a dual PRI interface is in slot 1, calls on both Serial1/0:1 and Serial1/1:1 will appear as NAS-Port = 20101. This is due to the 16-bit field size limitation associated with RADIUS IETF NAS-port attribute. In this case, replace the NAS-port attribute with a vendor-specific attribute (RADIUS IETF Attribute 26). The Cisco vendor-ID is 9, and the Cisco-NAS-Port attribute is subtype 2. Vendor-specific attributes (VSAs) can be turned on by entering the **radius-server vsa send** command. The port information in this attribute is provided and configured using the **aaa nas port extended** command.

To replace the NAS-Port attribute with RADIUS IETF Attribute 26 and to display extended field information, use the following commands in global configuration mode.

Enable the network Cisco AS5800 to recognize and use vendor-specific attributes as defined by RADIUS IETF Attribute 26.

```
radius-server vsa send [accounting | authentication]
```

Expand the size of the VSA NAS-Port field from 16 to 32 bits to display extended interface information.

```
aaa nas-port extended
```

The standard NAS-Port attribute (RADIUS IETF Attribute 5) will continue to be sent. If you do not want this information to be sent, you can suppress it by using the **no radius-server attribute nas-port** command. When this command is configured, the standard NAS-Port attribute will no longer be sent.

## Specify RADIUS Authentication

After you have identified the RADIUS server and defined the RADIUS authentication key, you need to define method lists for RADIUS authentication. Because RADIUS authentication is facilitated through AAA, you need to enter the **aaa authentication** command, and specify RADIUS as the authentication method. For more information, refer to information on configuring authentication in the security configuration guide for your Cisco IOS release.

## Specify RADIUS Authorization

AAA authorization lets you set parameters that restrict users network access. Authorization using RADIUS provides one method for remote access control, including one-time authorization or authorization for each service, per-user account list and profile, user group support, and support of IP, IPX, ARA, and Telnet. Because RADIUS authorization is facilitated through AAA, you need to issue the **aaa authorization** command, specifying RADIUS as the authorization method.

## Specify RADIUS Accounting

The AAA accounting feature enables you to track the services users access and the amount of network resources they consume. Because RADIUS accounting is facilitated through AAA, you need to issue the **aaa accounting** command, specifying RADIUS as the accounting method.

## RADIUS Attributes

The network Cisco AS5800 monitors the RADIUS authorization and accounting functions defined by RADIUS attributes in each user-profile.

## Vendor-Proprietary RADIUS Attributes

An Internet Engineering Task Force (IETF) draft standard for RADIUS specifies a method for communicating vendor-proprietary information between the network Cisco AS5800 and the RADIUS server. Some vendors, nevertheless, have extended the RADIUS attribute set in a unique way. Cisco IOS software supports a subset of vendor-proprietary RADIUS attributes.

## RADIUS Configuration Examples

RADIUS configuration examples in this section include the following:

- RADIUS Authentication and Authorization Example, page 4-21
- RADIUS Authentication, Authorization, and Accounting Example, page 4-21
- Vendor-Proprietary RADIUS Configuration Example, page 4-22

## RADIUS Authentication and Authorization Example

The following example shows a router configuration to authenticate and authorize using RADIUS.

```
aaa authentication login use-radius radius local
aaa authentication ppp user-radius if-needed radius
aaa authorization exec radius
aaa authorization network radius
```

These RADIUS authentication and authorization configuration commands are defined as follows:

- The **aaa authentication login use-radius radius local** command configures the router to use RADIUS for authentication at the login prompt. If RADIUS returns an error, the user is authenticated using the local database. In this example, use-radius is the name of the method list, which specifies RADIUS and then local authentication.
- The **aaa authentication ppp user-radius if-needed radius** command configures the Cisco IOS software to use RADIUS authentication for lines using Point-to-Point Protocol (PPP) with CHAP or PAP if the user has not already been authorized. If the EXEC facility has authenticated the user, RADIUS authentication is not performed. In this example, user-radius is the name of the method list defining RADIUS as the if-needed authentication method.
- The **aaa authorization exec radius** command sets the RADIUS information that is used for EXEC authorization, autocommads, and access lists.
- The **aaa authorization network radius** command sets RADIUS for network authorization, address assignment, and access lists.

## RADIUS Authentication, Authorization, and Accounting Example

The following sample is a general configuration using RADIUS with the AAA command set.

```
radius-server host 123.45.1.2
radius-server key myRaDiUSpassWoRd
username root password ALongPassword
aaa authentication ppp dialins radius local
aaa authorization network radius local
aaa accounting network start-stop radius
aaa authentication login admins local
aaa authorization exec local
line 1 16
  autoselect ppp
  autoselect during-login
  login authentication admins
  modem ri-is-cd
interface group-async 1
  encaps ppp
  ppp authentication pap dialins
```

The lines in this sample RADIUS authentication, authorization, and accounting configuration are defined as follows:

- **radius-server host** defines the IP address of the RADIUS server host.
- **radius-server key** defines the shared secret text string between the network Cisco AS5800 and the RADIUS server host.
- **aaa authentication ppp dialins radius local** defines the authentication method list “dialins,” which specifies that RADIUS authentication, then (if the RADIUS server does not respond) local authentication will be used on serial lines using PPP.
- **ppp authentication pap dialins** applies the “dialins” method list to the lines specified.

