CHAPTER 8

Traffic Filtering and Security

Network security is a topic that grows in importance every day. With an increased reliance on business-to-business and business-to-consumer communication, secure Internet connectivity has become mission critical for many organizations. For these companies, an Internet security breach can cause embarrassment, loss of consumer confidence, and ultimately fiscal loss.

When it comes to network security, one of the most neglected network devices is the switch. Organizations often enforce strict security policies on routers and firewalls, yet fail to recognize the dangers of leaving a switch unsecured. The switch is the most accessible device in your network, and often, it unwittingly provides unauthorized access to your network.

NOTE It is a common misconception that most network security breaches occur from an external party (e.g., a hacker on the Internet). In fact, most breaches occur from an internal party (e.g., a disgruntled employee).

Review of the following topics can help you secure your switched network infrastructure.

- Securing management access
- Securing network access
- Traffic filtering

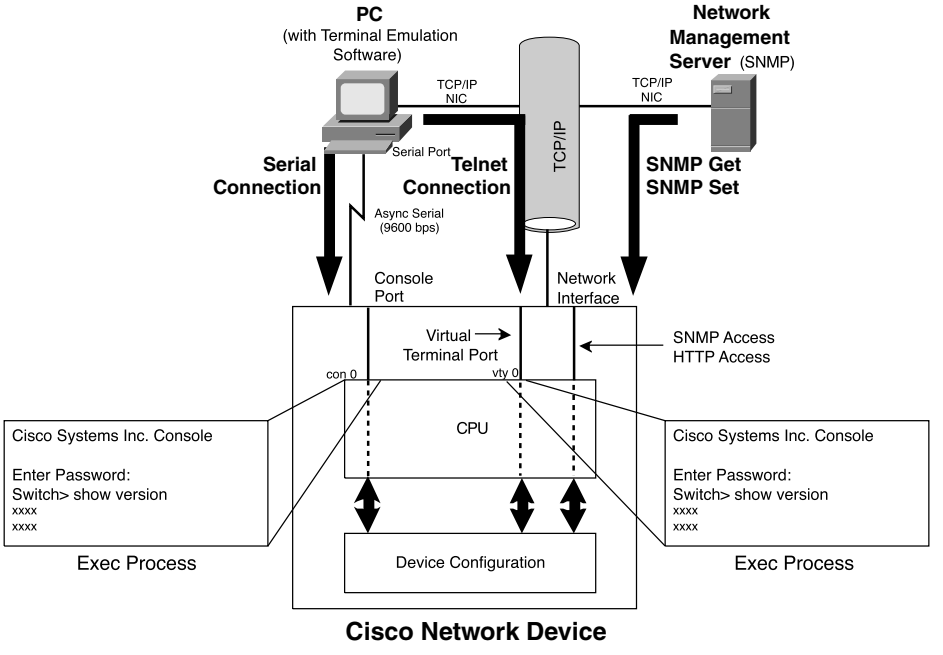
Securing Management Access

Cisco networking devices provide rich management access capabilities that give the network administrator powerful configuration and diagnostics tools.

The most common form of management access to a Cisco networking device is via an *EXEC session*. An EXEC session is similar to a UNIX shell and is accessible via Telnet, secure shell, or the console port. Cisco devices also support management via Simple

Network Management Protocol (SNMP) and Hypertext Transfer Protocol (HTTP). Figure 8-1 illustrates the various management interfaces and how they interact with a Cisco switch.

Figure 8-1 *Cisco Management Interfaces*



The first step to securing your switching infrastructure is to secure the switch's management interfaces. Next, you should implement techniques that improve the overall security of the switch.

The following switch security techniques are available:

- Configuring authentication, authorization, and accounting (AAA)
- Restricting management access
- Using secure management protocols
- Reducing other vulnerabilities

Configuring Authentication, Authorization, and Accounting (AAA)

The default authentication policy on a Cisco CatOS switch is extremely lax. Cisco IOS-based switches are slightly more secure by default, but the security of both platforms can be significantly improved. Table 8-1 shows the default authentication methods for accessing a CatOS and IOS switch.

Table 8-1 *Default Authentication Procedures*

Platform	Access Type	User EXEC Mode	Privileged EXEC Mode
CatOS	Console	None	Blank Password
	Telnet	Blank Password	Blank Password
IOS	Console	None	None
	Telnet	Password Required	Password Required

Table 8-1 represents the *access policy* for obtaining management access to the switch. Since default passwords should never be used in a production environment, the first thing you should do is configure passwords for all access methods (e.g., console, Telnet) and then configure an enable secret to protect privileged access.

You can further secure switch management access through the implementation of the techniques detailed in Table 8-2.

Table 8-2 *Techniques To Secure Switch Management Access*

Technique	Description
Local user authentication	Provides a per-user username and password, which can eliminate the need to share the enable secret, and adds username information to relevant log entries (e.g., configuration changes). The primary disadvantage of this technique is a lack of centralized account management. On CatOS, local user authentication is in CatOS 7.5.
Lockout parameters (CatOS only)	This feature disables access to a switch when a number of failed login attempts have occurred. This is meant to thwart brute force login attacks.
Privilege levels (IOS only)	Ranging from 0 to 15, 16 privilege levels exist. By default 1 is the user EXEC mode, and 15 is privileged EXEC mode. Commands can be assigned to each privilege level, which are then secured with a level specific enable secret.
Login banners (CatOS and IOS)	Login banners provide a means to communicate with anyone attempting to access a device. Typically these are used to inform visitors of their unwelcome status.
Session timeouts (CatOS and IOS)	Simply used to disconnect idle EXEC sessions.
Centralized AAA (CatOS and IOS)	Provides centralized user account management and accounting. Requires a TACACS+ or RADIUS server.

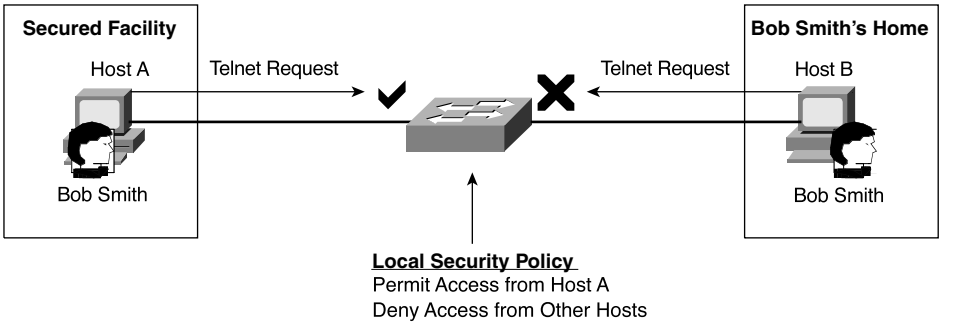
Restricting Management Access

In most networks, only a select handful of people need management access to switches. CatOS and Cisco IOS allow you to restrict which hosts can establish management sessions based on the source IP address.

NOTE Known as *host-based authentication*, this type of access control is extremely weak because any user on the allowed host could establish a management session. Host-based authentication should be used only to supplement user-based authentication mechanisms.

Figure 8-2 illustrates restricting management access.

Figure 8-2 *Restricting Management Access*



Through the use of permit lists on CatOS and access classes on IOS, management sessions can be controlled on a source IP address basis for the following protocols:

- Telnet
- Secure shell
- SNMP
- HTTP

Using Secure Management Protocols

In previous sections, we discussed secure access control mechanisms. Most of the time, management access is remote, which means that management communications are passed through the network. These communications could contain sensitive information, such as username/password combinations or device configuration information. If your management communications are transmitted in clear text, it is possible for other parties on

the network to eavesdrop on your management session, gleaning sensitive information such as a username/password pair. To circumvent this issue, you need to employ secure management protocols that protect the confidentiality of your management session.

Table 8-3 details the secure management protocols available on CatOS and IOS devices.

Table 8-3 *The Secure Management Protocols Available on CatOS and IOS Devices*

Protocol	Description
Secure shell	Provides encrypted Telnet-like terminal emulation to remote network devices. Secure shell client software and an SSH-enabled IOS or CatOS image is required.
SNMPv3	SNMPv3 greatly improves on the security of SNMP versions 1 and 2c by providing message confidentiality, authentication, and integrity. SNMPv3 is not in widespread use and device support is very limited.

Reducing Other Vulnerabilities

So far we have discussed switch access methods and protocols; now you can leverage a few other configuration tips to protect against some of the less common security vulnerabilities.

- **Password encryption**—Cisco IOS enables you to encrypt all passwords in the configuration file. This type of encryption is not secure and is meant only to prevent casual onlookers from learning passwords.

TIP Don't rely on password encryption used in conjunction with the standard enable password, because many tools available can decrypt the encrypted password.

- **Enable secret**—Cisco IOS can use two types of enable passwords, known as the enable password and enable secret. The enable password uses a weak algorithm that can easily be decrypted. The enable secret, however, uses MD5, a one-way encryption algorithm that greatly increases password security.
- **Disabling unnecessary services**—Various services are enabled by default that might not be required on your network. An example of this is the *Cisco Discovery Protocol* (*CDP*), which multicasts information about Cisco devices. Since CDP is a very valuable troubleshooting tool, it is common practice to disable CDP only on interfaces connecting to untrusted or insecure networks.

Securing Network Access

Once you have secured your switch, you are now ready to configure it to enforce your organization's security policy. Cisco Catalyst switches provide the security features aimed at securing network access found in Table 8-4.

Table 8-4 *Cisco Catalyst Switch Security Features Aimed at Securing Network Access*

Feature	Description
Port security	Binds a specific MAC address or group of addresses to a particular switch port. Configured on a per-port basis and disables the port if an unauthorized MAC address is seen.
VLAN membership policy server	Uses a central database to bind a MAC address to a specific VLAN. This awkward technology has many restrictions and does not enjoy widespread success.
802.1x	Based on the Extensible Authentication Protocol (EAP), 802.1x provides user-level authentication of devices wanting to connect to the network. RADIUS is used to authenticate users against a centrally managed user database.

Traffic Filtering

Traffic filtering has traditionally been used on routers and firewalls to enforce access control policies. Most traffic filtering is performed at Layer 3 and Layer 4; hence, traffic filtering on switches (traditionally being Layer 2 devices) is a relatively new practice. With the importance of security and quality of service, switches need extra intelligence to provide the features that enable end-to-end security and quality of service.

NOTE Support for these features varies by platform. For up to date feature support information, use the Feature Navigator at www.cisco.com/go/fn (CCO registration required).

Table 8-5 details the available traffic filtering features.

Table 8-5 *Available Traffic Filtering Features*

Feature	Description
Protocol filtering (CatOS only)	Filters Layer 2 frames based on Layer 3 protocol. Can explicitly permit or deny IP, IPX, and Group (includes AppleTalk and DECnet). Can automatically filter the protocols not in use on each switch port.

Table 8-5 *Available Traffic Filtering Features (Continued)*

Feature	Description
VLAN access control lists (Catalyst 6500 with CatOS only)	Applies IP- or MAC-based access control lists to a VLAN. The applied VLAN access control list (VACL) is used to filter traffic bridged within the VLAN, as well as traffic routed into the VLAN.
VLAN maps (Catalyst 3550 and 6500 with IOS)	Provides identical functionality to VACLs but with a different name.
Port-based access control lists (Catalyst 2950, 3550 and Cat6K with Supervisor 720)	On a Layer 2 port, Ccan filter inbound Layer 2 frames using MAC addresses as well as IP packets based on Layer 3 and Layer 4 information.
Routed access control lists	Identical to access control lists implemented on Cisco routers, these can be used on SVI and physical routed interfaces on Cisco Catalyst Layer 3 switches.
Private VLANs	Creates the idea of hierarchical VLANs with restricted Layer 2 connectivity. Promiscuous ports can communicate with any port in the VLAN, community ports can talk with any port in the same community and any promiscuous ports, and isolated ports can talk only with promiscuous ports.

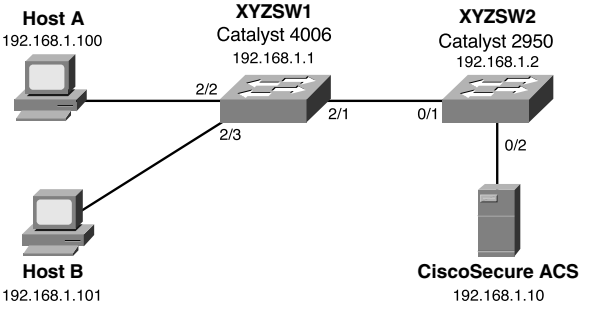
Scenario 8-1: Securing the Management Interface

In this scenario, you secure both a CatOS-based switch and an IOS-based switch. By securing each switch, you are not only reducing their vulnerabilities, but also increasing the security of the entire network.

Scenario Exercise

Figure 8-3 illustrates the scenario topology used for Scenarios 8-1, 8-2, and 8-3. Corporation XYZ requires their existing switches to be secured using best practices. They are also about to acquire a larger corporation and need to add new switches to the network. The Corporation XYZ CIO has specified that the current network must be secured, to ensure the new network maintains a tight security policy.

Figure 8-3 *Scenarios 8-1, 8-2, and 8-3 Topology*



A Catalyst 4006 (XYZSW1) provides access for Hosts A and B. Host A is a dedicated network administration workstation that is used to manage the network. Host B is a user's PC, and should not be allowed management access to any network devices. A Catalyst 2950 (XYZSW2) provides connectivity to the company servers, including a recently installed CiscoSecure asynchronous communications server (ACS). Both switches are interconnected by a single Fast Ethernet trunk.

Scenario Objectives

The scenario objectives are as follows:

- Configure local user-level authentication (Cisco IOS only)
- Configure a lockout policy (CatOS only)
- Configure a login banner
- Configure session timeouts
- Configure local authorization (Cisco IOS only)
- Restrict management access
- Enable SSH support (CatOS only)

Equipment Needed

The equipment needed is as follows:

- The workstations and servers in the diagram recommended set up and installed as per the scenario diagram
- One CatOS and one IOS switch

Command Syntax

This covers the following:

- Securing a CatOS Switch
- Securing a Cisco IOS Switch
- The following sections describe the commands used for each part of the scenario

CatOS Command Syntax

The following new CatOS commands are introduced in this scenario:

- **set authentication**
- **set banner motd**
- **set logout**

- **set ip permit**
- **set crypto key rsa**

The **set authentication** Command Syntax

Login authentication is configured using the **set authentication login** command. You can also control access to privileged configuration mode (enable mode) separately by using the **set authentication enable** command.

You can individually specify the maximum number of unsuccessful login or enable attempts using the following syntax:

```
set authentication {login | enable} attempt maximum_attempts [console | telnet]
```

Once the maximum number of attempts is reached, you can define a lockout policy by using the following syntax:

```
set authentication {login | enable} lockout time [console | telnet]
```

The time parameter is configurable between 30 and 600 seconds, with a value of 0 disabling any lockout; 0 is the default configuration.

NOTE Unlike a console lockout, which completely blocks console access, a Telnet lockout blocks only the IP address from which the login attempts failed.

The **set banner motd** Command Syntax

You configure a login banner on CatOS using the **set banner motd** command, as shown in Example 8-1.

Example 8-1 *Configuring a Banner on CatOS*

```
Switch (enable) set banner motd #
**********************************
* Unauthorized access prohibited *
**********************************
#
MOTD banner set
```

Notice the use of the # character as a delimiter, which allows you to enter a banner as free text until you terminate the input with the same delimiter. The delimiter can be any character, as long as it does not appear in the desired banner.

The **set logout** Command Syntax

The **set logout** command controls how long a session (e.g., a console or Telnet session) can remain idle before being disconnected by the system:

```
set logout timeout
```

The *timeout* parameter is specified in minutes and is configurable from 0 (no timeout) to 10000 minutes. The default setting is 20 minutes.

The **set ip permit** Command Syntax

The **set ip permit** command restricts management access for Telnet, SSH, and SNMP on CatOS. All restrictions are controlled by the **set ip permit** command. You must initially specify which hosts are to be permitted management access:

```
set ip permit ip-address [mask] [telnet | ssh | snmp | all]
```

You can specify network address ranges by configuring the optional *mask* parameter and you can specify different access policies based upon each management access protocol (e.g., Telnet or SNMP).

