Catalyst 6500 Series Switch Software Configuration Guide

Software Release 8.7

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

Text Part Number: OL-8978-04

Catalyst 6500 Series Switch Software Configuration Guide—Release 8.7

CONTENTS

Preface

Revised:OL-8978-04

This preface describes who should read the *Catalyst 6500 Series Switch Software Configuration Guide*, how it is organized, and its document conventions.

Audience

This publication is for experienced network administrators who are responsible for configuring and maintaining Catalyst 6500 series switches.

Organization

Note This publication includes the information that previously was in the *Catalyst 6000 Family Multilayer Switch Feature Card (12.x) and Policy Feature Card Configuration Guide*.

This publication is organized as follows:

Chapter	Title	Description
Chapter 1	Product Overview	Presents an overview of the Catalyst 6500 series switches.
Chapter 2	Command-Line Interfaces	Describes how to use the command-line interface (CLI).
Chapter 3	Configuring the Switch IP Address and Default Gateway	Describes how to perform a baseline configuration of the switch.
Chapter 4	Configuring Ethernet, Fast Ethernet, Gigabit Ethernet, and 10-Gigabit Ethernet Switching	Describes how to configure Ethernet, Fast Ethernet, and Gigabit Ethernet switching.
Chapter 5	Configuring Ethernet VLAN Trunks	Describes how to configure Inter-Switch Link (ISL) and IEEE 802.1Q VLAN trunks on Fast Ethernet and Gigabit Ethernet ports.
Chapter 6	Configuring EtherChannel	Describes how to configure Fast EtherChannel and Gigabit EtherChannel port bundles.

Chapter	Title	Description
Chapter 7	Configuring Spanning Tree	Describes how to configure the Spanning Tree Protocol and explains how spanning tree works.
Chapter 8	Configuring IEEE 802.1Q Tunneling and Layer 2 Protocol Tunneling	Describes how to configure 802.1Q tunneling.
Chapter 9	Configuring Spanning-Tree PortFast, UplinkFast, BackboneFast, and Loop Guard	Describes how to configure the spanning tree PortFast, UplinkFast, and BackboneFast features.
Chapter 10	Configuring VTP	Describes how to configure VLAN Trunking Protocol (VTP) on the switch.
Chapter 11	Configuring VLANs	Describes how to configure VLANs on the switch.
Chapter 12	Configuring InterVLAN Routing	Describes how to configure interVLAN routing on the MSFC.
Chapter 13	Configuring CEF for PFC2 and PFC3A	Describes how to configure Cisco Express Forwarding for Policy Feature Card 2 (CEF for PFC2).
Chapter 14	Configuring MLS	Describes how to configure Multilayer Switching (MLS).
Chapter 15	Configuring Access Control	Describes how to configure access control lists (ACLs).
Chapter 16	Configuring NDE	Describes how to configure NetFlow Data Export (NDE).
Chapter 17	Configuring GVRP	Describes how to configure GARP VLAN Registration Protocol (GVRP) on the switch.
Chapter 18	Configuring MVRP	Describes how to configure Multiple VLAN Registration Protocol (MVRP) on the switch.
Chapter 19	Configuring Dynamic Port VLAN Membership with VMPS	Describes how to configure dynamic port VLAN membership on the switch using the VLAN Management Policy Server (VMPS).
Chapter 20	Checking Status and Connectivity	Describes how to display information about modules and switch ports and how to check connectivity using ping, Telnet, and IP traceroute.
Chapter 21	Configuring GOLD	Describes how to configure the online diagnostics.
Chapter 22	Administering the Switch	Describes how to set the system name, create a login banner, and perform other administrative tasks on the switch.
Chapter 23	Configuring Redundancy	Describes how to install and configure redundant supervisor engines and MSFCs in the Catalyst 6500 series switches.
Chapter 24	Configuring NSF with SSO MSFC Redundancy	Describes how to configure MSFC redundancy using Cisco nonstop forwarding (NSF) with stateful switchover (SSO).
Chapter 25	Modifying the Switch Boot Configuration	Describes how to modify the switch boot configuration, including the BOOT environment variable and the configuration register.
Chapter 26	Working With the Flash File System	Describes how to work with the Flash file system.
Chapter 27	Working with System Software Images	Describes how to download and upload system software images.
Chapter 28	Working with Configuration Files	Describes how to create, download, and upload switch configuration files.
Chapter 29	Configuring System Message Logging	Describes how to configure system message logging (syslog).
Chapter 30	Configuring DNS	Describes how to configure Domain Name System (DNS).
Chapter 31	Configuring CDP	Describes how to configure Cisco Discovery Protocol (CDP).