- **aaa authorization network radius local** is used to assign an address and other network parameters to the RADIUS user.
- **aaa accounting network start-stop radius** tracks PPP usage.
- **aaa authentication login admins local** defines another method list, “admins,” for login authentication.
- **login authentication admins** applies the “admins” method list for login authentication.

### Vendor-Proprietary RADIUS Configuration Example

The following example is a general configuration using vendor-proprietary RADIUS with the AAA command set.

```
radius-server host alcatraz non-standard
radius-server key myRaDiUSpassWoRd
radius-server configure-nas
username root password ALongPassword
aaa authentication ppp dialins radius local
aaa authorization network radius local
aaa accounting network start-stop radius
aaa authentication login admins local
aaa authorization exec local
line 1 16
autoselect ppp
autoselect during-login
login authentication admins
modem ri-is-cd
interface group-async 1
encaps ppp
ppp authentication pap dialins
```

The lines in this example RADIUS AAA configurations are defined as follows:

- **radius-server host non-standard** defines the name of the RADIUS server host, and identifies that this RADIUS host uses a vendor-proprietary version of RADIUS.
- **radius-server key** defines the shared secret text string between the network Cisco AS5800 and the RADIUS server host.
- **radius-server configure-nas** defines that the Cisco AS5800 will query the RADIUS server for static routes and IP pool definitions when the device first starts up.
- **aaa authentication ppp dialins radius local** defines the authentication method list “dialins,” which specifies that RADIUS authentication, then (if the RADIUS server does not respond) local authentication will be used on serial lines using PPP.
- **ppp authentication pap dialins** applies the “dialins” method list to the lines specified.
- **aaa authorization network radius local** is used to assign an address and other network parameters to the RADIUS user.
- **aaa accounting network start-stop radius** tracks PPP usage.
- **aaa authentication login admins local** defines another method list, “admins,” for login authentication.
- **login authentication admins** applies the “admins” method list for login authentication.

## RADIUS Cisco IOS Software Support

The following Cisco IOS software support is available for RADIUS.

1. AAA commands
2. RADIUS commands
3. RADIUS & AAA debug commands

### AAA Commands

```
aaa new-model
aaa authentication login CONSOLE none
aaa authentication login RADIUS_LIST radius
aaa authentication login TAC_PLUS tacacs+ enable
aaa authentication login V.120 none
aaa authentication enable default enable tacacs+
aaa authentication ppp RADIUS_LIST if-needed radius
aaa authorization exec RADIUS_LIST radius if-authenticated
aaa authorization exec V.120 none
aaa authorization network default radius if-authenticated
aaa authorization network V.120 radius if-authenticated
aaa authorization network RADIUS_LIST if-authenticated radius
aaa accounting suppress null-username
aaa accounting delay-start
aaa accounting exec default start-stop radius
aaa accounting commands 0 default start-stop radius
aaa accounting network default start-stop radius
aaa accounting connection default start-stop radius
aaa accounting system default start-stop radius
aaa nas port extended
```

### RADIUS Commands

```
ip radius source-interface subinterface-name
radius-server configure-nas
radius-server dead-time minutes
radius-server extended-portnames (*deprecated)
radius-server attribute nas-port extended (old)
radius-server attribute nas-port format {a | b | c} (new)
radius-server host {hostname | ip} [auth-port port#] [acct-port port#]
radius-server host {hostname | ip} non-standard
radius-server host {hostname | ip} ignore
radius-server host {hostname | ip}
radius-server key {string}
radius-server retransmit retries
radius-server timeout seconds
```

### RADIUS & AAA Debug Commands

```
debug radius
debug aaa authorization
debug aaa authentication
debug aaa peruser
debug ppp negotiation
debug ppp authentication
debug isdn q931
```

## Configuring TACACS+

The following global configuration commands provide basic security and local database configuration.

- 
- Step 1** Enable the AAA access control modem that includes TACACS+.
- ```
5800-1(config)# aaa new-model
```
- Step 2** Enable AAA authentication method during login.
- ```
5800-1(config)# aaa authentication login default local
```
- Step 3** Enable AAA authentication method during login using a methods list.
- ```
5800-1(config)# aaa authentication login console none
```
- Step 4** Enable AAA authentication method for use on serial interfaces running PPP when TACACS+ is used.
- ```
5800-1(config)# aaa authentication ppp default if-needed local
```
- Step 5** Enter authorization for username and password.
- ```
5800-1(config)# username username password password
```
- 

## TACACS+ Authentication

Use the AAA facility to authenticate users with either a local or remote security database. For more information about a local and remote security database, refer to the “Local and Remote Server Authentication” section on page 4-13.