Once you have defined your permitted hosts, you then need to enable the permit list:

```
set ip permit enable [telnet | ssh | snmp | all]
```

You can selectively enable the permit list based upon management protocol, or you can enable all permit lists.

The **set crypto key rsa** Command Syntax

To enable SSH support, you must create a public/private key pair on the switch using the following syntax:

```
set crypto key rsa nbits
```

The argument *nbits* is used to specify the length of the key in bits; valid values are from 512 to 2048. Once this key pair has been created, you are able to connect to the switch using a SSH client.

Cisco IOS Command Syntax

The following new Cisco IOS commands are introduced in this scenario:

- The **username** command
- The **login local** command
- The **banner** command
- The **exec-timeout** command
- The **privilege** command
- The **access-class** command

The **username** Command and **login local** Command Syntax

When enabling user-level authentication, the first step is to create user accounts for each user that requires access to the switch. This is achieved by executing the **username** global configuration command:

```
username name password secret
```

Next you need to configure each management interface to use local authentication. This is achieved by executing the **login local** line configuration command, as shown in Example 8-2.

Example 8-2 *Enabling Local User-Level Authentication*

```
Switch(config)# line con 0
Switch(config-line)# login local
Switch(config-line)# line vty 0 4
Switch(config-line)# login local
```

In Example 8-2, both the console and vty ports are configured to use the local user account database to authenticate users.

The **banner** Command Syntax

You configure a login banner on Cisco IOS using the **banner motd** global configuration command, as shown in Example 8-3.

Example 8-3 *Configuring a Banner on Cisco IOS*

```
Switch(config)# banner motd #
**********************************
* Unauthorized access prohibited *
**********************************
#
```

Notice the use of the # character as a delimiter, which allows you to enter a banner as free text until you terminate the input with the same delimiter.

It is also possible to display other banners by using one of the arguments to the **banner** global configuration command listed in Table 8-6.

Table 8-6 *Arguments to the* **banner** *Global Configuration Command*

Argument	Description
exec	Displayed when an exec session is created
incoming	Displayed when a reverse Telnet session is established through a router
login	Displayed after the message of the day (MOTD) but before the username and password prompt

The **exec-timeout** Command Syntax

The **exec-timeout** command is used to control how long a session (e.g., a console or telnet session) can remain idle before being disconnected by the switch. The command is applied in line configuration mode as shown in Example 8-4.

Example 8-4 *Configuring Session Timeouts on Cisco IOS*

```
Switch(config)# line con 0
Switch(config-line)# exec-timeout 20 30
```

The first numeric parameter of the **exec-timeout** command specifies the number of minutes, while the second numeric parameter specifies the number of seconds. In Example 8-4, the console idle session timeout is set to 20 minutes and 30 seconds.

The **privilege** Command Syntax

The **privilege** global configuration mode command is used to define custom, local authorization levels for Cisco IOS commands. You can assign a particular command (or set of commands) to a particular privilege level using the following syntax:

privilege {**configure** | **exec** | **interface**} **level** *privilege-level command*

You must specify which configuration mode the command exists in (e.g., configure, exec, interface); indicate the desired privilege level; and then specify the command you want to assign. You can replace the **level** *privilege-level* portion with the **reset** keyword to reset the command to its default privilege level.

Once you have assigned the appropriate commands to the privilege level, you must now create an enable password for the new privilege level. This is configured by using the **enable password** or **enable secret** (recommended) command:

enable secret level *privilege-level secret*

To access the new privilege level, a user simply appends the desired privilege level when executing the **enable** command in user mode as shown in Example 8-5.

Example 8-5 *Accessing a Custom Privilege Level*

```
Switch> enable 10
Password: ********
Switch#
```

By adding the level to the **enable** command (e.g., **enable 10**), the desired level is accessed rather than the default enable mode (level 15).

TIP If you access a higher privilege level, you can use any commands specified in lower privilege levels. When creating privilege levels, it is a good idea to simulate all the commands a user would execute and then add them to the privilege level. Don't forget commands such as **configure terminal** and **exit**.

The **access-class** Command Syntax

The **access-class** line configuration command is used to apply access lists to management interfaces such as vty ports. To restrict Telnet and SSH access, you first create a simple access list that defines the source addresses of authorized hosts and then apply that access list to the management interface (e.g., line vty 0 4), as shown in Example 8-6. SSH connections are treated as coming in via the virtual terminal (vty) ports and, hence, are configured identically.

Example 8-6 *Restricting Telnet and SSH Access on Cisco IOS*

```
Switch(config)# access-list 1 permit 192.168.1.0 0.0.0.255
Switch(config)# line vty 0 4
Switch(config-line)# access-class 1 in
```

In Example 8-6, only hosts on the 192.168.1.0/24 subnet are able to access the switch via Telnet or SSH. You must bind the access list that defines the source hosts to the vty ports using the **access-class** command.

Configuration Tasks

In this scenario, you perform the following tasks:

- Step 1—Preparing the switches
- Step 2—Securing the Catalyst OS switch (XYZSW1)
- Step 3—Securing the Cisco IOS switch (XYZSW2)

Step 1—Preparing the Switches

In this step you:

- Configure the system name and management IP address
- Interconnect each switch and ensure ping connectivity
- Provide connectivity for Hosts A, B, and the AAA server

Configuring the System name and Management IP Address

On each switch, ensure you can access the switch via the console port.

Step 1 On XYZSW1, configure system name, Telnet/enable password of "cisco" and an IP address of 192.168.1.1/24, as shown in Example 8-7.

Example 8-7 *Configuring Basic Parameters on XYZSW1*

```
Console enable
Enter password:
Console (enable) set system name XYZSW1
System name set.
XYZSW1 (enable) set password
Enter old password: *****
Enter new password: *****
Retype new password: *****
Password changed.
XYZSW1 (enable) set enablepass
Enter old password: *****
Enter new password: *****
Retype new password: *****
Password changed.
XYZSW1 (enable) set interface sc0 192.168.1.1 255.255.255.0
```

Step 2 On XYZSW2 configure system name, Telnet/enable password of *cisco* and an IP address of 192.168.1.2/24, as shown in Example 8-8.

Example 8-8 *Configuring Basic Parameters on XYZSW2*

```
Switch> enable
Password:
Switch# configure terminal
Switch(config)# hostname XYZSW2
XYZSW2(config)# enable secret cisco
XYZSW2(config)# line vty 0 4
XYZSW2(config-line)# password cisco
XYZSW2(config-line)# login
XYZSW2(config-line)# interface VLAN1
XYZSW2(config-if)# ip address 192.168.1.2 255.255.255.0
XYZSW2(config-if)# end
XYZSW2# copy running-config startup-config
Building configuration...
[OK]
```

Interconnecting the switches

For this scenario you interconnect the switches using crossover unshielded twisted-pair (UTP) cables between Fast Ethernet 802.1Q trunk ports (you can use gigabit Ethernet trunks if you have these). Refer to Figure 8-1 for port assignments.

Step 1 On XYZSW1, configure 100 Mbps speed and full duplex on port 2/1 and enable trunking using 802.1Q, as shown in Example 8-9.

Example 8-9 *Configuring Trunks on XYZSW1*

```
XYZSW1> (enable) set port speed 2/1 100
Port 2/1 transmission speed set to 100Mbps.
XYZSW1> (enable) set port duplex 2/1 full
Port 2/1 to full-duplex.
XYZSW1> (enable) set trunk 2/1 on dot1q
Port(s)  2/1 trunk mode set to on.
Port(s)  2/1 trunk type set to dot1q.
```

Step 2 On XYZSW2, configure 100 Mbps speed and full duplex on port 0/1 and enable trunking using 802.1Q, as show in Example 8-10.

Example 8-10 *Configuring Trunks on XYZSW2*

```
XYZSW2# configure terminal
XYZSW2(config)# interface fastEthernet0/1
XYZSW2(config-if)# no shutdown
XYZSW2(config-if)# speed 100
XYZSW2(config-if)# duplex full
XYZSW2(config-if)# switchport mode trunk
XYZSW2(config-if)# switchport trunk encapsulation dot1q
XYZSW2(config-if)# end
XYZSW2# copy running-config startup-config
Building configuration...
[OK]
```

Once these configurations are complete, wait for at least 50 seconds to allow the spanning-tree state of the trunk ports to transition to forwarding.

Step 3 Verify you are able to ping XYZSW1 from XYZSW2 as shown in Example 8-11.

Example 8-11 *Verifying ping Connectivity Between XYZSW1 and XYZSW2*

```
XYZSW1> (enable) ping 192.168.1.2
!!!!!
```

Connecting the Hosts and AAA Server

Ensure all hosts are configured with an IP address, as shown in Figure 8-3. Then, connect each host as shown in Figure 8-3 to the appropriate switch and port.

Step 1 On XYZSW1, configure 100 Mbps speed, full duplex, and spanning-tree PortFast for ports 2/2 and 2/3 as shown in Example 8-12.

Example 8-12 *Configuring Access Ports on XYZSW1*

```
XYZSW1> (enable) set port speed 2/2-3 100
Ports 2/2-3 transmission speed set to 100Mbps.
XYZSW1> (enable) set port duplex 2/2-3 full
Port(s)  2/2-3 to full-duplex.
XYZSW1> (enable) set spantree portfast 2/2-3 enable
Warning: Spantree port fast start should only be enabled on ports connected
to a single host.  Connecting hubs, concentrators, switches, bridges, etc. to
a fast start port can cause temporary spanning tree loops.  Use with caution.
Spantree ports 2/2-3 fast start enabled.
```

Step 2 On XYZSW2, configure 100 Mbps speed and full duplex for port 0/2, as shown in Example 8-13.

Example 8-13 *Configuring Access Ports on XYZSW2*

```
XYZSW2# configure terminal
XYZSW2(config)# interface fastEthernet0/2
XYZSW2(config-if)# no shutdown
XYZSW2(config-if)# speed 100
XYZSW2(config-if)# duplex full
XYZSW2(config-if)# switchport mode access
XYZSW2# copy running-config startup-config
Building configuration...
[OK]
```

Step 3 Ensure that all devices in the network (switches, hosts, servers) can ping each other, as shown in Example 8-14.

Example 8-14 *Verifying ping Connectivity Between XYZSW1 and Hosts A, B, and the AAA Server*

```
XYZSW1> (enable) ping 192.168.1.10
!!!!!
XYZSW1> (enable) ping 192.168.1.100
!!!!!
XYZSW1> (enable) ping 192.168.1.101
!!!!!
```

Step 2—Securing the Catalyst OS Switch (XYZSW1)

On XYZSW1, you now perform minor security configurations to enhance the security of Telnet and console access to XYZSW1. In this step you:

- Set banner, lockout, and session Timeout Parameters
- Restrict Telnet and SNMP access
- Enable SSH support

Setting Banner, Lockout, and Session Timeout Parameters

Step 1 On XYZSW1, configure a banner that is displayed at each login prompt, as shown in Example 8-15.

Example 8-15 *Configuring a Banner on XYZSW1*

```
XYZSW1> (enable) set banner motd #
**********************************
*             WARNING            *
* Unauthorized access prohibited  *
**********************************
#
```

Step 2 On XYZSW1, configure a maximum of three unsuccessful login attempts for Telnet access and five for console access, and set a lockout of 180 seconds for Telnet access and 300 seconds for console access. Also configure a maximum of three unsuccessful enable mode login attempts for all modes of access, with a lockout of 300 seconds, as shown in Example 8-16. You also need to set the idle session timeout to be 5 minutes.

Example 8-16 *Configuring Lockout Policy on XYZSW1*

```
XYZSW1> (enable) set authentication login attempt 3 telnet
Login authentication attempts for telnet set to 3.
XYZSW1> (enable) set authentication login attempt 5 console
Login authentication attempts for console set to 5.
XYZSW1> (enable) set authentication login lockout 180 telnet
Login lockout time for telnet set to 180.
XYZSW1> (enable) set authentication login lockout 300 console
Login lockout time for console set to 300.
XYZSW1> (enable) set authentication enable attempt 3
Enable mode authentication attempts for console and telnet logins set to 3.
XYZSW1> (enable) set authentication enable lockout 300
Enable mode lockout time for console and telnet logins set to 300.
XYZSW1> (enable) set logout 5
Sessions will be automatically logged out after 5 minutes of idle time.
```

Restricting Telnet Access on XYZSW1

For this scenario, you permit Telnet access only from Host A (192.168.1.100), SNMP access from an SNMP management system at 192.168.1.20, and block Telnet access from all other hosts for both switches.

Step 1 On XYZSW1, add Host A (192.168.1.100) to the Telnet permit list and add 192.168.1.20 to the SNMP permit list, as shown in Example 8-17.

Example 8-17 *Creating Telnet and SNMP Permit Lists on XYZSW1*

```
XYZSW1> (enable) set ip permit 192.168.1.100 telnet
192.168.1.100 added to Telnet permit list.
XYZSW1> (enable) set ip permit 192.168.1.20 snmp
192.168.1.20 added to Snmp permit list.
```

Step 2 On XYZSW1, enable the Telnet and SNMP permit lists, as shown in Example 8-18.

Example 8-18 *Enabling Telnet and SNMP Permit Lists on XYZSW1*

```
XYZSW1 (enable) set ip permit enable telnet
Telnet permit list enabled.
XYZSW1 (enable) set ip permit enable snmp
SNMP permit list enabled.
```

Step 3 Verify that you now cannot Telnet to XYZSW1 from Host B. Next verify the permit lists on XYZSW1, as shown in Example 8-19.

Example 8-19 *Verifying Telnet and SNMP Permit Lists on XYZSW1*

```
XYZSW1> (enable) show ip permit
   Telnet permit list enabled.
   Ssh permit list disabled.
   Snmp permit list enabled.
Permit List        Mask               Access-Type
---------------    ---------------    ------------
192.168.1.20                           snmp
192.168.1.100                          telnet
Denied IP Address Last Accessed Time Type
192.168.1.101     01/30/02,03:13:44  Telnet
```

As you can see, the Telnet and SNMP permit lists are enabled, and the switch has logged the unauthorized Telnet connection attempt from Host B. Example 8-20 shows what a denied host receives when trying to Telnet to XYZSW1.

Example 8-20 *Denied Telnet Connection*

```
C:\>telnet 192.168.1.100
Connecting To 192.168.1.100...
Access not permitted.  Closing connection...
Connection to host lost.
C:\>
```

Enabling SSH Support

For this section, you enable SSH support and then disable Telnet access to XYZSW1.

Step 1 On XYZSW1, generate a 1024-bit RSA public/private key pair as shown in Example 8-21.

Example 8-21 *Generating an RSA Key Pair on XYZSW1*

```
XYZSW1> (enable) set crypto key rsa 1024
Generating RSA keys.... [OK]
```

Step 2 On XYZSW1, configure an IP permit list for SSH access, allowing only Host A (192.168.1.100) to connect via SSH. Then enable the IP permit list (for SSH) as shown in Example 8-22.

Example 8-22 *Configure an IP Permit List for SSH on XYZSW1*

```
XYZSW1> (enable) set ip permit 192.168.1.100 ssh
192.168.1.100 added to Ssh permit list.
XYZSW1> (enable) set ip permit enable ssh
SSH permit list enabled.
```

Step 3 On XYZSW1, verify the creation of the RSA keys, shown below as Example 8-23.

Example 8-23 *show crypto key on XYZSW1*

```
XYZSW1> (enable) show crypto key
```

Step 3—Securing the Cisco IOS Switch (XYZSW2)

On XYZSW2, you now perform minor security configurations to enhance the security of Telnet and console access to XYZSW2. In this step you:

- Set banner, lockout, and session timeout parameters
- Restrict Telnet and SNMP access
- Configure privilege levels to provide command authorization

Setting Banner, Lockout, and Session Timeout Parameters

Step 1 On XYZSW2, configure a banner that is displayed at each login prompt, as shown in Example 8-24.