Chapter	Title	Description
Chapter 32	Configuring UDLD	Describes how to configure the UniDirectional Link Detection (UDLD) protocol.
Chapter 33	Configuring DHCP Snooping and IP Source Guard	Describes how to configure DHCP snooping and IP source guard.
Chapter 34	Configuring NTP	Describes how to configure Network Time Protocol (NTP).
Chapter 35	Configuring Broadcast Suppression	Describes how to configure hardware and software broadcast suppression.
Chapter 36	Configuring Layer 3 Protocol Filtering	Describes how to configure protocol filtering on Ethernet, Fast Ethernet, and Gigabit Ethernet ports.
Chapter 37	Configuring the IP Permit List	Describes how to configure the IP permit list.
Chapter 38	Configuring Port Security	Describes how to configure secure port filtering.
Chapter 39	Configuring the Switch Access Using AAA	Describes how to configure authentication, authorization, and accounting (AAA) to monitor and control access to the CLI.
Chapter 40	Configuring 802.1X Authentication	Describes how to configure 802.1X authentication.
Chapter 41	Configuring MAC Authentication Bypass	Describes how to configure MAC authentication bypass.
Chapter 42	Configuring Web-Based Proxy Authentication	Describes how to configure web-based proxy authentication.
Chapter 43	Tracking Host Aging	Describes how to configure IP device tracking.
Chapter 44	Configuring Network Admission Control	Describes how to configure Network Admission Control (NAC).
Chapter 45	Configuring Unicast Flood Blocking	Describes how to configure unicast flood blocking.
Chapter 47	Configuring SNMP	Describes how to configure SNMP.
Chapter 48	Configuring RMON	Describes how to configure Remote Monitoring (RMON).
Chapter 49	Configuring SPAN, RSPAN and the Mini Protocol Analyzer	Describes how to configure the Switch Port Analyzer (SPAN) and Remote SPAN (RSPAN).
Chapter 50	Using Switch TopN Reports	Describes how to generate switch TopN reports.
Chapter 51	Configuring Multicast Services	Describes how to configure Internet Group Management Protocol (IGMP) snooping, GARP Multicast Registration Protocol (GMRP), and Router Group Management Protocol (RGMP).
Chapter 52	Configuring QoS	Describes how to configure Quality of Service (QoS).
Chapter 53	Using Automatic QoS	Describes how to use the automatic QoS configuration features.
Chapter 54	Configuring ASLB	Describes how to configure accelerated server load balancing (ASLB).
Chapter 46	Configuring the Switch Fabric Modules	Describes how to configure the Switch Fabric Modules.
Chapter 55	Configuring a VoIP Network	Describes how to configure a Voice-over-IP (VoIP) network.
Chapter 56	Configuring the MSFC Cisco IOS Features	Describes how Cisco IOS features that are used with the Catalyst operating system provide feature functionality and parity between these operating systems.
Appendix A	Acronyms	Lists the acronyms used in this publication.

Related Documentation

The following publications are available for the Catalyst 6500 series switches:

- *Catalyst 6500 Series Ethernet Modules Installation Guide*
- *Catalyst 6500 Series Switches Installation Guide*
- *Catalyst 6500 Series Switch Command Reference*
- *ATM Software Configuration and Command Reference—Catalyst 5000 Family and Catalyst 6000 Family Switches*
- *Catalyst 6500 Series Switch System Message Guide*
- *Release Notes for Catalyst 6500 Series Switch Software Release 7.x*
- Cisco IOS Configuration Guides and Command References—Use these publications to help you configure the Cisco IOS software that runs on the MSFC, MSM, and ATM modules.
- For information about MIBs, refer to this URL: http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

Conventions

Note Throughout this publication, except where specifically differentiated, the term *supervisor engine* is used to refer to Supervisor Engine 1, Supervisor Engine 2, and Supervisor Engine 720. The term *MSFC* is used to refer to the MSFC, MSFC2, and MSFC3 except where specifically differentiated.

This publication uses the following conventions:

Convention	Description
boldface font	Commands, command options, and keywords are in **boldface**.
italic font	Arguments for which you supply values are in *italics*.
[ ]	Elements in square brackets are optional.
{ x | y | z }	Alternative keywords are grouped in braces and separated by vertical bars.
[x | y | z]	Optional alternative keywords are grouped in brackets and separated by vertical bars.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.
`screen` font	Terminal sessions and information the system displays are in `screen` font.
`boldface screen` font	Information you must enter is in **`boldface screen`** font.
italic screen font	Arguments for which you supply values are in *italic screen* font.
→	`This pointer highlights an important line of text in an example.`

Convention	Description
^	The symbol ^ represents the key labeled Control—for example, the key combination ^D in a screen display means hold down the Control key while you press the D key.
< >	Nonprinting characters, such as passwords are in angle brackets.