Whether you maintain a local or remote security database, or use TACACS+ or RADIUS authentication and authorization, the process of configuring the Cisco AS5800 for these different databases and protocols is similar. The basic process of configuring the Cisco IOS software for authentication requires the following tasks:

1. Securing Access to Privileged EXEC and Configuration Mode
2. Communicating Between the Access and Security Servers
3. Enabling AAA Globally
4. Defining Authentication Method Lists
  - Issue the aaa authentication Command, page 4-30
  - Specify Protocol or Login Authentication, page 4-30
  - Identify a List Name, page 4-30
  - Specify the Authentication Method, page 4-31
  - Populate the Local Username Database if Necessary, page 4-32
5. Applying Authentication Method Lists, page 4-33

## Securing Access to Privileged EXEC and Configuration Mode

The first step is to secure access to privileged EXEC (enable) mode. Enable mode provides access to configuration mode, which enables any type of configuration change to the Cisco AS5800. To secure privileged EXEC mode access, use one of the following commands.

- The **enable password** *password* command requires that network administrators enter a password to access privileged EXEC mode. Do not provide access to users who are not administrators.
- The **enable secret** *password* command specifies a secret password that is encrypted, so that the password cannot be read when crossing a network. After you issue this command, the encryption cannot be reversed.

The enable secret password takes precedence over the enable password when it exists. The same password cannot be used for both commands. You can view the encrypted version of the enable secret password using the **show running-config** or **show startup-config** commands. (The encrypted version of the password is noted with \* in the following example.)

```
5800-1(config)# show running-config
Using 1899 out of 126968 bytes
!
Version x AA
.
.
.
!
hostname 5800-1
!
enable secret 5 $1$60L4$X2JY0woDc0.kqa11o0/w8/*
.
.
.
```



### Note

For more information about the enable password and enable secret commands and their complete syntax, refer to the security command reference for your Cisco IOS release in the Cisco IOS configuration guides and command references documentation.



### Caution

If you use the **enable secret** command and specify an encryption type, you *must* enter the *encrypted version* of a specific password. Do not enter the cleartext version of the password after specifying an encryption type, otherwise you will be locked out of privileged EXEC (enable) mode. To regain access to privileged EXEC mode, erase the contents of NVRAM and your entire configuration, and reconfigure the Cisco AS5800.



### Note

The enable secret password overrides the enable password.

The following global configuration commands provide an encrypted password using **enable secret**.

**Step 1** Enter the cleartext password used to gain access to privileged EXEC mode. Do not specify an encryption type.

```
5800-1(config)# enable secret password
5800-1(config)#
```

**Step 2** Type the **exit** command to exit out of global configuration mode.

```
5800-1(config)# exit
5800-1#
```

**Step 3** Enter the **show running-config** command to view the encrypted version of the cleartext password that was entered in Step 1. The encrypted password is noted with **\*\***.

```
5800-1# show running-config
Building configuration...

Current configuration:
!
version x AA
! some of the configuration skipped
enable secret 5 $1$h7dd$VTNs4.BAfQMUU0Lrvw6570**
! the rest of the configuration skipped
```



**Note** Encryption type **5** is the only valid encryption type for enable secret.

**Step 4** Type **Ctrl-Z** to return to privileged EXEC mode.

```
5800-1(config)# Ctrl-Z
5800-1#
```

**Step 5** Save changes.

```
5800-1# copy running-config startup-config
```

You can also specify additional protection for privileged EXEC mode, including the following:

- Privilege levels for Cisco IOS software commands
- Privileged EXEC passwords for different privilege levels
- Privilege levels for specific lines on the Cisco AS5800
- Encrypt passwords using **service password-encryption**

For more information about these security tools, refer to the security configuration guide for your Cisco IOS release in the Cisco IOS configuration guides and command references documentation.

## Communicating Between the Access and Security Servers

This section describes the Cisco IOS software commands that enable the Cisco AS5800 to communicate with a security server. This procedure is similar for communicating with TACACS+ and RADIUS servers, and the following sections describe the process.

If you are using a remote security server for authentication and authorization, you must configure the security server before performing the tasks described in this section. TACACS+ Security Examples, page 4-38 shows typical TACACS+ and RADIUS server entries corresponding to the Cisco AS5800 security configurations.

## Communicating with a TACACS+ Server

The following global configuration commands enable communication between the TACACS+ security (database) server and the Cisco AS5800.