Example 8-24 *Configuring a Banner on XYZSW2*

```
XYZSW2(config)# banner motd #
Enter TEXT message.  End with the character '#'.
**********************************
```

continues

Example 8-24 *Configuring a Banner on XYZSW2 (Continued)*

```
*             WARNING             *
* Unauthorized access prohibited  *
**********************************
#
```

Step 2 On XYZSW2, configure the idle session timeout to be 5 minutes for all management ports, as shown in Example 8-25.

Example 8-25 *Configuring Idle Session Timeouts on XYZSW2*

```
XYZSW2(config)# line vty 0 4
XYZSW2(config-line)# exec-timeout 5 0
XYZSW2(config-line)# line con 0
XYZSW2(config-line)# exec-timeout 5 0
XYZSW2(config-line)# end
```

Restricting Telnet Access on XYZSW2

In this section, you permit Telnet access only from Host A (192.168.1.100), SNMP access from an SNMP management system at 192.168.1.20, and block Telnet access from all other hosts.

Step 1 On XYZSW2, create two simple access lists to allow only the source IP address of Host A and the SNMP manager, as shown in Example 8-26.

Example 8-26 *Creating Access Lists on XYZSW2*

```
XYZSW2(config)# access-list 1 permit host 192.168.1.100
XYZSW2(config)# access-list 2 permit host 192.168.1.20
```

Step 2 On XYZSW2, configure the vty lines (Telnet management interfaces) to restrict management access based on the first access list you created in Step 1, as shown in Example 8-27.

Example 8-27 *Restricting Telnet Access on XYZSW2*

```
XYZSW2(config)# line vty 0 4
XYZSW2(config-line)# access-class 1 in
```

Step 3 On XYZSW2, configure the SNMP read-only and read-write community strings to be accepted only from the hosts specified in the second access list you created in Step 1, as shown in Example 8-28.

Example 8-28 *Restricting SNMP Access on XYZSW2*

```
XYZSW2(config)# snmp-server community cisco123 ro 2
XYZSW2(config)# snmp-server community cisco321 rw 2
```

Step 4 Verify that you can connect to XYZSW2 only via Telnet from Host A and cannot connect from Host B (see Example 8-29).

Example 8-29 *Failed Telnet to XYWSW2 from Host B*

```
C:\>telnet 192.168.1.2
Connecting To 192.168.1.2...Could not open a connection to host: Connect failed
C:\>
```

Configuring Privilege Levels to Provide Command Authorization

For this section, you configure privilege levels that allow network operators to view the system configuration and allow the operator to add a description to an interface.

The following commands are added to a custom privilege level; then a password is assigned to allow operators to gain access to the command set:

- **show running-config** (exec mode)
- **configure terminal** (exec mode)
- **interface** (global configuration mode)
- **description** (interface configuration mode)

Step 1 On XYZSW2, add the appropriate commands to a custom privilege level of 5, as shown in Example 8-30.

Example 8-30 *Assigning Commands to a Custom Privilege Leve*

```
XYZSW2(config)# privilege exec level 5 configure terminal
XYZSW2(config)# privilege exec level 5 show running-config
XYZSW2(config)# privilege configure level 5 interface
XYZSW2(config)# privilege interface level 5 description
```

Be sure that you understand the configuration mode (e.g., global configuration or exec) of the commands that you want to add.

Step 2 On XYZSW2, configure an enable secret password for the custom privilege level, as shown in Example 8-31.

Example 8-31 *Configuring an Enable Secret for a Custom Privilege Level*

```
XYZSW2(config)# enable secret level 5 cisco123
```

In Example 8-31, a password of "cisco123" is assigned to privilege level 5.

Step 3 Test your new privilege level by connecting to the switch via Telnet and logging into the new privilege level. Try all the permitted commands, as well as non-permitted commands (e.g., **erase**), as shown in Example 8-32.

Example 8-32 *Testing Custom Privilege Levels*

```
User Access Verification

Password: ******
XYZSW2> enable 5
Password:
XYZSW2# show running-config
Building configuration...

Current configuration:
interface FastEthernet0/1
…
…
XYZSW2# configure terminal
XYZSW2(config)# interface fastEthernet0/1
XYZSW2(config-if)# description TESTING
XYZSW2(config-if)# end
XYZSW2# erase flash
        ^
% Invalid input detected at '^' marker.
```

In Example 8-32, you access the custom privilege level by using the **enable 5** command. Notice that you can execute all the required commands, but when you try to execute an unauthorized command (e.g., **erase**), IOS notifies you that the command is invalid.

Scenario 8-2: Enhancing Security by Using AAA

In this scenario, you secure both a Catalyst CatOS-based switch and an IOS-based switch. In doing so, you increase the security of each device and consequently the network. This exercise builds on the previous scenarios by including the configuration of AAA.

Scenario Exercise

Figure 8-3 illustrates the topology used for this scenario. Corporation XYZ requires their existing switches to be secured using best practices. Corporation XYZ is about to acquire a larger corporation and needs to add new switches to the network. A new CiscoSecure ACS 3.2 server has been installed to allow Corporation XYZ to evaluate the use of both TACACS+ and RADIUS to provide a suitable access control model. It is through the configuration of AAA that the ACS server can be used.

Scenario Objectives

The scenario objectives are as follows:

- Configure security server support
- Configure AAA authentication
- Configure AAA authorization
- Configure AAA accounting

Additional Equipment Needed

Building on the previous scenario, only the following additional equipment is required to complete the following steps:

- A server with CiscoSecure ACS v3.2 software installed. CiscoSecure ACS software requires Windows 2000 Server with Service Pack 3 installed and Internet Explorer 6.0 SP1.

Scenario Planning

Planning AAA can seem reasonably complex at first, with many options available for authentication, authorization, and accounting. Many AAA configuration options are designed towards Point-to-Point Protocol (PPP) access (such as dial-up access to an Internet service provider, ISP), so the number of options that you actually need to secure network devices is reduced.

The first step in configuring any AAA services is to establish a relationship with a security server on the network. The TACACS+ and RADIUS security server protocols require the IP address of the security server and a shared secret key (password) be defined on the switch. The security server must also be configured with the IP address of the switch and must share the same shared secret key to successfully communicate. Once you have configured security server support, you are ready to complete the implementation of AAA.

Planning Authentication

The following services can be authenticated on both CatOS and IOS:

- **Login**—This refers to a user attempting to initially login to the switch via Telnet or the console.
- **Enable**—This refers to a user attempting to login to enable mode on the switch via Telnet or the console.

For example, login access is required to establish a user mode connection to a switch, while enable access is required to establish privileged access to the switch.

Planning Authorization

The following services can be authorized on both CatOS and Cisco IOS:

- **Exec**—This refers to a user attempting to start an *exec* process. An exec process is basically the shell (or command-line interface (CLI)) you use to input commands from a console or Telnet connection.
- **Enable (CatOS only)**—This refers to a user attempting to access privileged mode (enable mode). This occurs when a user executes the **enable** command in user mode.
- **Commands**—All commands that are executed can be authorized by the security server. This allows restricted use of dangerous commands such as **erase** or **copy**.

Authorization requires that a user be authenticated so that the switch can query the security server with a username and type of service required.

Planning Accounting

The following events can be accounted for on both CatOS and Cisco IOS:

- **Exec**—Records are generated for each exec process (e.g., console or telnet session) and include the user who invoked the process and the duration of the session.
- **System**—Records are generated every time a non-user system event occurs (such as a system reset).
- **Command**—Records are generated every time a command is issued. This enables you to see exactly what a user did during their exec session
- **Connect**—Records are generated every time a user running an exec session attempts a connection (such as Telnet or rlogin) to a remote device.

When you configure accounting, you also specify when to create accounting records. The following options are available:

- **Start-Stop**—Records are generated at both the start and completion of each event.
- **Stop-Only**—Records are generated at the completion of each event.
- **Wait-Start (Cisco IOS only)**—Records are generated at both the start and completion of each event. However, the switch does not allow the service to commence until an accounting acknowledgement is received from the security server.

NOTE Wait-Start should be used with great caution. For example, if Wait-Start is configured for EXEC sessions, you will not be able to log on to the network device if the AAA server is unavailable for any reason.

Command Syntax

The following sections describe the commands used for CatOS and Cisco IOS in this scenario.

CatOS Command Syntax

The following new CatOS commands are introduced in this scenario:

- The **set radius** command
- The **set tacacs** command
- The **set authentication** command
- The **set authorization** command
- The **set accounting** command

The set radius Command Syntax

The **set radius server** command is used to define the IP address of a RADIUS server:

```
set radius server ip-address [auth-port port] [acct-port port] [primary]
```

The **auth-port** and **acct-port** keywords specify the server User Datagram Protocol (UDP) ports that should be used for authentication and accounting communications (the various RADIUS products available differ; by default, ports 1812 and 1813 are used). If you specify the **primary** keyword and have multiple RADIUS servers defined, then this RADIUS server is contacted first. You can have up to three RADIUS servers defined. You must also specify a RADIUS secret key by using the **set radius key** command:

```
set radius key secret
```

This command sets the RADIUS key for *all* configured RADIUS servers.

The **set tacacs** Command Syntax

The **set tacacs server** command is used to define the IP address of a TACACS+ server:

```
set tacacs server ip-address [primary]
```

The **primary** keyword specifies that this TACACS+ server should always contacted first if multiple TACACS+ servers are defined. You can have up to three TACACS+ servers configured. You must also specify a TACACS+ secret key by using the **set tacacs key** command:

```
set tacacs key secret
```

This command sets the TACACS+ key for all configured TACACS+ servers.

The **set authentication** Command Syntax

To configure authentication, you use the **set authentication** command:

```
set authentication {login | enable} {radius | tacacs | kerberos}
  enable [all | console | telnet | http] [primary]
```

You can specify the authentication method for either login authentication (access to exec mode) or enable authentication (access to privileged configuration mode). You then specify the use of RADIUS or TACACS+ and can apply the configuration to whichever management interfaces you require. The **primary** keyword is used when you have multiple authentication methods (e.g., using TACACS+ and RADIUS simultaneously), and you want to specify which authentication method is attempted first. You can also disable local authentication by using the following command:

```
set authentication {login | enable} local disable
  [console | telnet | http | all]
```

WARNING Be careful when disabling local authentication. If you disable local authentication for every management interface, if your security server is down you will be unable to access the switch. A common practice is to disable local authentication for Telnet access, but leave it enabled for console access.

The set authorization Command Syntax

To configure authorization, you use the **set authorization** command. To configure authorization for exec mode and/or enable mode access use the following syntax:

```
set authorization {enable | exec} enable option fallback
  [both | console | telnet]
```

The *option* parameter specifies which security server protocol to use. Because RADIUS authorization is integrated with the authentication process, only tacacs is a valid option here. The fallback parameter specifies what action you should take if communication with the TACACS+ server fails (for example, you can specify **none**, meaning the service requested is granted if the TACACS+ server is down). Valid fallback options are **tacacs+**, **deny**, **if-authenticated**, and **none**.

To configure authorization for commands that can be executed use the following syntax:

```
set authorization commands enable {config | enable | all}
  option fallback [both | console | telnet]
```

Using the **config** parameter limits command authorization to configuration commands only (i.e., **show** commands do not need to be authorized).

The **set accounting** Command Syntax

To configure accounting, you use the **set accounting** command. To configure accounting for connect, exec, and system events, use the following syntax:

```
set accounting {connect | exec | system} enable
  {start-stop | stop-only} {tacacs+ | radius}
```

To configure accounting for *command* events, use the following syntax:

```
set accounting commands {config | enable | all} [stop-only] tacacs+
```

Notice that your only security server protocol option is **tacacs+**, because RADIUS does not support command authorization and accounting.

Cisco IOS Command Syntax

The following new Cisco commands are introduced in this scenario:

- The **radius-server** command
- The **tacacs-server** command
- The **aaa authentication** command
- The **aaa authorization** command
- The **aaa accounting** command

The **radius-server** Command Syntax

Before configuring RADIUS support, you must enable AAA by using the **aaa new-model** global configuration mode command:

```
aaa new-model
```

The **radius-server** global configuration command can then be used to configure the IP address of the RADIUS server:

```
radius-server host ip-address [auth-port port] [acct-port port] [key secret]
```

If you do not specify a key using the optional **key** keyword, you must specify a RADIUS secret key by using the **radius-server key** global configuration command, as shown here:

```
radius-server key secret
```

This command sets the RADIUS key for all RADIUS servers defined (unless a host has a specific key configured via the **radius-server host** command).

The **tacacs-server** Command Syntax

The **tacacs-server** global configuration command is used to define the IP address of a TACACS+ server:

```
tacacs-server host ip-address
```

You must also specify a TACACS+ secret key by using the **tacacs-server key** global configuration command:

```
tacacs-server key secret
```

This command sets the TACACS+ key for all configured TACACS+ servers. The key can optionally be configured on a per-server basis using the **key** parameter to the **tacacs-server host** global configuration command.

The **aaa authentication** Command Syntax

Before configuring AAA on Cisco IOS, you must enable AAA support explicitly using the **aaa new-model** command:

```
aaa new-model
```

To configure authentication for login (exec) access, you use the **aaa authentication login** global configuration command:

```
aaa authentication login {default | list} method1 [method2..]
```

The preceding command creates a profile that can be applied to different interfaces (e.g., a console port), allowing you to create different policies for different access methods. The **default** keyword specifies the default login authentication profile that is used for all management interfaces. You can specify multiple methods of authentication (e.g., RADIUS, TACACS+, line, none). To use a custom profile that you have created, you must bind the profile to the management interface that you want to control. The **login authentication** line configuration mode command is used to bind a profile to a management interface, as shown in Example 8-33.

Example 8-33 *Creating and Applying an AAA Authentication Profile*

```
Switch(config)# aaa authentication login PROFILE-A radius line
Switch(config)# line con 0
Switch(config-line)# login authentication PROFILE-A
```

In Example 8-33, an AAA authentication profile is created called PROFILE-A that uses RADIUS authentication as its primary method, and line authentication (i.e., the password assigned to the line to which access is being attempted) is used if the configured RADIUS server is down. The profile is then bound to the console port, meaning this profile is applied when access is attempted from the console port.

WARNING When you enable AAA by using the **aaa new-model** command, the default method of login authentication for Telnet access is to use the local method. The local method requires users to be defined locally using the **username** command. If no users are defined when you turn on AAA, you will be unable to gain Telnet access to the switch. A good rule of thumb is to set the default authentication method as the line method, which uses the line password (e.g., the vty password) as the default mechanism.

To configure authentication for enable mode access, you use the **aaa authentication enable** global configuration command:

```
aaa authentication enable default method1 [method2..]
```

You can create only a single (the default) enable authentication profile, and you do not need to bind this to any management interface because enable mode access is independent from a management interface.

The **aaa authorization** Command Syntax

To configure authorization, you use the **aaa authorization** global configuration command:

```
aaa authorization {network | exec | commands level} {default | list}
  method1 [method2..]
```

Similar to authentication, the preceding command creates a profile that can be applied to different interfaces. You can control authorization for either exec access (i.e., starting a command session on the switch), or you can control authorization for commands entered at a specific privilege level. Example 8-34 shows a sample AAA authorization configuration:

Example 8-34 *Creating and Applying an AAA Authorization Profile*

```
Switch(config)# aaa authorization exec PROFILE-EXEC radius none
Switch(config)# line vty 0 4
Switch(config-line)# authorization exec PROFILE-EXEC
```

In Example 8-34, an AAA authorization profile called PROFILE-EXEC is created that authorizes exec access using RADIUS. If the RADIUS server is down, the switch permits the access as indicated by the use of the **none** keyword.

TIP When configuring both AAA authentication and authorization, it is good practice to configure backup methods, as shown in Example 8-34. It is important to understand that these methods are invoked only if the primary security server is down. If the primary security server rejects a request, the switch rejects the requested access (and does not try the second method).

The **aaa accounting** Command Syntax

To configure *accounting*, you use the **aaa accounting** global configuration command:

```
aaa accounting {network | exec | connection | system | commands level}
  {default | list} {start-stop | stop-only | wait-start}
  method1 [method2..]
```

Similar to both authentication and authorization, the preceding command creates a profile that can be applied to different interfaces. You can specify accounting for exec, connection, system, or command events. You can also control when the accounting events are created.

Configuration Tasks

In this scenario, you perform the following tasks:

- Step 1—Configuring the CiscoSecure server for AAA support
- Step 2—Configuring each switch for AAA support
- Step 3—Confirming your AAA configuration

Step 1—Configuring the CiscoSecure Server for AAA Support

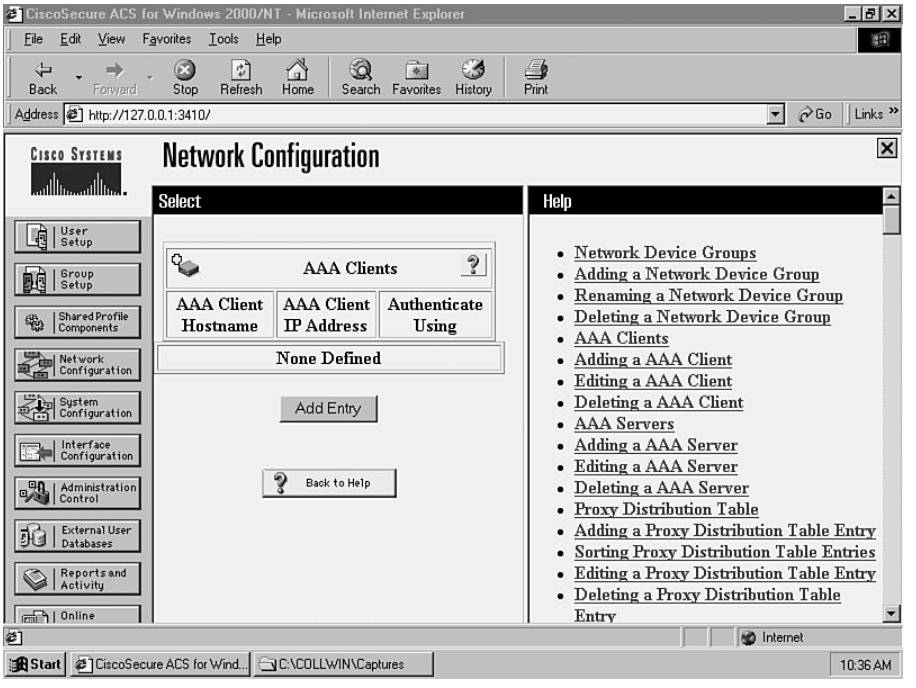
This scenario assumes that you have installed CiscoSecure ACS and performed preliminary switch configurations from Scenario 8-1. CiscoSecure ACS 3.2 must be installed on a Windows 2000 Server.

Once you have installed CiscoSecure ACS, you need to create an AAA client definition for each switch and then create user accounts for network administrators.

Step 1 Start the web-based CiscoSecure administration application by opening the URL *http://127.0.0.1:2002* from the ACS server.

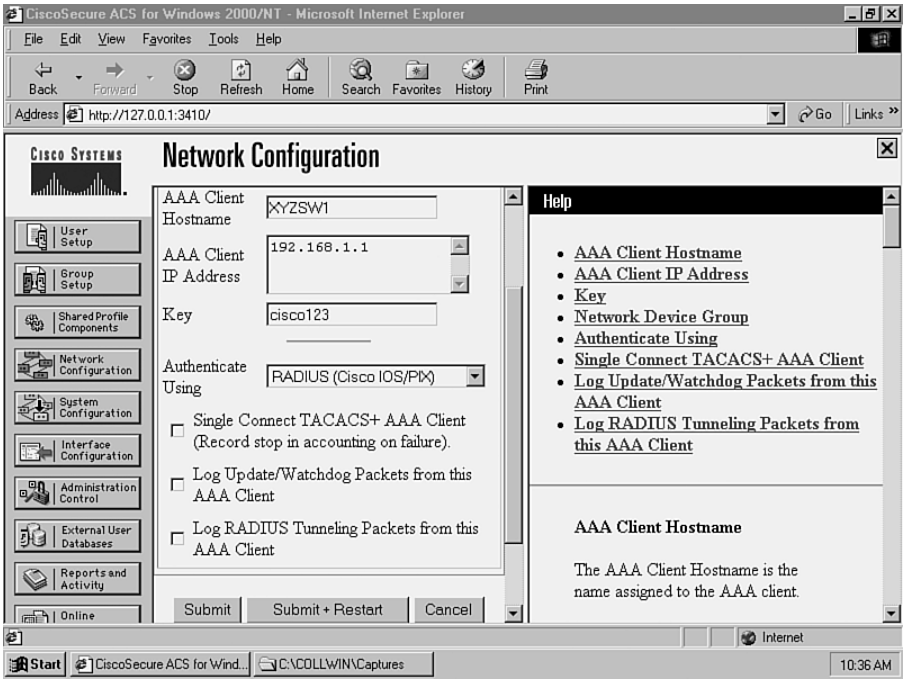
Step 2 Click on the *Network Configuration* button and click on the *Add Entry* button, as shown in Figure 8-4.

Figure 8-4 *Creating an AAA Client*



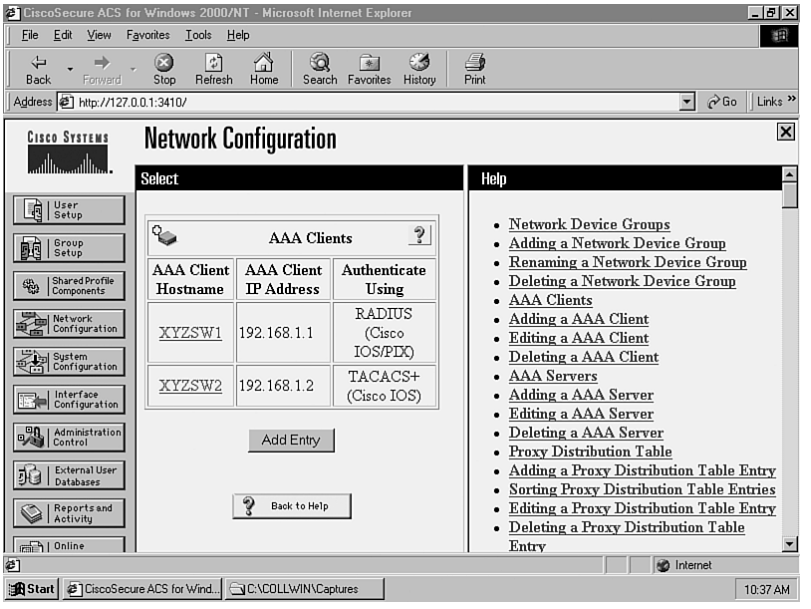
Step 3 Enter the appropriate parameters for XYZSW1, using a secret key of *cisco123* and ensuring *RADIUS (Cisco IOS/PIX)* is selected as the authentication protocol. Once complete, click the Submit + Restart button, as shown in Figure 8-5.

Figure 8-5 *Creating an AAA Client*



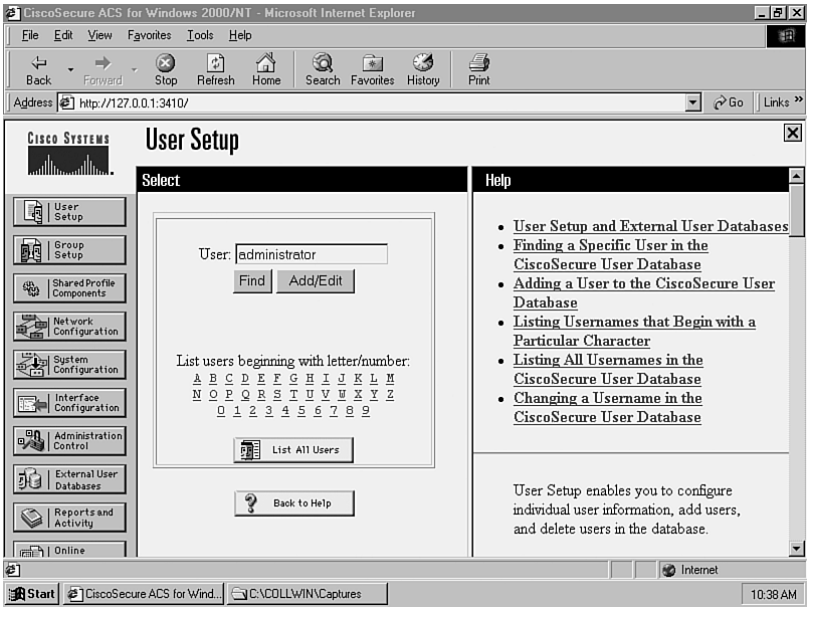
Step 4 Repeat Steps 2 and 3 for XYZSW2, except ensure that TACACS+ is selected as the authentication protocol. The AAA Client list should now contain entries for both XYZSW1 and XYZSW2, as shown in Figure 8-6.

Figure 8-6 *Verifying AAA Clients*



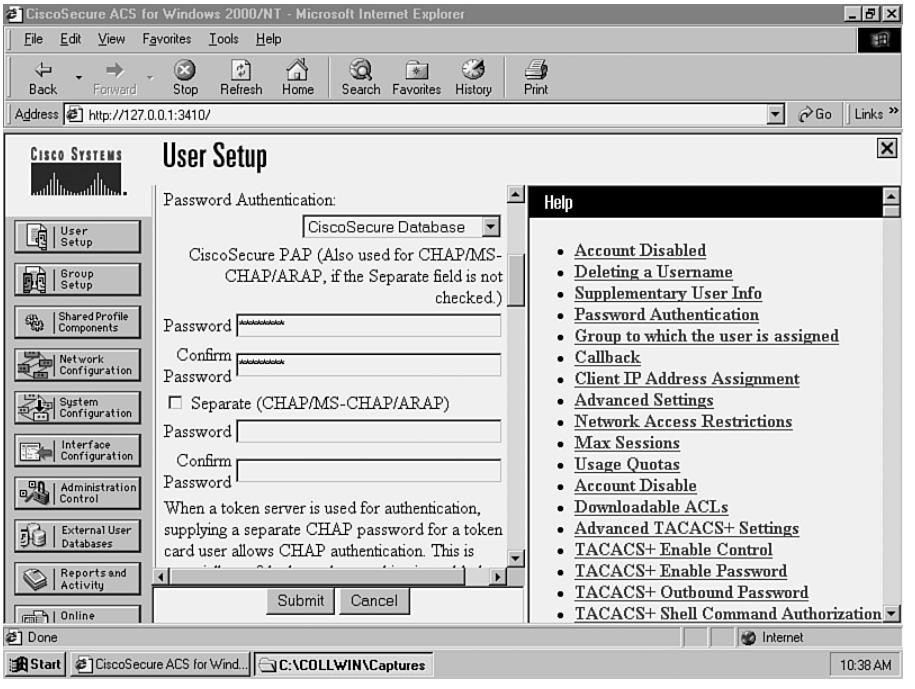
Step 5 Click on the User Setup button, and enter the username "administrator" in the User field. Click on the Add/Edit once complete as shown in Figure 8-7.

Figure 8-7 *Creating a User*



Step 6 In the User Setup page, scroll down to the Password Authentication section, configure a password of "password," and then click the *Submit* button as shown in Figure 8-8.

Figure 8-8 *Configuring a User Password*



Step 2—Configuring Each Switch for AAA Support

In this section, you configure each switch to use AAA. The following actions are required:

- Configuring the appropriate security server protocol support
- Configuring each switch to use AAA for authentication
- Configuring each switch to use AAA for management access authorization
- Configuring each switch to use AAA for management access accounting

Configuring the Appropriate Security Server Protocol Support

In this section, you configure both RADIUS (on XYZSW1) and TACACS+ (on XYZSW2) support.

Step 1 Configure RADIUS support on XYZSW1 by specifying the RADIUS server IP address (192.168.1.10) and RADIUS server secret key (cisco123), as shown in Example 8-35.

Example 8-35 *Configuring RADIUS Support on XYZSW1*

```
XYZSW1> (enable) set radius server 192.168.1.10
192.168.1.10 with auth-port 1812 acct-port 1813 added to radius server table
    as primary server.
XYZSW1> (enable) set radius key cisco123
Radius key set to cisco123
```

Step 2 Configure TACACS+ support on XYZSW2 by specifying the TACACS+ server IP address (192.168.1.10) and TACACS+ server secret key (cisco123), as shown in Example 8-36.

Example 8-36 *Configuring TACACS+ Support on XYZSW2*

```
XYZSW2(config)# tacacs-server host 192.168.1.10
XYZSW2(config)# tacacs-server key cisco123
```

Configuring Each Switch to Use AAA for Authentication

In this section, you learn how to use AAA authentication for management access on each switch.

Step 1 Configure XYZSW1 to use RADIUS authentication and authorization for Telnet logins only, as shown in Example 8-37.

Example 8-37 *Configuring RADIUS Authentication for Telnet Login on XYZSW1*

```
XYZSW1> (enable) set authentication login radius enable telnet
radius login authentication set to enable for telnet session.
```

In Example 8-37, the use of the **telnet** keyword enables RADIUS authentication for Telnet access only.

Step 2 Configure XYZSW2 to use TACACS+ authentication for Telnet logins only, as shown in Example 8-38.

Example 8-38 *Configuring TACACS+ Authentication for Telnet Login on XYZSW2*

```
XYZSW2(config)# aaa new-model
XYZSW2(config)# aaa authentication login default line
XYZSW2(config)# aaa authentication login TELNET group tacacs+ line
XYZSW2(config)# line vty 0 4
XYZSW2(config-line)# login authentication TELNET
XYZSW2(config)# line con 0
XYZSW2(config-line)# password cisco
XYZSW2(config-line)# end
```

In Example 8-38, notice that you have to globally enable AAA by using the **aaa new-model** command. The next command tells the switch to use the locally configured **line** password as the default login authentication mechanism. Next, you create a login authentication profile called **TELNET**, which uses TACACS+ as the authentication method, and uses the **line** authentication method as a backup in case the TACACS+ server is down. The last step is to apply the profile (TELNET) to the vty lines, which then enables TACACS+ authentication for Telnet access.

WARNING Be careful when setting the default authentication mechanism as the line password. If your console port does not have a line password configured, access is denied to the console port. Ensure that you set a line password for the console port if you are using line password as the default authentication mechanism.

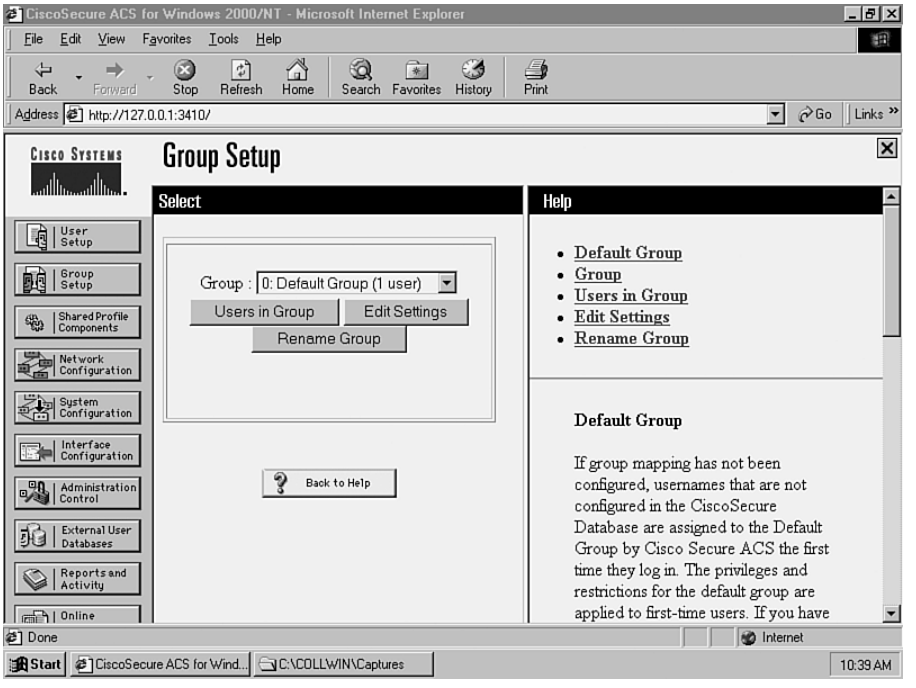
Configuring Each Switch to Use AAA for Management Access Authorization

In this section, you learn how to use AAA authorization to allow enable mode management access on each switch and to deny the use of the **erase** command on XYZSW2 using TACACS+.