- 
- Step 1** Specify the IP address or the host name of the remote TACACS+ server host. This host is typically a UNIX or NT system running TACACS+ software.
- ```
5800-1(config)# tacacs-server host {hostname | ip-address}
```
- Step 2** Specify a shared secret text string used between the Cisco AS5800 and the TACACS+ server. The Cisco AS5800 and TACACS+ server use this text string to encrypt passwords and exchange responses.
- ```
5800-1(config)# tacacs-server key shared-secret-text-string
```
- Step 3** Type **Ctrl-Z** to return to privileged EXEC mode.
- ```
5800-1(config)# Ctrl-Z
5800-1#
```
- Step 4** Save your changes when ready.
- ```
5800-1# copy running-config startup-config
```
- 

For example, to enable the remote TACACS+ server to communicate with the Cisco AS5800, enter the commands as follows:

```
5800-1# configure terminal
5800-1(config)# tacacs-server host alcatraz
5800-1(config)# tacacs-server key abra2cad
```

The host name of the TACACS+ server in the previous example is alcatraz. The key in the previous example (abra2cad) is the encryption key shared between the TACACS+ server and the Cisco AS5800. Substitute your own TACACS+ server host name and password for those shown.

For more information about these commands, refer to the security command reference for your Cisco IOS release, which is part of the Cisco IOS configuration guides and command references documentation.

## Configuring Authentication on a TACACS+ Server

On most TACACS+ security servers, there are three ways to authenticate a user for login:

- Include a cleartext (DES) password for a user or for a users group (each user can belong to only one group). Note that CHAP and global user authentication must be specified in cleartext.

The following is the configuration for global authentication:

```
user = birdman {global = cleartext "birdman global password"}
```

To assign different passwords for CHAP, and a normal login, you must enter a string for each user. Each string must specify the security protocols, state whether the password is cleartext, and specify if the authentication is performed with a DES card. The following example shows a user `aaaa`, who has authentication configured for CHAP and login. The users CHAP password, "chap password," is shown in cleartext and the login password has been encrypted.

```
user = aaaa
  chap = cleartext "chap password"
  login = des XQj4892fjk}
```

- Use password (5) files instead of entering the password into the configuration file directly.

The default authentication is to deny authentication. You can change this at the top level of the configuration file to have the default use `passwd(5)` file, by issuing the following command:

```
default authentication = /etc/passwd
```

- Authenticate using an `s/key`. If you have built and linked in an `s/key` library and compiled TACACS+ to use the `s/key`, you can specify that a user be authenticated using the `s/key`, as shown in the following example:

```
user= bbbb {login = skey}
```

On the Cisco AS5800, configure authentication on all lines including the VTY and Console lines by entering the following commands:

```
5800-1# configure terminal
5800-1(config)# aaa new-model
5800-1(config)# aaa authentication login default tacacs+ enable
```



### Caution

When you issue the `aaa authentication login default tacacs+ enable` command, you are specifying that if your TACACS+ server fails to respond (because it is set up incorrectly), you can log in to the Cisco AS5800 by using your enable password. If you do not have an enable password set on the Cisco AS5800, you will not be able to log in until you have a functioning TACACS+ daemon configured with user names and passwords. The enable password in this case is a last-resort authentication method. You can also specify `none` as the last-resort method, which means that no authentication is required if all other methods have failed.

## Enabling AAA Globally

To use the AAA security facility in the Cisco IOS software, you must issue the **aaa new-model** command from global configuration mode.

When you issue the **aaa new-model** command, all lines on the Cisco AS5800 receive the implicit **login authentication default** method list, and all interfaces with PPP enabled have an implicit **ppp authentication pap default** method list applied.



### Caution

If you authenticate users by a security server, do not inadvertently lock yourself out of the Cisco AS5800 ports after you issue the **aaa new-model** command. Enter line configuration mode and issue the **aaa authentication login default tacacs+ enable** global configuration command. This command specifies that if your TACACS+ (or RADIUS) server is not functioning properly, you can enter your enable password to log in to the Cisco AS5800. In general, verify that you have a last-resort access method before you are certain that your security server is set up and functioning properly. For more information about the **aaa authentication** command, refer to the “Defining Authentication Method Lists” section on page 4-29.



### Note

Cisco recommends that you use CHAP authentication with PPP, rather than PAP. CHAP passwords are encrypted when they cross the network, whereas PAP passwords are cleartext when they cross the network. The Cisco IOS software selects PAP as the default, so you must manually select CHAP. The process for specifying CHAP is described in Applying Authentication Method Lists, page 4-33.

```
5800-1# configure terminal
5800-1(config)# aaa new-model
```

## Defining Authentication Method Lists

After you enable AAA globally on the Cisco AS5800, you need to define authentication method lists, which you then apply to lines and interfaces. These authentication method lists are security profiles that indicate the protocol (PPP) or login and authentication method (TACACS+, RADIUS, or local authentication).

To define an authentication method list:

1. Issue the **aaa authentication** command.
2. Specify protocol (PPP) or login authentication.
3. Identify a list name or **default**. A list name is any alphanumeric string you choose. You assign different authentication methods to different named lists.
4. Specify the authentication method. You can specify multiple methods, such as **tacacs+**, followed by **local** if a TACACS+ server is not available on the network.
5. Populate the local username database if you specified **local** as the authentication method (or one of the authentication methods). To use a local username database, you must issue the **username** global configuration command. Refer to the “Populate the Local Username Database if Necessary” section on page 4-32.

After defining these authentication method lists, apply them to your interfaces (synchronous or asynchronous) configured for PPP.