NOTE You can enable mode authorization with two approaches. The first is to apply enable mode authentication when an authenticated user types in the **enable** command. The second is to authorize enable mode access when a user first attempts management access (e.g., authenticates at a Telnet prompt), which takes the user straight to enable mode if authenticated and authorized. TACACS+ supports both of these methods, while RADIUS supports only the second method (RADIUS does support the first, but not very well). For this reason, we look exclusively at the second method of enable mode authorization.

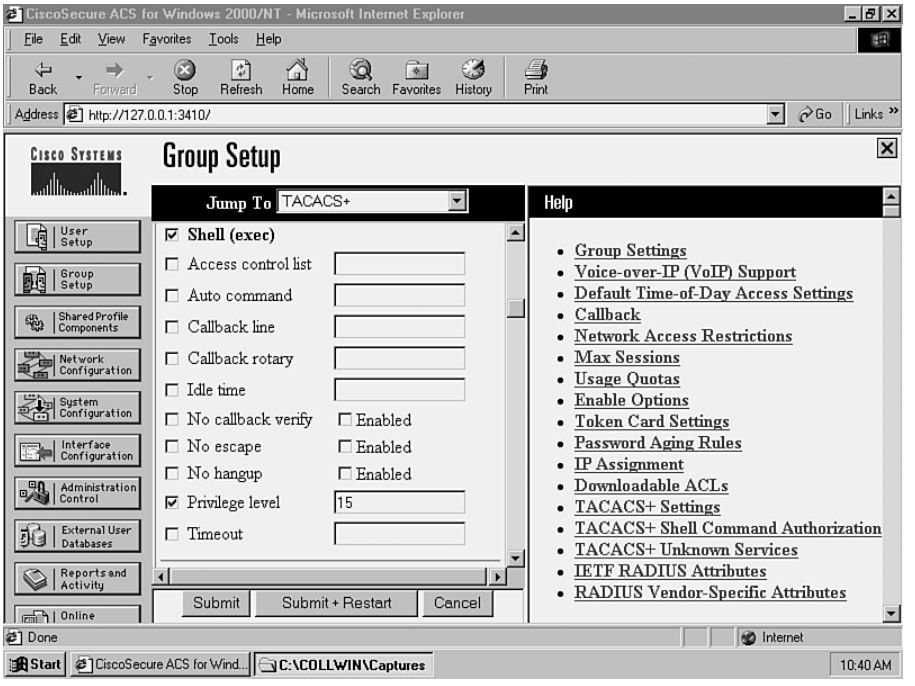
Step 1 On the CiscoSecure ACS server, click on the *Group Setup* button, ensure the *Default Group* is selected (which contains the user *administrator*), and click on the *Edit Settings* button, as shown in Figure 8-9.

Figure 8-9 *Configuring Group Settings*



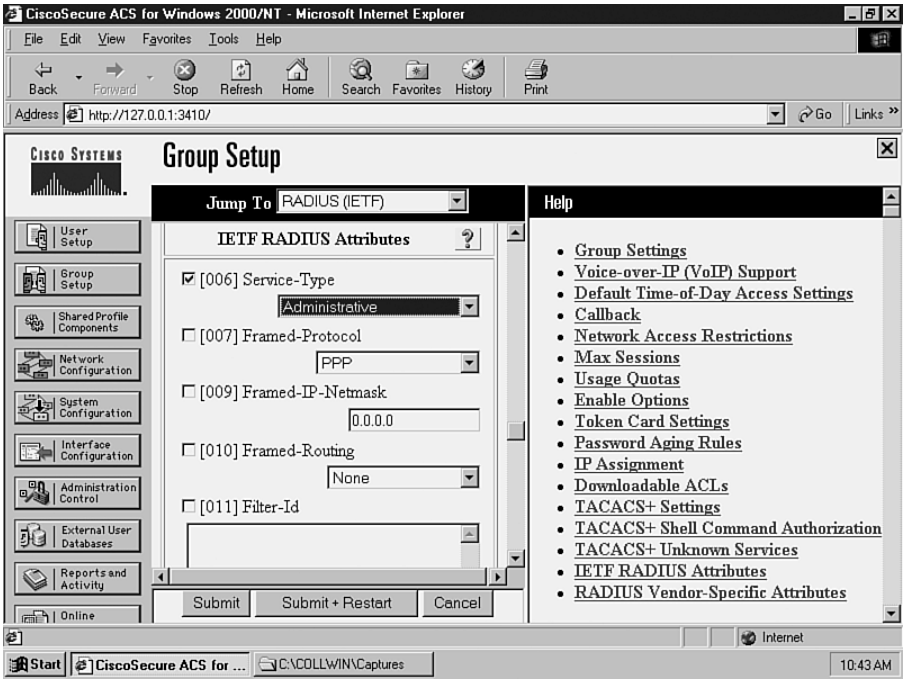
Step 2 On the Group Setup page, scroll down to the *TACACS+ Settings* section and enable the Shell (exec) setting and configure Privilege level to be 15 (enable mode), as shown in Figure 8-10. This authorizes the user for enable mode (privilege level 15) access via TACACS+.

Figure 8-10 *Enabling TACACS+ Enable Mode Authorization*



Step 3 On the Group Setup page, scroll down to the *IETF RADIUS Attributes* section and enable the *[006] Service-Type* attribute, changing the attribute value to *Administrative*. This *authorizes* the user for enable mode access via RADIUS. Once complete, click on the *Submit + Restart* button to apply the configuration, as shown in Figure 8-11.

Figure 8-11 *Enabling RADIUS Enable Mode Authorization*



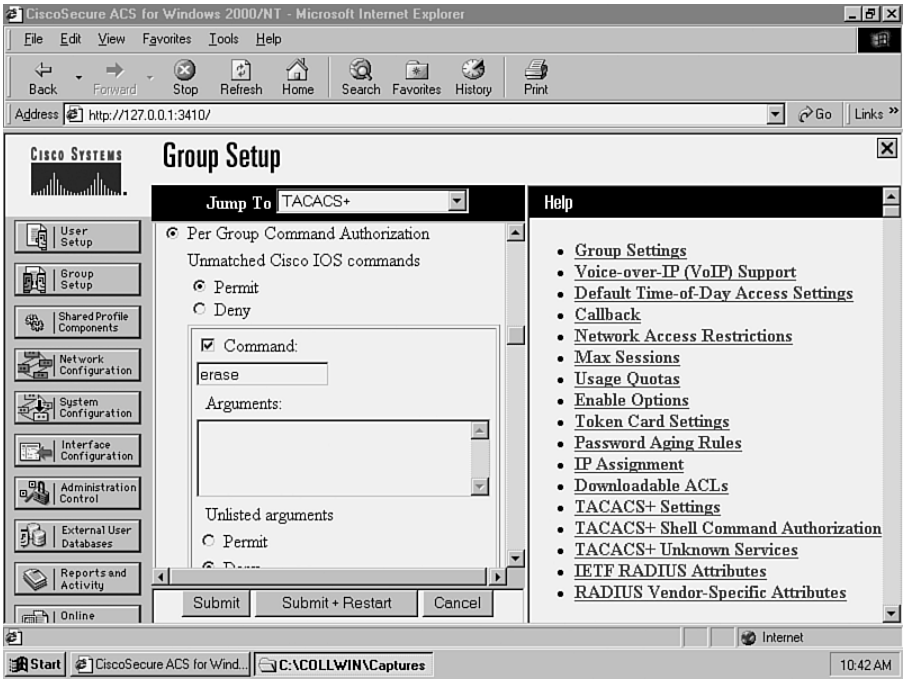
Step 4 Remaining on the Group Setup page, scroll down to the *Shell Command Authorization Set* section and enable the *Per Group Command Authorization* setting. Configure the *Unmatched Cisco IOS commands* setting to *Permit*, and then disable the **erase** command, as shown in Figure 8-12. This prevents the user from using the **erase** command.

Step 5 Configure XYZSW2 to use TACACS+ authorization for enable mode access, as shown in Example 8-39.

Example 8-39 *Configuring TACACS+ Authorization for Enable Mode Access on XYZSW2*

```
XYZSW2(config)# aaa authorization exec TELNET group tacacs+ none
XYZSW2(config)# line vty 0 4
XYZSW2(config-line)# authorization exec TELNET
XYZSW2(config-line)# end
```

Figure 8-12 *Enabling TACACS+ Command Authorization*



In Example 8-39, you use the **exec** keyword to tell the switch to contact the TACACS+ server for authorization when a user starts an exec process. Notice the use of the **none** keyword to ensure the user can at least get user mode access if the TACACS+ server is down (if you omitted this and the TACACS+ server is down, you would not be able to access the switch via Telnet).

Step 6 Configure XYZSW2 to use TACACS+ authorization for commands, as shown in Example 8-40.

Example 8-40 *Configuring TACACS+ Authorization for Commands on XYZSW2*

```
XYZSW2(config)# aaa authorization commands 15 TELNET group tacacs+ none
XYZSW2(config)# line vty 0 4
XYZSW2(config-line)# authorization commands TELNET
XYZSW2(config-line)# end
```

In Example 8-40, you use the **commands** keyword to tell the switch to contact the TACACS+ server for authorization of commands when in enable mode (privilege level 15). Notice that this authorization is used only for Telnet access because we have created a specific AAA profile (TELNET) and applied it only to the vty lines.

Configuring Each Switch to Use AAA for Management Access Accounting

In this section, you learn how to use AAA accounting to audit exec events (e.g., starting a Telnet session) and command events (invoked each time a command is issued).

Step 1 On XYZSW1, configure AAA accounting for exec events (audit both the start and stop of each event), as shown in Example 8-41.

Example 8-41 *Configuring AAA Accounting on XYZSW1*

```
XYZSW1> (enable) set accounting exec enable start-stop radius
Accounting set to enable for exec events in start-stop mode.
```

Step 2 On XYZSW2, configure AAA accounting for exec events (audit both the start and stop of each event) and enable mode command events (ensure you can contact the TACACS+ server before allowing commands to proceed) for Telnet sessions, as shown in Example 8-42.

Example 8-42 *Configuring AAA Accounting on XYZSW2*

```
XYZSW2(config)# aaa accounting exec TELNET start-stop group tacacs+
XYZSW2(config)# aaa accounting commands 15 TELNET-CMD wait-start group tacacs+
XYZSW2(config)# line vty 0 4
XYZSW2(config-line)# accounting exec TELNET
XYZSW2(config-line)# accounting commands 15 TELNET-CMD
XYZSW2(config)# end
```

You create two separate profiles for each event category you are accounting and then apply the profiles to the vty lines. Notice the use of the **wait-start** keyword to ensure enable mode command events are always audited.

Step 3 Now you are ready to test your accounting configuration. From Host A, make a Telnet connection to XYZSW1 logging in as the administrator user you created earlier, leaving the session open for 30 seconds or so, and then disconnect. On the CiscoSecure ACS server, click on the *Reports and Activity* button, then click the *RADIUS Accounting* hyperlink, and then click the *RADIUS Accounting active.csv* hyperlink. Figure 8-13 shows the accounting information that you should see.

Figure 8-13 shows a start and stop record around 09:09 a.m. The *Service Type* column indicates this is a *NAS Prompt* (exec) event, and the *Acct-Session-Time* column for the *stop* record indicates the session lasted for 35 seconds. The *User-Name* column indicates that the user administrator established the exec connection.

Step 4 From Host A, make a Telnet connection to XYZSW2, perform some minor configuration changes (e.g., change an interface description), and then disconnect. On the CiscoSecure ACS server, click on the *Reports and Activity* button, then click the *TACACS+ Accounting* hyperlink, and then click the *TACACS+ Accounting active.csv* hyperlink. Figure 8-14 shows the accounting information that you should see.

Figure 8-13 *Viewing RADIUS Accounting Information*

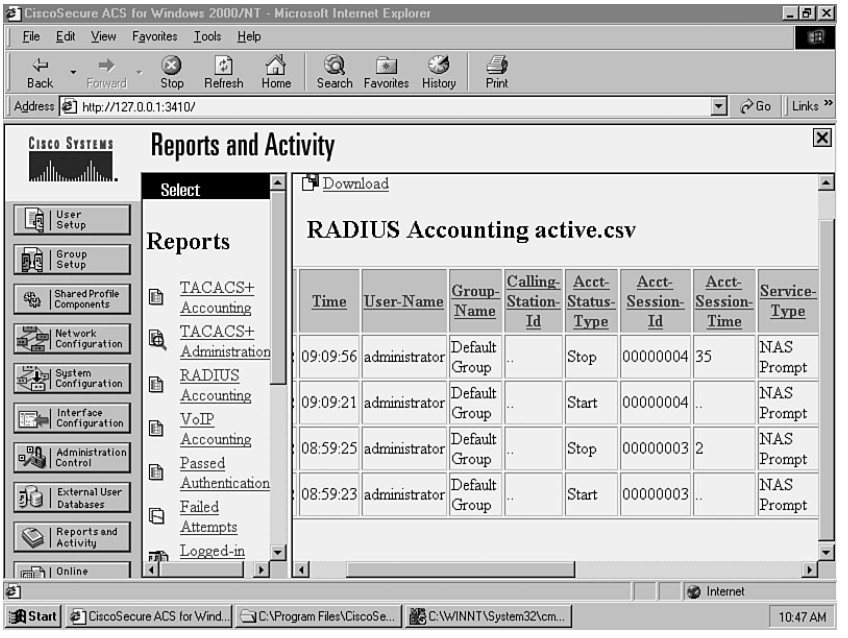


Figure 8-14 *Viewing TACACS+ Accounting Information*

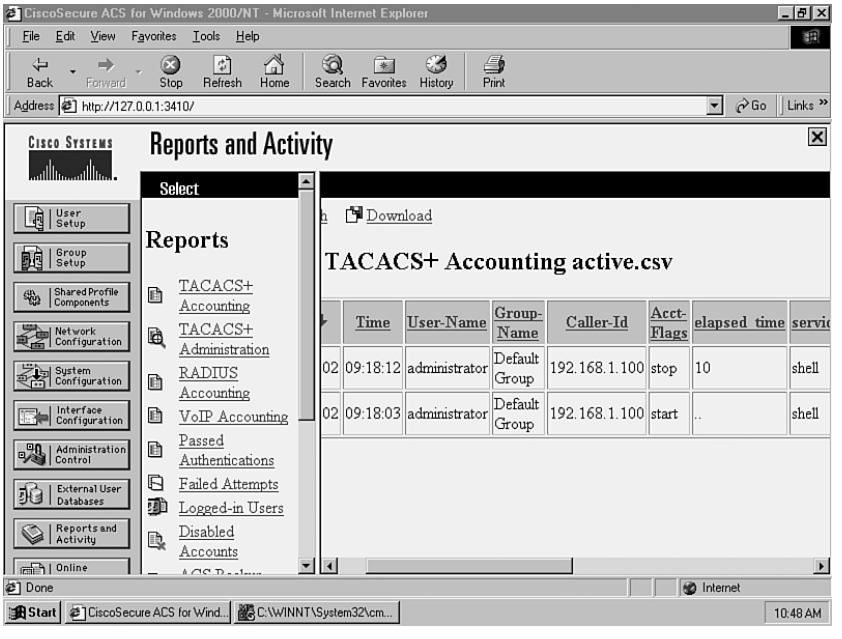


Figure 8-14 shows a start and stop record around 09:18 a.m. The *Service* column indicates this is a *shell* (exec) event, and the *elapsed time* column for the *stop* record indicates the session lasted for 10 seconds. The *User-Name* column indicates that the user *administrator* established the exec connection, and the *Caller-Id* column indicates that the exec session was initiated from 192.168.1.100 (Host A).