Refer to the “Applying Authentication Method Lists” section on page 4-33 for information about applying these lists.

### Issue the `aaa authentication` Command

To define an authentication method list, enter the **aaa authentication** global configuration command, as shown in the following example:

```
5800-1# configure terminal
5800-1(config)# aaa authentication
```

### Specify Protocol or Login Authentication

After you enter **aaa authentication**, you must specify one of the following dial-in protocols as applicable for your network:

- If you are enabling dial-in PPP access, specify **ppp**.
- If you are enabling users to connect to the EXEC facility, specify **login**.

You can specify only one dial-in protocol per authentication method list; however, you can create multiple authentication method lists with each of these options. You must give each list a different name, as described in Identify a List Name, page 4-30.

If you specify the **ppp** option, the default authentication method for PPP is PAP. For greater security, specify CHAP. The full command is **aaa authentication ppp chap**.

For example, if you specify PPP authentication, the configuration looks like this:

```
5800-1# configure terminal
5800-1(config)# aaa authentication ppp
```

### Identify a List Name

A list name identifies each authentication list. You can choose either to use the keyword **default**, or choose any other name that describes the authentication list. For example, you name it `ppp-radius` if you intend to apply it to interfaces configured for PPP and RADIUS authentication. The list name can be any alphanumeric string. Use **default** as the list name for most lines and interfaces, and use different names on an exception basis.

You can create different authentication method lists and apply them to lines and interfaces selectively. You can even create a named authentication method list that you do not apply to a line or interface, but which you intend to apply at some later point, such as when you deploy a new log-in method for users.

After you define a list name, you must identify additional security attributes (such as local authentication versus TACACS+ or RADIUS).

In the following example, the default authentication method list for PPP dial-in clients uses the local security database:

```
5800-1# configure terminal
5800-1(config)# aaa authentication ppp default
```

In the following example, the PPP authentication method list name is `insecure`:

```
5800-1# configure terminal
5800-1(config)# aaa authentication ppp insecure
```

In the following example, the login authentication method list name is `deveng`:

```
5800-1# configure terminal
5800-1(config)# aaa authentication login deveng
```

## Specify the Authentication Method

After you identify a list name, you must specify an authentication method to identify how users will be authenticated. Authentication methods are defined with optional keywords in the **aaa authentication** command.

The following global configuration commands configure authentication methods for PPP.

---

**Step 1** Configure for AAA.

```
5800-1(config)# aaa new-model
```

**Step 2** Create a local authentication list. Methods include **if-needed**, **krb5**, **local**, **none**, **radius**, **tacacs+**.

```
5800-1(config)# aaa authentication ppp {default | list-name} method1 [method2]
```

**Step 3** Apply the authentication list to a line or set of lines.

```
5800-1(config)# ppp authentication {chap | pap | chap pap | pap chap} [if-needed]
{default | list-name} [callin]
```

**Step 4** Type **Ctrl-Z** to return to privileged EXEC mode.

```
5800-1(config)# Ctrl-Z
5800-1#
```

**Step 5** Save your changes when ready.

```
5800-1# copy running-config startup-config
```

---

The keyword *list-name* is any character string used to name the list you are creating. The *keyword* method refers to the actual method the authentication algorithm tries. The additional methods of authentication are used only if the previous method returns an error, not if it fails. To specify that the authentication should succeed even if all methods return an error, specify **none** as the final method in the command line.



### Timesaver

---

If you are not sure whether you should use TACACS+ or RADIUS, consider the following: TACACS+ encrypts the entire payload of packets passed across the network, whereas RADIUS only encrypts the password when it crosses the network. TACACS+ can query the security server multiple times, whereas a RADIUS server gives one response only and is not as flexible regarding per-user authentication and authorization attempts.

---

You can specify multiple authentication methods for each authentication list. The following authentication method example for PPP first queries a TACACS+ server, then a RADIUS server, then the local security database. Multiple authentication methods can be useful if you have multiple types of security servers on the network, and one or more types of security server do not respond.

```
5800-1(config)# aaa authentication ppp testbed tacacs+ radius local
```

If you specify more than one authentication method and the first method (TACACS+ in the previous example) is not available, the Cisco IOS software attempts to authenticate using the next method (such as RADIUS). If, in the previous example, the RADIUS server has no information about the user, or if no RADIUS server can be found, the user is authenticated using the local username database that was populated with the **username** command.

If authentication *fails* using the first method listed, the Cisco IOS software does *not* permit access. It does not attempt to authenticate using the subsequent security methods if the user entered the incorrect password.

### Populate the Local Username Database if Necessary

If you specify **local** as the security method, you must specify username profiles for each user who might log in. An example of specifying local authentication is as follows:

```
5800-1(config)# aaa authentication login deveng local
```

This command specifies that anytime a user attempts to log in to a line on an Cisco AS5800, the Cisco IOS software checks the username database. To create a local username database, define username profiles using the **username** global configuration command.

The following example shows how to use the **username** command and password:

```
5800-1(config)# username username password password
```

The **show running-config** command shows the encrypted version of the password, as follows:

```
5800-1# show running-config
Building configuration...

Current configuration:
!
version x AA
! most of config omitted
username xxx password 7 0215055500070C294D
```



#### Note

The Cisco IOS software adds the encryption type of 7 automatically for passwords. If you were to manually enter the number 7 to represent an encryption type, you must follow the 7 with the *encrypted* version of the password. If you specify the number 7, enter a cleartext password, the user will not have access to the line, interface, or the network the user is trying to access, and you must reconfigure the users authentication profile.

## Authentication Method List Examples

This section includes authentication method list examples for:

- Users Logging In to the Cisco AS5800
- Users Dialing In Using PPP

### Users Logging In to the Cisco AS5800

The following example creates a local authentication list for users logging in to any line on the Cisco AS5800:

```
5800-1(config)# aaa authentication login default local
```

The following example specifies login authentication using RADIUS (the RADIUS daemon is polled for authentication profiles):

```
5800-1(config)# aaa authentication login default radius
```

The following example specifies login authentication using TACACS+ (the TACACS+ daemon is polled for authentication profiles):

```
5800-1(config)# aaa authentication login default tacacs+
```

## Users Dialing In Using PPP

The following example creates a TACACS+ authentication list for users connecting to interfaces configured for dial-in using PPP. The name of the list is **marketing**. This example specifies that a remote TACACS+ daemon be used as the security database. If this security database is not available, the Cisco IOS software then polls the RADIUS daemon. Users are not authenticated if they are already authenticated on a TTY line.

```
5800-1(config)# aaa authentication ppp marketing if-needed tacacs+ radius
```

In this example, **default** can be substituted for **marketing** if the administrator wants this list to be the default list.

## Applying Authentication Method Lists

As described in Defining Authentication Method Lists, page 4-29, the **aaa authentication** global configuration command creates authentication method lists or profiles. You apply these authentication method lists to lines or interfaces by issuing the **login authentication** or **ppp authentication** command, as described in Table 4-5.

**Table 4-5** Line and Interface Authentication Method Lists

| Interface and Line Command             | Action                                         | Port to Which List Is Applied | Corresponding Global Configuration Command |
|----------------------------------------|------------------------------------------------|-------------------------------|--------------------------------------------|
| <b>login authentication</b>            | Logs directly in to the Cisco AS5800           | Console port or VTY lines     | <b>aaa authentication login</b>            |
| <b>ppp authentication</b> <sup>1</sup> | Uses PPP to access IP or IPX network resources | Interface                     | <b>aaa authentication ppp</b>              |

1. If you issued the **ppp authentication** command, you must specify either CHAP or PAP authentication. PAP is enabled by default, but Cisco recommends that you use CHAP because CHAP is more secure. For more information, refer to the security configuration guide for your Cisco IOS release, which is part of the Cisco IOS configuration guides and command references documentation.

You can create more than one authentication list or profile for login and protocol authentication and apply them to different lines or interfaces. The following examples show the line or interface authentication commands that correspond to the **aaa authentication** global configuration command.

### Login Authentication Example

The following example shows the default log-in authentication list applied to the console port and the default virtual terminal (VTY) lines on the Cisco AS5800:

```
5800-1(config)# aaa authentication login default local
5800-1(config)# line console 0
5800-1(config-line)# login authentication default
5800-1(config-line)# line vty 0 69
5800-1(config-line)# login authentication default
```

In the following example, the login authentication list named `rtp2-office`, which uses RADIUS authentication, is created. It is applied to all 54 lines on an AS5800 configured with a channelized T1 PRI card, including the console (CTY) port, the 48 physical asynchronous (TTY) lines, the auxiliary (AUX) port, and 69 virtual terminal (VTY) lines:

```
5800-1(config)# aaa authentication login rtp2-office radius
5800-1(config)# line 0 118
5800-1(config-line)# login authentication rtp2-office
```

The following sample output shows lines and their status on the Cisco AS5800.

```
5800-1# show line
Tty Typ Tx/Rx A Modem Roty AccO AccI Uses Noise Overruns
* 0 CTY - - - - - 0 0 0/0
I 1 TTY 115200/115200 - inout - - - 0 0 0/0
I 2 TTY 115200/115200 - inout - - - 0 0 0/0
...
I 48 TTY 115200/115200 - inout - - - 0 0 0/0
49 AUX 9600/9600 - - - - - 0 0 0/0
50 VTY - - - - - 0 0 0/0
51 VTY - - - - - 0 0 0/0
52 VTY - - - - - 0 0 0/0
53 VTY - - - - - 0 0 0/0
54 VTY - - - - - 0 0 0/0
```

## PPP Authentication Example

The following example creates the PPP authentication list `marketing`, which uses TACACS+ and RADIUS authentication. The list `marketing` requires authentication only if the user has not been authenticated on another line. It is then applied to asynchronous lines 1-48 on a Cisco AS5800 and uses CHAP authentication, instead of the default of PAP.

```
5800-1(config)# aaa authentication ppp marketing if-needed tacacs+ radius
5800-1(config)# line shelf/slot/1 shelf/slot/48
5800-1(config-line)# ppp authentication chap marketing
```

## TACACS+ Authorization

You can configure the Cisco AS5800 to restrict user access to the network so that users can only perform certain functions after successful authentication. As with authentication, authorization can be used with either a local or remote security database. This guide describes only remote security server authorization.