Notice that no events are related to commands in Figure 8-14. This is because command accounting records are stored in the TACACS+ Administration database on CiscoSecure ACS. On the CiscoSecure ACS server, click on the *Reports and Activity* button, then click the TACACS+ Administration hyperlink, and then click the TACACS+ Administration active.csv hyperlink. Figure 8-15 shows the accounting information that you should see.

Figure 8-15 *Viewing TACACS+ Administration Information*

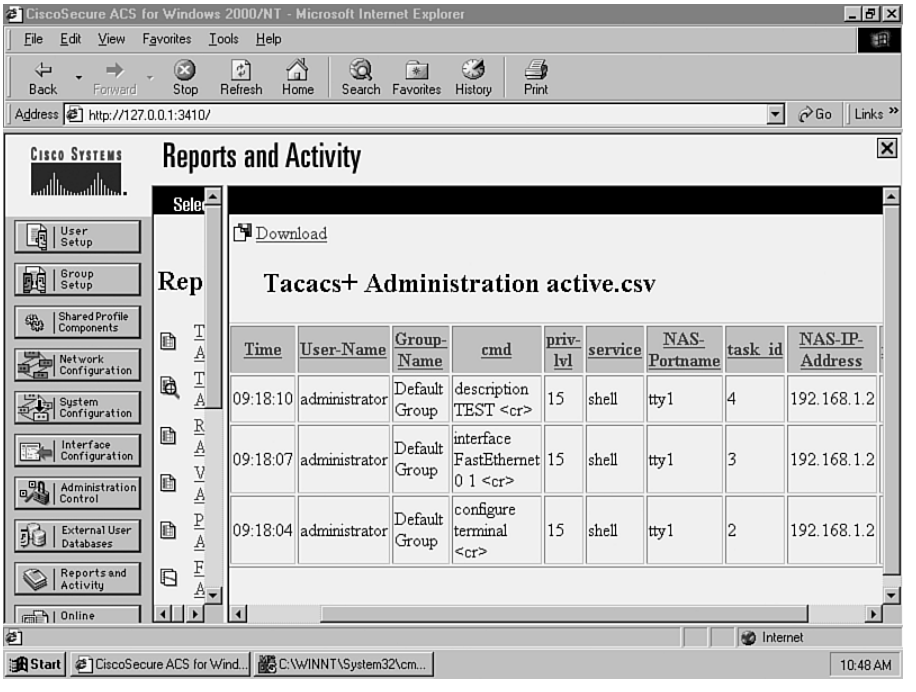


Figure 8-15 shows three command events that occurred around 09:18 a.m. The events indicate that the user administrator configured a description of TEST for interface fastEthernet0/1. The NAS-IP-Address column indicates the configuration was performed on 192.168.1.2 (XYZSW2).

Step 3—Confirming Your AAA Configuration

Now it is time to test your configuration by attempting Telnet access to both switches and requesting various services.

Step 1 From Host A, attempt a Telnet connection to XYZSW1. Enter in the administrator credentials (password = *password*), as shown in Example 8-43.

Example 8-43 *Testing Telnet Access to XYZSW1*

```
C:\> telnet 192.168.1.1
Cisco Systems, Inc. Console
Username: administrator
Password: ********
XYZSW1> (enable)
```

Notice that once you have successfully authenticated, you are taken directly to enable mode on the switch. This is because the RADIUS server returned the *006 Service-Type* attribute with a value of *Administrative*, which tells the switch to grant the user enable mode access.

Step 2 From Host A, attempt a Telnet connection to XYZSW2. Enter in the administrator credentials (password = *password*), as shown in Example 8-44.

Example 8-44 *Testing Telnet Access to XYZSW2*

```
C:\> telnet 192.168.1.2
Username: administrator
Password: ********
XYZSW2#
```

Again, you have been granted enable mode access directly. This is because you configure the switch to authorize exec access, and the TACACS+ server authorized the user with a privilege level of 15 (enable mode).

Step 3 In the Telnet session established in Step 2, try entering configuration mode, execute a few commands, and then exit back to enable mode. Then attempt to erase the startup configuration of the switch, as shown in Example 8-45.

Example 8-45 *Testing Command Authorization on XYZSW2*

```
XYZSW2# configure terminal
XYZSW2(config)# interface fastEthernet0/10
XYZSW2(config-if)# description Just Mucking around here…
XYZSW2(config-if)# end
XYZSW2# erase startup-config
Command authorization failed.
```

In Example 8-45, notice how you can execute normal configuration commands, but once you attempt to execute the **erase** command, you cannot. This is because every time the user executes a command, the switch contacts the TACACS+ server to authorize the command and we had previously denied it.

Step 4 (Optional) Access each switch via the console port. Notice that you do not need to enter user credentials (only a line password or local username authentication), and you can perform all commands. This is because you have applied the AAA configurations only for Telnet access.

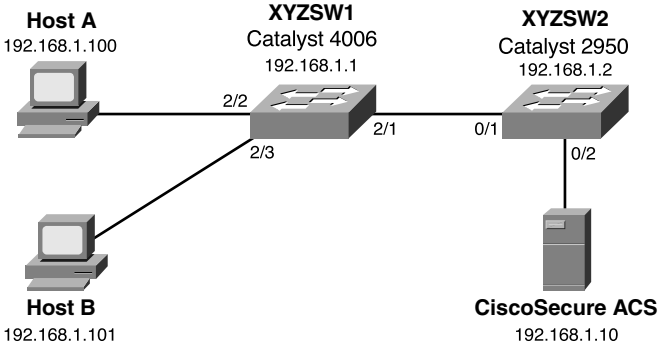
Scenario 8-3: Securing Device Access

In this scenario, you secure Catalyst switch ports using the various techniques available. Although in the real world you would generally not configure all of these techniques on a single network, the scenario shows you how to configure each technology.

Scenario Exercise

Figure 8-16 illustrates the scenario topology used for this scenario.

Figure 8-16 *Scenario 8-3 Topology*



Corporation XYZ wants to try the various port security techniques available to determine which technique works best for the network. You learn how to configure the following port security techniques:

- Simple port security
- 802.1x authentication

Scenario Objectives

The scenario objectives are as follows:

- Configure basic port security
- Configure 802.1x authentication

Additional Equipment Needed

The additional equipment Needed is as follows:

- Hosts B must be Windows XP or Windows 2000 SP4 hosts

Scenario Planning

Securing device access is a process that must be well-planned before implementation. Implementing port security techniques without careful planning can cause administrative headaches, with users being unable to connect to certain ports or users being granted access to the network when they shouldn't be. In this scenario, you configure the following methods of securing device access:

- Basic port security
- 802.1x authentication

Planning Basic Port Security

Basic port security is easily configured and allows you to secure access to a port based upon a MAC address basis. It does not allow you to dynamically determine the VLAN a port should be placed into, so make sure you understand this. Basic port security is also configured locally and has no mechanism for controlling port security in a centralized fashion for distributed switches. Basic port security is normally configured on ports that connect servers or fixed devices, because the likelihood of the MAC address changing on that port is low. A common example of using basic port security is applying it to a port that is in an area of the physical premises that is publicly accessible. This could include a meeting room or reception area that might have an IP telephone available. By restricting the port to accept only the MAC address of the IP telephone, you prevent unauthorized access if somebody plugged another device into the port.

When configuring port security, you should be aware of the default configurations:

- **The maximum number of secure MAC addresses permitted**—Depending on the platform, both CatOS and Cisco IOS switches can permit multiple hosts on a port when port security is configured. If you don't manually specify these addresses, they

are auto-learned in the order MAC addresses are heard on the port. If your goal is to allow only a single MAC address on the port, you might be required to manually configure the maximum MAC addresses on each port as one.

- **The security violation action**—The default security violation action on both CatOS and Cisco IOS switches is to shut down the port, requiring manual re-enabling of the port by an administrator. This action could be used as a denial of service attack, so consider this action very carefully.

Planning 802.1x Authentication

The IEEE 802.1x standard provides a framework that allows users (rather than MAC addresses) to be authenticated for switch access to a port. 802.1x can use a centralized security database to provide authentication information, which allows for scalability and ease of management. On Cisco platforms, 802.1x support requires a RADIUS server, so you must configure this server and enable RADIUS support on the switch before proceeding. In recent versions of Cisco IOS and CatOS, RADIUS authorization attributes can also be associated with a user, which define the VLAN a user belongs to as well a port-based access control list that should be applied to traffic received from the user.

A restriction of 802.1x is the requirement for the host connecting to the switch port to be 802.1x aware. This means the operating system must have 802.1x client support. Microsoft Windows XP and Windows 2000 are the only operating systems that currently support 802.1x natively, although third-party clients do exist for other operating systems.

NOTE 802.1x client support is available for Windows 2000 starting in Service Pack 4.

Command Syntax

The following sections describe the commands used for configuring standard port security and 802.1x security in this scenario.

Standard Port Security Command Syntax

The following commands, which are used to configure port security, are introduced in this scenario:

- The **set port security** command (CatOS)
- The **switchport security** command (Cisco IOS)

The set port security Command Syntax

To enable port security on CatOS, you use the **set port security** command. The first step you must take is to enable port security on a particular port. You then can allow one or more MAC addresses to use a secured port. You can manually specify these addresses, allow the switch to auto-learn the addresses, or use a mixture of both. Finally, you can specify a *violation* action (either shut down the entire port or block unauthorized traffic), which occurs when an unauthorized MAC address is detected on the port. The **set port security** command has the following syntax:

```
set port security mod/port [enable | disable] [mac_addr] [age age_time]
  [maximum limit] [shutdown shutdown-time] [violation {shutdown | restrict}]
```

Example 8-46 illustrates configuring port security.

Example 8-46 *Configuring Port Security*

```
Switch> (enable) set port security 2/1 enable
Port 2/1 port security enabled with the learned mac address.
Trunking disabled for Port 2/1 due to Security Mode
Switch> (enable) set port security 2/1 maximum 10
Maximum number of secure addresses set to 10 for port 2/1.
Switch> (enable) set port security 2/1 00-d0-b5-11-22-33
Mac address 00-d0-b5-11-22-33 set for port 2/1.
Switch> (enable) set port security 2/1 violation restrict
Port security violation on port 2/1 will cause insecure packets to be dropped.
```

NOTE When following this scenario, do not use the MAC addresses shown in the text; use the correct MAC address of your Host A instead.

Example 8-46 sets port 2/1 to allow a maximum of ten hosts. A single static host is permitted, with the remaining nine MAC addresses added dynamically as new hosts send traffic through the port. If an insecure packet is received, the port drops the packets (as opposed to the default configuration of shutting down the entire port).

The **switchport security** Command Syntax

To enable port security on Cisco IOS, you use the **switchport security** interface configuration command syntax:

```
switchport port-security [maximum number] [mac-address mac-address]
```

If you omit the optional parameters, port security is enabled and allows for up to 132 secure MAC addresses. The optional **maximum** keyword allows you to specify the maximum number of MAC addresses allowed on the interface. The optional **mac-address** keyword allows you to add specific MAC addresses to the secure MAC address list (if you do not do this, then the switch auto-learns the secure MAC addresses).

By default, if an unauthorized MAC address is detected on a secure port, the port is shut down and must be administrative enabled. To configure what happens when an unauthorized MAC address is detected on the interface, you use the **switchport security violation** command:

```
switchport port-security violation {protect | restrict | shutdown}
```

The **protect** keyword drops any frames from unauthorized hosts, but still forwards traffic for authorized hosts. The **restrict** keyword generates a trap violation (SNMP and SYSLOG), which is sent to the network management station.

Example 8-47 shows a sample configuration that allows only a single host (MAC address of 00-01-02-00-D8-1D) on the switch port. If another host connects to the port, the port is shut down and must be re-enabled by an administrator.

Example 8-47 *Configuring Port Security on Cisco IOS*

```
Switch(config)# interface fastEthernet0/1
Switch(config-if)# switchport port-security
Switch(config-if)# switchport port-security maximum 1
Switch(config-if)# switchport port-security mac-address 00-01-02-00-D8-1D
Switch(config-if)# switchport port-security violation shutdown
```

802.1x Security Command Syntax

The following commands, which are used to configure 802.1x security, are introduced in this scenario:

- The **set dot1x** and **set port dot1x** command (CatOS)
- The **aaa authentication dot1x** command (Cisco IOS)
- The **dot1x port-control** command (Cisco IOS)

The **set dot1x** and **set port dot1x** Command Syntax (CatOS)

When configuring 802.1x, you must have a RADIUS server configured before enabling 802.1x support. Once you have configured RADIUS support, you must globally enable 802.1x support using the **set dot1x system-auth-control** command:

```
set dot1x system-auth-control {enable | disable}
```

Once 802.1x is enabled globally for the switch, you can then configure individual ports to use 802.1x security by using the **set port dot1x** command. By default, each port has a specific 802.1x port state of *force-authorized*, which means that each port is automatically authorized to forward traffic (in effect, 802.1x is turned off). You must set the port state to *auto* to enable 802.1x on the port, using the **set port dot1x port-control** command:

```
set port dot1x mod/port port-control {auto | force-authorized | force-unauthorized}
```

Once you have set the 802.1x port state, you must initialize the port using the set **port dot1x initialize** command:

```
set port dot1x mod/port initialize
```

TIP 802.1x supports the use of multiple hosts attached to a single port. This can occur when a hub is connected to the switch. To enable support for multiple hosts, you must configure the **set port dot1x** *mod/port* **multiple-host enable** command on the appropriate port.

The **aaa authentication dot1x** Command Syntax (Cisco IOS)

To configure 802.1x support on Cisco IOS Catalyst switches, the following prerequisites must be configured:

- AAA enabled (using the **aaa new-model** command)
- RADIUS support configured (using the **radius-server** command)

Once these prerequisites have been configured, you must configure the 802.1x authentication profile to use RADIUS, using the **aaa authentication dot1x** global configuration command:

```
aaa authentication dot1x default method1 [method2…]
```

Example 8-48 shows how to configure 802.1x using RADIUS authentication.

Example 8-48 *Enabling 802.1x Authentication*

```
Switch(config)# aaa authentication dot1x default group radius
```

NOTE You can configure 802.1x authentication to use the local switch user database by specifying the **local** keyword. This method is recommended only for testing purposes.

The **dot1x port-control** Command Syntax (Cisco IOS)

Once 802.1x authentication has been enabled on a Cisco IOS switch, you must then configure 802.1x on each port that you want to use it. As for CatOS, all ports by default are in the force-authorized state and must be set to the auto state to enable 802.1x support. The **dot1x port-control** interface configuration command is used to enable 802.1x on a port:

```
dot1x port-control {auto | force-authorized | force-unauthorized}
```

TIP To enable support for multiple hosts on Cisco IOS, you must configure the **dot1x multiple-hosts** interface configuration command on the appropriate port.

Configuration Tasks

In this scenario, you perform the following tasks:

- Step 1—Configure basic port security
- Step 2—Configure 802.1x authentication

Step 1 Configuring Basic Port Security

Basic port security is supported on both Cisco IOS and CatOS switches. In this scenario, you configure basic port security for the ports connected to Host A and the CiscoSecure ACS server. In this step:

- Configure XYZSW1 for basic port security
- Configure XYZSW2 for basic port security
- Verify that port security is functioning

Configuring XYZSW1 for Basic Port Security

On XYZSW1 you permit only Host A's MAC address on port 2/2, and block access from any other source MAC addresses detected on the port.

On XYZSW1, configure port security for port 2/2, allowing only Host A (MAC address = 00-40-96-39-FA-0A) access to the switch port (see Example 8-49).

Example 8-49 *Configure Port Security on XYZSW1*

```
XYZSW1 (enable) set port security 2/2 enable 00-40-96-39-FA-0A violation restrict
Port 2/2 security enabled, violation mode restrict.
Mac address 00-40-96-39-fa-0a set for port 2/2.
```

The **restrict** keyword configures the port to reject frames from unauthorized MAC addresses (also known as the *violation action*). The default violation action is to shut down the port if an unauthorized MAC address is detected.

Configuring XYZSW2 for Basic Port Security

On XYZSW2 you permit only the CiscoSecure ACS server MAC address on port 0/2 and shut down the port if any other source MAC addresses are detected on the port.