A typical configuration often uses the EXEC facility and network authorization. EXEC authorization restricts access to the EXEC, and network authorization restricts access to network services, including PPP.

Authorization must be configured on both the Cisco AS5800 and the security daemon. The default authorization is different on the Cisco AS5800 and the security server:

- By default, the Cisco AS5800 *permits* access for every user until you configure the system to make authorization requests to the daemon.
- By default, the daemon *denies* authorization of anything that is not explicitly permitted. Therefore, you have to explicitly allow all per-user attributes on the security server.

**Timesaver**

If authentication has not been set up for a user, per-user authorization attributes are not enabled for that user. That is, if you want a user to obtain authorization before gaining access to network resources, you must first require that the user provide authentication. For example, if you want to specify the **aaa authorization network tacacs+** (or **radius**) command, you must first specify the **aaa authentication {ppp} default if-needed tacacs+** (or **radius**) command.

## Configuring Authorization on the Security Server

You typically have the three following methods for configuring default authorization on the security server:

- To override the default denial or authorization from a nonexistent user, specify authorization at the top level of the configuration file:

```
default authorization = permit
```

- At the user level, inside the braces of the user declaration, the default for a user who does not have a service or command explicitly authorized is to deny that service or command. To permit it:

```
default service = permit
```

- At the service authorization level, arguments are processed according to the following algorithm: For each AV pair sent from the Cisco AS5800, the following process occurs:
  - a. If the AV pair from the Cisco AS5800 is mandatory, look for an exact match in the daemons mandatory list. If found, add the AV pair to the output.
  - b. If an exact match does not exist, look in the daemons optional list for the first attribute match. If found, add the Cisco AS5800 AV pair to the output.
  - c. If no attribute match exists, deny the command if the default is to deny. If the default is permit, add the Cisco AS5800 AV pair to the output.
  - d. If the AV pair from the Cisco AS5800 is optional, look for an exact attribute, value match in the mandatory list. If found, add the daemons AV pair to output.
  - e. If not found, look for the first attribute match in the mandatory list. If found, add daemons AV pair to output.
  - f. If no mandatory match exists, look for an exact attribute, value pair match among the daemons optional AV pairs. If found, add the daemons matching AV pair to the output.
  - g. If no exact match exists, locate the first attribute match among the daemons optional AV pairs. If found, add the daemons matching AV pair to the output.
  - h. If no match is found, delete the AV pair if default is deny. If the default is permit, add the Cisco AS5800 AV pair to the output.
  - i. If there is no attribute match already in the output list after all AV pairs have been processed for each mandatory daemon AV pair, add the AV pair. Add only one AV pair for each mandatory attribute.

## Configuring Authorization (Network or EXEC)

The following global configuration commands configure network and EXEC authorization.

---

**Step 1** Prevents unauthorized users from accessing network resources.

```
5800-1(config)# aaa authorization network
```

**Step 2** Prevents users from logging in to the privileged EXEC facility.

```
5800-1(config)# aaa authorization exec
```

**Step 3** Type **Ctrl-Z** to return to privileged EXEC mode.

```
5800-1(config)# Ctrl-Z
5800-1#
```

**Step 4** Save your changes when ready.

```
5800-1# copy running-config startup-config
```



**Note**

---

You can also require authorization before a user can issue specific commands by using the **aaa authorization** command. For more information, refer to the security configuration guide for your Cisco IOS release, which is part of the Cisco IOS configuration guides and command references.

---

## Specifying an Authorization Method

Authorization methods are defined as optional keywords in the **aaa authorization** command. The following global configuration command configure both network and EXEC AAA authorization. Table 4-5 defines authorization methods.

---

**Step 1** Prevents unauthorized users from accessing network resources.

```
5800-1(config)# aaa authorization {if-authenticated | local | none | radius | tacacs+}
```

**Step 2** Type **Ctrl-Z** to return to privileged EXEC mode.

```
5800-1(config)# Ctrl-Z
5800-1#
```

**Step 3** Save your changes when ready.

```
5800-1# copy running-config startup-config
```

---

**Table 4-6 Authorization Methods**

| <b>Authorization Methods</b> | <b>Purpose</b>                                                                                                                                                                                                                                               |
|------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>if-authenticated</b>      | User is authorized if already authenticated.                                                                                                                                                                                                                 |
| <b>local</b>                 | Uses the local database for authorization. The local database is created using the <b>username privilege</b> command to assign users to a privilege level from 0 to 15, and the <b>privilege level</b> command to assign commands to these different levels. |
| <b>none</b>                  | Authorization always succeeds.                                                                                                                                                                                                                               |
| <b>radius</b>                | Uses RADIUS authorization as defined on a RADIUS server.                                                                                                                                                                                                     |
| <b>tacacs+</b>               | Uses TACACS+ authorization as defined on a TACACS+ server.                                                                                                                                                                                                   |

## Specifying Authorization Parameters on a TACACS+ Server

When you configure authorization, you must ensure that the parameters established on the Cisco AS5800 correspond with those set on the TACACS+ server.

## Authorization Examples

The following example uses a TACACS+ server to authorize the use of network services, including PPP. If the TACACS+ server is not available or has no information about a user, no authorization is performed, and the user can use all network services.

```
5800-1(config)# aaa authorization network tacacs+ none
```

The following example permits the user to run the EXEC process if the user is authenticated. If the user is not authenticated, the Cisco IOS software defers to a RADIUS server for authorization information.

```
5800-1(config)# aaa authorization exec if-authenticated radius
```

The following example configures network authorization. If the TACACS+ server does not respond or has no information about the username being authorized, the RADIUS server is polled for authorization information for the user. If the RADIUS server does not respond, the user still can access all network resources without authorization requirements.

```
5800-1(config)# aaa authorization network tacacs+ radius none
```

## TACACS+ Security Examples

The following examples show complete security configuration components of a configuration file on a Cisco AS5800. Each example shows authentication and authorization.

### Local TACACS+ Security Example

The following sample configuration uses AAA to configure default authentication using a local security database on the Cisco AS5800. All lines and interfaces have the default authentication lists applied. Users **aaaa**, **bbbb**, and **cccc** have been assigned privilege level 7. This prevents them from issuing **ppp** and **slip** commands because these commands have been assigned to privilege level 8.

```

aaa new-model
aaa authentication login default local
aaa authentication arap default local
aaa authentication ppp default local
aaa authorization exec local
aaa authorization network local
aaa authorization
!
username aaaa privilege exec level 7 privilege network level 8 password 7 095E470B1110
username bbbb privilege network level 7 password 7 0215055500070C294D
username cccc privilege network level 7 password 7 095E4F10140A1916
!
privilege exec level 8 ppp
privilege exec level 8 slip

line console 0
login authentication default
!
line 2/2/0 2/2/47
interface Group-Async1
ppp authentication chap default
group-range 2/2/0 2/2/47

```

The following configuration displays the sign-on dialog from a remote PC:

```

atdt5551234
CONNECT 14400/ARQ/V32/LAPM/V42BIS
User Access Verification
Username: username
Password: password
5800-1> enable
Password: password
5800-1#

```

### TACACS+ Security Example for Login and PPP

The following example shows how to create and apply the following authentication lists:

- A TACACS+ server named AAA is polled for authentication information (so you do not need to define a local username database). The shared key between the Cisco AS5800 and the TACACS+ security server is 007.
- A login authentication list named rtp-office is created, then applied to the console port.
- A PPP authentication list named marketing is created, and applied to group async interface 0, which includes asynchronous interfaces 2/2/0 to 2/2/47.

**Note**

---

The authentication method lists used in this example use names other than **default**. However, you generally specify **default** as the list name for most lines and interfaces, and apply different named lists on an exception basis. These names are used only for illustrative purposes.

---

```
hostname 5800-1
!
tacacs-server host aaa
tacacs-server key 007
!
aaa authentication login rtp-office tacacs+
aaa authentication ppp marketing if-needed tacacs+
!
line console0
login authentication rtp-office
!
tacacs-server host aaa
tacacs-server key 007
!
aaa authentication login rtp-office tacacs+
aaa authentication ppp marketing if-needed tacacs+
!
line console0
login authentication rtp-office
!
interface group-async0
ppp authentication chap marketing
group-range 2/2/0 2/2/47
!
line 2/2/0 2/2/47
```

The following example shows how to create the following authentication lists:

- A RADIUS server named AAA is polled for authentication information (so you do not need to define a local username database). The shared key between the Cisco AS5800 and the RADIUS security server is 007.
- A login authentication list named fly is created, then applied to all lines that users can log in to, except the console port. In this example, the console port is physically secure and does not need password protection. The Cisco AS5800 is locked in a closet and secured behind a deadbolt lock.

- A PPP authentication list `itsme` is created, then applied to group `async` interface 6, that includes asynchronous interfaces `2/2/0` to `2/2/47`. The more secure CHAP authentication is used over PAP.

```
radius-server host aaa
radius-server key 007
!
privilege exec level 14 configure
privilege exec level 14 reload
privilege exec level 8 ppp
!
aaa authentication login fly radius
aaa authentication ppp itsme if-needed radius
aaa authorization network radius
aaa authorization exec radius
!
line 2/1/0 2/1/53
login authentication fly
!
interface group-async6
ppp authentication chap itsme
group-range 2/2/0 2/2/47
```