On XYZSW2, configure port security for port 0/2 allowing only one MAC address on the port, as shown in Example 8-50.

Example 8-50 *Configure Port Security on XYZSW2*

```
XYZSW2(config)# interface fastEthernet0/2
XYZSW2(config-if)# switchport port-security
XYZSW2(config-if)# switchport port-security maximum 1
XYZSW2(config-if)# switchport port-security mac-address 00-01-02-00-D8-1D
XYZSW2(config-if)# switchport port-security violation shutdown
```

The maximum 1 configuration means that only a single MAC address is allowed on the switch port (the default is 132). The **violation shutdown** configuration means that any frames received from unauthorized MAC addresses causes a shut down of the port.

NOTE The **switchport port-security** command replaces the **port security** interface configuration command used in older IOS versions on the Catalyst 2900XL/3500XL switches.

Verifying That Port Security Is Functioning

The following outlines how to verify that port security is functioning:

Step 1 On XYZSW1, disconnect Host A from port 2/2 and plug in Host B to port 2/2. Try and ping any other hosts in the network as shown in Example 8-51.

Example 8-51 *Testing Connectivity from an Unauthorized Host*

```
C:\WINNT\System32> ping 192.168.1.100
Pinging 192.168.1.101 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.
```

You should not be able to ping anywhere in the network from Host B, because a security violation has occurred and the port blocks frames from unauthorized hosts.

Step 2 On XYZSW2, disconnect the CiscoSecure ACS server from port 0/2 and plug in Host B to port 0/2. Verify the port security status, as shown in Example 8-52.

Example 8-52 *Verifying Port Security Status on XYZSW2*

```
XYZSW2# show port-security
Secure Port MaxSecureAddr CurrentAddr SecurityViolation Security Action
              (Count)       (Count)        (Count)
--------------------------------------------------------------------------------
-------
Fa0/2            1            1                    1              Shutdown
XYZSW2# show interface fastEthernet0/2
FastEthernet0/2 is administratively down, line protocol is down
…
```

The **show port-security** output in Example 8-52 shows the switch has registered a security violation. The **show interface** command indicates the port has been shut down and must manually be re-enabled by issuing the **no shutdown** interface configuration command.

Step 2 Configuring 802.1x Authentication

You now configure 802.1x authentication, which authenticates switch port access based on user credentials rather than MAC address. Cisco's current 802.1x implementation requires the use of a RADIUS server, although the standard allows for any authentication mechanism to be used. In this step:

- Configure XYZSW2 for RADIUS support
- Configure 802.1x support
- Configure the host operating system 802.1x support

Configuring XYZSW2 for RADIUS Support

In Scenario 8-1, you configured RADIUS support on XYZSW1. You now need to configure RADIUS support on XYZSW2 to enable 802.1x authentication.

Step 1 On the CiscoSecure ACS server, add a new NAS definition for XYZSW2 (call it XYZSW2_RADIUS) to use RADIUS, as shown in Figure 8-17. Click on the *Network Configuration* button, then the *XYZSW2* AAA Client, and modify the *Authenticate Using* field to *RADIUS (Cisco IOS/PIX)*. Then click the *Submit + Restart* button to apply your changes.

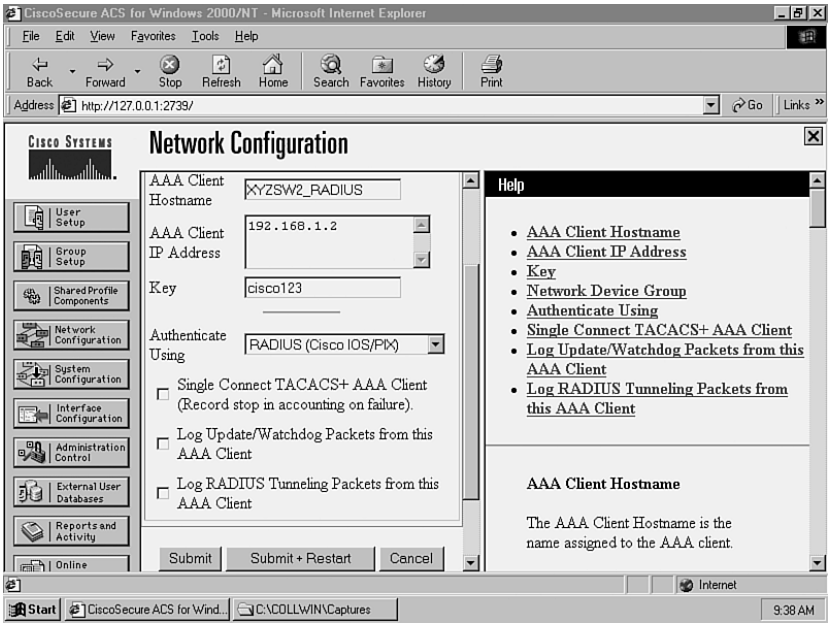
Example 8-54 shows that enabling 802.1x support disables trunking and turns on the spanning-tree PortFast feature on that port.

Step 2 On XYZSW2, configure RADIUS support as shown in Example 8-53.

Example 8-53 *Configuring RADIUS support on XYZSW2*

```
XYZSW2(config)# radius-server host 192.168.1.10 key cisco123
```

Figure 8-17 *Configuring RADIUS Support for XYZSW2*



TIP You can dynamically determine VLAN membership for a user using 802.1x. This is achieved by configuring the following RADIUS attributes on a per-user or per-group basis:

- **[64] Tunnel-Type** indicates the tunnel attributes returned relates to VLANs. For 802.1x, this value must always be VLAN.
- **[65] Tunnel-Medium-Type** indicates the tunnel attributes returned relate to LAN access. For 802.1x, this value must always be 802.
- **[81] Tunnel-Private-Group-ID** indicates the VLAN name (not VLAN ID) the user should be assigned to.

You can also apply per-user ACLs to restrict traffic for a user using either the [11] Filter-ID attribute or the [026/009/001] cisco-av-pair attribute.

Configuring 802.1x Support

Do the following to configure 802.1x support:

Step 1 On XYZSW1, enable 802.1x support globally for the switch, and configure port 2/3 for 802.1x, as shown in Example 8-54.

Example 8-54 *Configuring 802.1x Support on XYZSW1*

```
XYZSW1 (enable) set dot1x system-auth-control enable
dot1x system-auth-control enabled.
XYZSW1 (enable) set port dot1x 2/3 port-control auto
Port 2/3 dot1x port-control is set to auto.
```

continues

Example 8-54 *Configuring 802.1x Support on XYZSW1 (Continued)*

```
Trunking disabled for port 2/3 due to Dot1x feature.
Spantree port fast start option enabled for port 2/3.
XYZSW1 (enable) set port dot1x 2/3 initialize
Port 2/3 initializing...
Port 2/3 dot1x initialization complete.
```

Step 2 On XYZSW1, verify your 802.1x configuration, as shown in Example 8-55.

Example 8-55 *Verifying 802.1x Configuration on XYZSW1*

```
XYZSW1 (enable) show port dot1x 2/3
Port  Auth-State          BEnd-State Port-Control        Port-Status
----- ------------------- ---------- ------------------- ------------
 2/3  connecting          finished   auto                unauthorized
Port  Multiple Host Re-authentication
----- ------------ -----------------
 2/3  disabled      enabled
```

Example 8-55 shows that the port status of the port is unauthorized, indicating the 802.1x client on the port is not present or is providing invalid credentials.

Step 3 On XYZSW2, enable 802.1x support globally for the switch and configure port 0/3 for 802.1x, as shown in Example 8-56.

Example 8-56 *Configuring 802.1x Support on XYZSW2*

```
XYZSW2(config)# aaa authentication dot1x default group radius
XYZSW2(config)# interface fastEthernet0/3
XYZSW2(config-if)# dot1x port-control auto
XYZSW2(config-if)# end
```

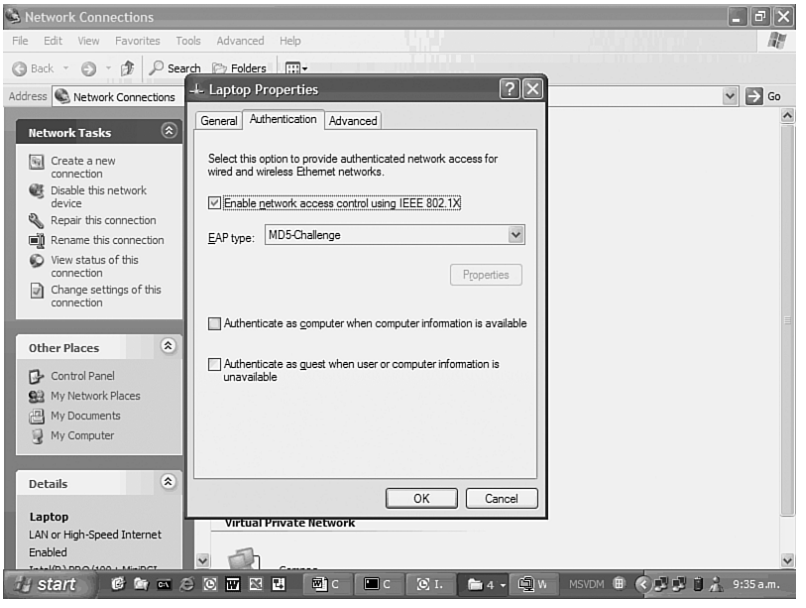
Configuring Host Operating System Support for 802.1x

Configuring host operating system support for 802.1x is done as follows:

Step 1 On Host A, try pinging any other network device (e.g. 192.168.1.1). You should not be able to ping because you have not yet authenticated on the switch port.

Step 2 On Host A, click on Start → All Programs → Accessories → Communications → Network Connections. Right-click on the icon that represents the local LAN interface and select Properties. Next, select the Authentication tab and select the *Enable network access control using IEEE 802.1X* check box, choosing the EAP type as *MD5-Challenge,* as shown in Figure 8-18. Click on the OK button to complete the configuration:

Figure 8-18 *Configuring 802.1x Support on Windows XP*



Step 3 Connect the LAN interface to XYZSW1 port 2/3. You should receive a notification, asking you to authenticate LAN access, as shown in Figure 8-19. Enter *administrator* as the username and *password* as the password, leaving the domain field blank.

Figure 8-19 *Authenticating LAN Access Using 802.1x*

Step 4 Now try pinging another device on the network. If you entered the correct credentials, you should be able to ping okay.

Step 5 Repeat Steps 1 to 4 for Host B. This demonstrates the 802.1x functionality via a Cisco IOS-based switch.

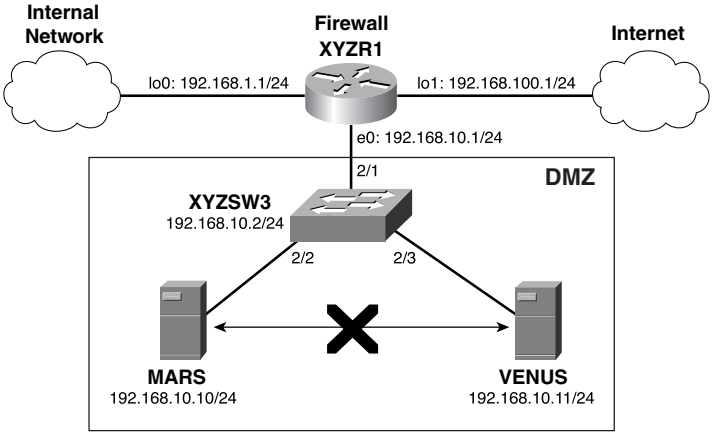
Scenario 8-4: Securing LAN Segments

In this scenario, you enhance the security of a DMZ segment that is used to provide public access from the Internet for web content and Internet e-mail. The DMZ architecture used in this scenario is a very common architecture in use today, so learning how to secure the segment gives you valuable real world experience.

Scenario Exercise

Figure 8-20 illustrates the scenario topology used for this scenario.

Figure 8-20 *Topology*



Corporation XYZ is providing public Internet access for the two external web sites on a DMZ segment, as shown in Figure 8-20. Hosts MARS and VENUS are Windows 2000 web servers. Corporation XYZ wants to provide as much security as possible for this segment and also wants to restrict communications between hosts on the DMZ segment.

The following access control policy must be implemented:

- The DMZ hosts must reside in the same IP subnet, but cannot directly communicate with each other.
- The DMZ hosts must accept inbound HTTP connections.

- The DMZ hosts must be able to issue echo requests and DNS queries to any external network.
- All other access to/from the DMZ hosts must be blocked.

Scenario Objectives

The scenario objectives are as follows:

- Configure private VLANs to prevent communications between hosts on the same IP subnet
- Configure VLAN access control lists (ACLs) to provide access control of traffic flowing between hosts in the same VLAN

Equipment Needed

The equipment needed is as follows:

- One CatOS Catalyst 6000/6500 switch
- One Cisco IOS router with at least one Ethernet interface
- The two hosts in this example are not required, but to verify the objectives you need these two hosts present.

Scenario Planning

Private VLANs are an excellent security feature that allows you to reduce the number of VLANs required to implement access control in the network. By restricting access between devices on the same VLAN (IP subnet), you reduce the need for firewalls with multiple interfaces and the complexity of your firewall security policy. Because the switch provides isolation between devices rather than the firewall, this leaves the firewall free to perform other security functions.

When planning a private VLAN architecture you need to consider the following:

- Which devices require isolation?
- Which devices are to be connected to promiscuous ports?
- Do you have a group of devices that require connectivity within the group, but isolation from the remaining hosts on a segment?

In this scenario, you have two hosts that require isolation from each other. Rather than placing these hosts in separate VLANs and using multiple physical or virtual interfaces on a firewall device to restrict access between the two hosts, you attach each host to an isolated port that can communicate only with the a promiscuous port which connects the firewall. Straight away, you have isolated each host, and even though they are on the same logical IP subnet, they cannot communicate.

One way to bypass the security of private VLANs is to add static host routes on each host, directing frames for another isolated host to the promiscuous port router/firewall, rather than trying to send the frame directly. You can circumvent this issue by using traffic filtering on either the router/firewall or the switch. In this scenario, you use VLAN access control lists on the switch to prevent this.

Command Syntax

This section describes the commands used for configuring private VLANs and VLAN access control lists on the Catalyst 6000/6500 series switches. The following commands are introduced in this scenario:

- The **set vlan** *vlan-id* **pvlan-type** command
- The **set pvlan** command
- The **set security acl** command

The **set vlan** *vlan-id* **pvlan-type** Command

When creating private VLANs, you create the following types of VLANs to implement your required security configuration:

- Primary VLAN
- Isolated VLAN
- Community VLAN

You use the **set vlan** *vlan-id* **pvlan-type** command to create a new VLAN and assign it the appropriate private VLAN role:

```
set vlan vlan-id pvlan-type {primary | isolated | community | twoway-community}
```

The **set pvlan** Command

Once you have created your private VLANs, you must now perform the following tasks:

- Assign ports to isolated and community VLANs.
- Map the isolated and community (secondary) VLANs to the primary VLAN and promiscuous port.

To assign ports to isolated/community VLANs, you use the **set pvlan** command:

```
set pvlan primary-vlan-id secondary-vlan-id {mod/port | sc0}
```

The *mod/port* parameter represents the isolated or community port, and you must repeat the command for each secondary VLAN.

Once primary and secondary VLANs have been created, you use the **set pvlan mapping** command to associate secondary VLANs with a primary VLAN and promiscuous port:

```
set pvlan mapping primary-vlan-id {isolated-vlan | community-vlan} mod/port
```

The *mod/port* parameter represents the promiscuous port, and you must repeat the command for each secondary VLAN.

The **set security acl** Command

The **set security acl** command is used to create VLAN access control lists (VACLs). VACLs allow you to filter upon Layer 3 and Layer 4 parameters and are applied for all inbound access on the entire VLAN. Although very similar to router ACLs, VACLs can be applied only in the inbound direction and can also filter traffic switched within a VLAN. The following tasks are required to configure VACLs:

- Create the VACL
- Commit the VACL to hardware
- Apply the VACL to a VLAN

To create a VACL for IP traffic, you use the **set security acl ip** command:

```
set security acl ip acl-name {permit | deny} {ip | tcp | udp} source destination
```

Example 8-57 shows a VACL called EXAMPLE that filters on IP UDP and TCP traffic.

Example 8-57 *VACL Example*

```
Switch> (enable) set security acl ip EXAMPLE permit ip host 10.1.1.1 any
Switch> (enable) set security acl ip EXAMPLE permit tcp 10.1.1.0 0.0.0.255 any eq 80
Switch> (enable) set security acl ip EXAMPLE permit udp 10.1.1.0 0.0.0.255 any eq 53
Switch> (enable) set security acl ip EXAMPLE deny ip any any
```

Once you have created your VACL, you need to commit the VACL to the Policy Feature Card (PFC), PFC2 or PFC3 on the Catalyst 6000/6500 Supervisor using the **commit security acl** command:

```
commit security acl {vacl-name | all}
```

Finally, you apply the VACL to a particular VLAN using the **set security acl map** command:

```
set security acl map vacl-name vlan
```

Configuration Tasks

The following steps are required to successfully perform the scenario configuration:

- Step 1—Prepare the switch and router
- Step 2—Configure private VLANs and VLAN ACLs
- Step 3—Confirm the desired access control has been achieved

Step 1 Preparing the Switch and Router

In this step, you:

- Configure the system name and IP parameters of the switch
- Configure the router (IP addressing only required)
- Provide connectivity for the router and hosts

Configuring the System Name and IP Parameters of the Switch

Configuring the system name and IP parameters of the switch is done as follows:

Step 1 On the switch, configure the system name, prompt, an IP address of 192.168.10.2/24, and the appropriate default route, as shown in Example 8-58.

Example 8-58 *Configuring Basic Parameters on XYZSW1*

```
Console enable
Enter password: *****
Console (enable) set system name XYZSW3
System name set.
XYZSW3 (enable) set interface sc0 192.168.10.2 255.255.255.0
XYZSW3 (enable) set ip route default 192.168.10.1
Route added.
```

Configuring the Router

Do the following to configuring the router:

Step 1 On the router, configure the system name (XYZR1), Ethernet interfaces, loopback interfaces, and the appropriate IP addressing, as shown in Example 8-59. Refer to Figure 8-20 for the correct IP addressing.

Example 8-59 *Configuring the router*

```
Router(config)# hostname XYZR1
XYZR1(config)# interface ethernet0
XYZR1(config-if)# no shutdown
XYZR1(config-if)# ip address 192.168.10.1 255.255.255.0
XYZR1(config-if)# interface loopback0
XYZR1(config-if)# ip address 192.168.1.1 255.255.255.0
XYZR1(config-if)# interface loopback1
XYZR1(config-if)# ip address 192.168.100.1 255.255.255.0
```

Providing Connectivity for the Router

Step 1 On XYZSW3, configure port 2/1 as 10 Mbps half-duplex and ports 2/2-3 as 100Mbps full-duplex, also naming each port appropriately.

Step 2 Connect the router and hosts with the appropriate cabling to the switch. After at least 30 seconds, ensure that you can ping all hosts and all interfaces on the router as demonstrated in Example 8-60.

Example 8-60 *Confirming Connectivity*

```
XYZSW3> (enable) ping 192.168.10.1
!!!!!
XYZSW3> (enable) ping 192.168.10.10
!!!!!
XYZSW3> (enable) ping 192.168.10.11
!!!!!
XYZSW3> (enable) ping 192.168.1.1
!!!!!
XYZSW3> (enable) ping 192.168.100.1
!!!!!
```

NOTE Try pinging MARS from VENUS or vice versa. Notice that you are able to ping each other because this is normal behavior when both hosts are in the same VLAN, on the same IP subnet with a switch interconnecting the devices. The goal of this scenario is to prevent this intra-VLAN communication using the switch.

Step 2 Configuring Private VLANs and VLAN ACLs

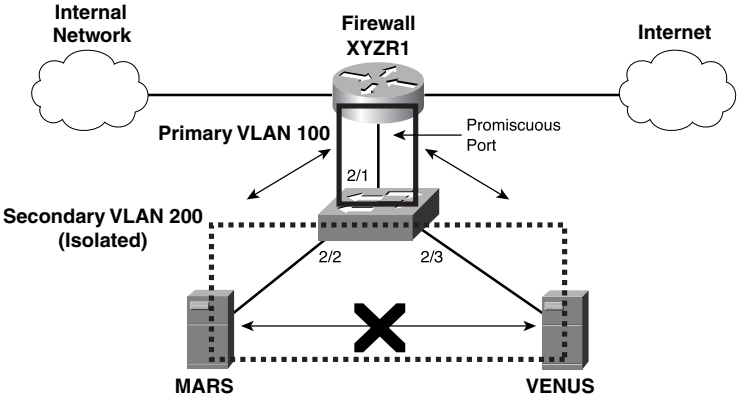
In this step you:

- Configure the required VLANs to implement private VLANs
- Configure the appropriate VLAN ACLs to enforce the required access control

Configuring the Required VLANs to Implement Private VLANs

Figure 8-21 illustrates the VLANs that you use to implement private VLANs.

Figure 8-21 *Private VLANs Used for Scenario 8-4*



VLAN 100 is designated as the primary VLAN, to which any promiscuous ports are assigned. VLAN 200 is designated as a secondary VLAN, to which any isolated ports are assigned. You assign the server ports *and* the management interface port sc0 to the isolated VLAN. This ensures the hosts cannot communicate with each other and also cannot communicate with the switch IP address (if you assigned the sc0 interface to the primary VLAN, it is designated as a promiscuous port and, hence, would be open to communications from the isolated ports).

NOTE In production environments, the switch management interface should not be placed on the same VLAN as users or servers, especially where security is critical.

Step 1 The switch must operate in VTP transparent mode to support private VLANs, so configure this first. Next, create the primary VLAN (100) on XYZSW3, assigning it a private VLAN type of primary, as shown in Example 8-61.

Example 8-61 *Creating the Primary VLAN on XYZSW3*

```
XYZSW3> (enable) set vtp mode transparent
VTP domain  modified
XYZSW3> (enable) set vlan 100 pvlan-type primary
Vlan 100 configuration successful
```

Step 2 Create the secondary VLAN (200) on XYZSW3 assigning it a private VLAN type of *isolated* and assign ports 2/2 and 2/3 to the isolated VLAN. Also assign the management interface sc0 to the isolated VLAN, as shown in Example 8-62.

Example 8-62 *Creating the Secondary (Isolated) VLAN on XYZSW3*

```
XYZSW3> (enable) set vlan 200 pvlan-type isolated
Vlan 200 configuration successful
XYZSW3> (enable) set pvlan 100 200 2/2-3
Successfully set the following ports to Private Vlan 100,200: 2/2-3
XYZSW3> (enable) set pvlan 100 200 sc0
Successfully set the following ports to Private Vlan 100,200: sc0
```

Step 3 Map the secondary (isolated) VLAN to the primary VLAN on the promiscuous port, as shown in Example 8-63.

Example 8-63 *Mapping the Secondary (Isolated) VLAN to the Primary VLAN Promiscuous Port 2/1*

```
XYZSW3> (enable) set pvlan mapping 100 200 2/1
Successfully set mapping between 100 and 200 on 2/1
```

Step 4 Verify your private VLAN configuration by using the **show pvlan** command, as shown in Example 8-64.

Example 8-64 *Verifying the Private VLAN Configuration*

```
XYZSW3> (enable) show pvlan
Primary Secondary Secondary-Type Ports
------- --------- -------------- ------------
100     200       isolated       2/2-3, sc0
XYZSW3> (enable) show pvlan mapping
Port  Primary  Secondary
----- -------- ----------
2/1   100      200
```

The first **show pvlan** command verifies which ports are isolated, while the second **show pvlan mapping** command shows the promiscuous ports.

Step 5 Verify that the private VLANs are working as desired by performing the same ping tests described in Example 8-60.

Configuring the Appropriate VLAN ACLs to Enforce the Required Access Control

You now configure VACLs to provide the required access control on the switch, enhancing the overall security of the architecture. You configure a VACL on the primary VLAN and a VACL on the secondary VACL.

The VACL on the primary VLAN is used to prevent the hosts on the DMZ from routing local traffic (e.g., traffic from MARS to VENUS) to the router to bypass the private VLAN security.

The VACL on the secondary VLAN is used to restrict the services that are allowed for each host. In this scenario, you allow only HTTP and Domain Name System (DNS) traffic.

Step 1 Configure a VACL for the primary VLAN called PROTECT-DMZ on XYZSW3, as shown in Example 8-65.

Example 8-65 *Creating the VACL for the Primary VLAN*

```
XYZSW3> (enable) set security acl ip PROTECT-DMZ permit ip host
  192.168.10.1 192.168.10.0 0.0.0.255
XYZSW3> (enable) set security acl ip PROTECT-DMZ deny ip
  192.168.10.0 0.0.0.255 192.168.10.0 0.0.0.255
XYZSW3> (enable) set security acl ip PROTECT-DMZ permit ip any
  192.168.10.0 0.0.0.255
```

This VACL allows the router to communicate with the DMZ segment, then prevents any hosts on the DMZ from routing local traffic via the router, and then allows the DMZ hosts to communicate with remote networks. Note the importance of the ordering of the VACL.

TIP This VACL is applied to traffic coming from the router only. Traffic from the hosts is not filtered as you might expect (VACLs normally filter inbound traffic to the VLAN), because the VACL is not applied when the secondary to primary VLAN mapping is performed. The reverse applies for the secondary VACL (i.e., only traffic from the hosts is filtered).

Step 2 Configure a secondary VACL called *DMZ-OUT* on XYZSW3, as shown in Example 8-66.

Example 8-66 *Creating the Secondary VACL*

```
XYZSW3> (enable) set security acl ip DMZ-OUT deny icmp any any fragment
XYZSW3> (enable) set security acl ip DMZ-OUT permit tcp host 192.168.10.10
  eq 80 any established
XYZSW3> (enable) set security acl ip DMZ-OUT permit tcp host 192.168.10.11
  eq 80 any established
XYZSW3> (enable) set security acl ip DMZ-OUT permit udp host 192.168.10.10
  any eq 53
XYZSW3> (enable) set security acl ip DMZ-OUT permit udp host 192.168.10.11
  any eq 53
XYZSW3> (enable) set security acl ip DMZ-OUT permit icmp host 192.168.10.10
  any echo
XYZSW3> (enable) set security acl ip DMZ-OUT permit icmp host 192.168.10.11
  any echo
```

This VACL enforces the access control policy for the DMZ. This configuration can be much more effective than filtering on a firewall or router because VACL filtering is performed at wire speed and any dropped packets have no effect on performance (which means denial-of-service attacks can't bring down the switch).

Step 3 Commit the VACLs to hardware and bind them to the appropriate VLANs, as shown in Example 8-67.

Example 8-67 *Committing and Binding the VACLs*

```
XYZSW3> (enable) commit security acl all
ACL commit in progress.
ACL PROTECT-DMZ is committed to hardware.
ACL DMZ-OUT is committed to hardware.
XYZSW3> (enable) set security acl map PROTECT-DMZ 100
ACL PROTECT-DMZ mapped to vlan 100
XYZSW3> (enable) set security acl map DMZ-OUT 200
ACL DMZ-OUT mapped to vlan 200
```

Step 3 Confirm the Desired Access Control Has Been Achieved

The final task is to confirm that you have implemented the correct access control policy. Perform the following traffic tests to verify your configuration:

Step 1 Verify that you can ping all router interfaces from both MARS and VENUS, as shown in Example 8-68 (this verifies that your secondary VACL is allowing outbound ICMP echo requests).

Example 8-68 *Pinging XYZR1 Interfaces from MARS*

```
M:\>ping 192.168.10.1
Pinging 192.168.10.1 with 32 bytes of data:
Reply from 192.168.10.1: bytes=32 time<1ms TTL=255
…
M:\>ping 192.168.1.1
Pinging 192.168.1.1 with 32 bytes of data:
Reply from 192.168.1.1: bytes=32 time<1ms TTL=255
…
M:\>ping 192.168.100.1
Pinging 192.168.100.1 with 32 bytes of data:
Reply from 192.168.100.1: bytes=32 time<1ms TTL=255
…
```

Step 2 Verify that you cannot ping MARS from VENUS and vice versa, as shown in Example 8-69 (this indicates that your private VLAN configuration is working).

Example 8-69 *Pinging VENUS from MARS*

```
M:\>ping 192.168.10.11
Pinging 192.168.10.11 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
```

Step 3 Add a static route on both MARS and VENUS, which routes traffic for the other DMZ host to the router IP address on the DMZ segment (192.168.10.1). For example, on MARS you would add a route defining VENUS (192.168.10.11) as being reachable via XYZR1 (192.168.10.1). Now verify that you still *cannot* ping MARS from VENUS and vice versa, as shown in Example 8-70 (this indicates that your primary VACL is working).

Example 8-70 *Adding a Static Route on MARS and Pinging VENUS from MARS*

```
M:\>route add 192.168.10.11 mask 255.255.255.0 192.168.10.1

M:\>ping 192.168.10.11
Pinging 192.168.10.11 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
```

Step 4 Verify that you can Telnet to MARS and VENUS on port 80 from the router XYZR1, using a source interface address of loopback 1, as shown in Example 8-71 (this verifies that your secondary VACL is allowing the appropriate access). Press Enter a few times once you have connected to get the HTTP Bad Request output shown.

Example 8-71 *Verifying HTTP Connectivity to MARS from XYZR1*

```
XYZR1# telnet 192.168.10.10 80 /source-interface loopback 1
HTTP/1.1 400 Bad Request
Server: Microsoft-IIS/5.1
Date: Sun, 03 Feb 2002 12:03:10 GMT
Content-Type: text/html
Content-Length: 87
<html><head><title>Error</title></head><body>The parameter is incorrect. </body>
</html>
```

Step 5 Verify that you *cannot* perform an extended ping to MARS and VENUS from the router XYZR1 using a source interface address of loopback 1, as shown in Example 8-72 (this verifies that your secondary VACL is blocking any unauthorized services).

Example 8-72 *Verifying ICMP Traffic Is Dropped to MARS from XYZR1*

```
XYZR1# ping ip
Target IP Address: 192.168.10.10
Repeat Count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: y
Source address or interface: 192.168.100.1
…
Sending 5, 100-byte ICMP Echos to 192.168.10.10, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5), round-trip min/avg/max = 0/0/0 ms
```

Summary

In this chapter, you have learned how to secure your switch infrastructure. Securing your switch infrastructure comprises the following key components:

- Securing management access to the switch
- Securing network access
- Implementing traffic filtering

The first step you should take is to secure management access to the switch. Because the switch has substantial control over the network and how traffic is directed, you must ensure it is secure, as secure as possible. Securing management access consists of the following:

- Configuring banners, lockout parameters, and session timeouts
- Configuring user-level authentication and privilege levels
- Using secure protocols such as SSH and SNMPv3 to protect against eavesdropping

Once you have secured your switch, you can place it on your network and implement security features for connecting devices. Port security and 802.1x allow the switch to control access to ports for hosts based upon parameters such as MAC address or a login name and password. The following methods are available for implementing port security:

- **Standard port security**—All port security is configured locally on the switch and is based upon a list of secure MAC addresses for the interface.
- **802.1x security**—Port access is controlled via the use of the IEEE 802.1x standard. The 802.1x standard allows for switch access to be controlled independently of hardware (MAC address) on a per-user basis. 802.1x uses RADIUS to provide centralized authentication and authorization.

Finally, Cisco Catalyst switches include traffic filtering features that allow you to filter traffic based upon Layer 2, 3, and 4 criteria. From a protocol perspective, you can specify that a port forwards only IP, IPX, or AppleTalk/DEC traffic, allowing you to eliminate unnecessary protocols where they are not required. For a more finely grained approach, Catalyst 6000/6500 switches have a VLAN access control list (VACL) feature that filters IP, IPX, or Ethernet traffic at wire speed (requires PFC, PFC2, or PFC3) for an entire VLAN.