Notes use the following conventions:

Note Means *reader take note*. Notes contain helpful suggestions or references to material not covered in the publication.

Cautions use the following conventions:

Caution Means *reader be careful*. In this situation, you might do something that could result in equipment damage or loss of data.

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, submitting a service request, and gathering additional information, see the monthly *What's New in Cisco Product Documentation*, which also lists all new and revised Cisco technical documentation, at:

http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html

Subscribe to the *What's New in Cisco Product Documentation* as a Really Simple Syndication (RSS) feed and set content to be delivered directly to your desktop using a reader application. The RSS feeds are a free service and Cisco currently supports RSS version 2.0.

CHAPTER 1

Product Overview

The Catalyst 6500 series switches support the following configurations:

- Supervisor Engine 32 with the following onboard components: Policy Feature Card 3B (PFC3B) or PFC3BXL, and Multilayer Switch Feature Card 2A (MSFC2A)
- Supervisor Engine 720 with the following onboard components: PFC3A, PFC3B, or PFC3BXL, MSFC3, and integrated 720-Gbps switch fabric
- Supervisor Engine 2, PFC2, and MSFC2
- Supervisor Engine 2 and PFC2
- Supervisor Engine 1, PFC, and MSFC or MSFC2
- Supervisor Engine 1 and PFC
- Supervisor Engine 1

Refer to the *Release Notes for Catalyst 6500 Series Switch Software Release 8.x* publication for complete information about the chassis, modules, software features, protocols, and MIBs that are supported by the Catalyst 6500 series switches.

Note Throughout this publication, except where specifically differentiated, the term *supervisor engine* is used to refer to Supervisor Engine 1, Supervisor Engine 2, Supervisor Engine 720, and Supervisor Engine 32. The term *MSFC* is used to refer to the MSFC, MSFC2, MSFC2A, and MSFC3 except where specifically differentiated. The term *PFC3* is used to refer to the PFC3A, PFC3B, or PFC3BXL except where specifically differentiated.

Note This publication includes the information that previously was in the *Catalyst 6000 Family Multilayer Switch Feature Card (12.x) and Policy Feature Card Configuration Guide*.

Note For the complete descriptions of all Cisco IOS commands that are used in this publication, refer to the *Catalyst 6500 Series Switch Cisco IOS Command Reference* at this URL:

http://www.cisco.com/en/US/products/hw/switches/ps708/prod_command_reference_list.html

CHAPTER 2

Command-Line Interfaces

This chapter describes the command-line interface (CLI) that you use to configure the Catalyst 6500 series switch modules. For descriptions of all switch and ROM monitor commands, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

Note For a description of the ATM Cisco IOS CLI and commands, refer to the *ATM Software Configuration Guide and Command Reference—Catalyst 5000 Family* and *6000 Family Switches* publication.

This chapter consists of these sections:

Catalyst Command-Line Interface

These sections describe the Catalyst CLI:

ROM-Monitor Command-Line Interface

The ROM monitor is a ROM-based program that executes upon platform power-up, reset, or when a fatal exception occurs. The system enters ROM-monitor mode if the switch does not find a valid system image, if the NVRAM configuration is corrupted, or if the configuration register is set to enter ROM-monitor mode. From the ROM-monitor mode, you can load a system image manually from Flash memory, from a network server file, or from bootflash.

You can enter ROM-monitor mode by restarting the switch and pressing the **Break** key during the first 60 seconds of startup.

Note The Break key is always enabled for 60 seconds after rebooting the system, regardless of whether the Break key is configured to be off by configuration register settings.

To access the ROM monitor through a terminal server, you can escape to the Telnet prompt and enter the **send break** command for your terminal emulation program to break into ROM-monitor mode.

Once you are in ROM-monitor mode, the prompt changes to rommon>. Use the **?** command to see the available ROM-monitor commands.

Switch Command-Line Interface

The switch CLI is a basic command-line interpreter, similar to the UNIX C shell.

These sections describe how to use the switch CLI:

- Accessing the Switch CLI, page 2-2
- Accessing the MSFC from the Switch, page 2-3
- Working With the Command-Line Interface, page 2-5

Accessing the Switch CLI

You can access the CLI through the supervisor engine console port or through a Telnet session.

These sections describe how to access the switch CLI:

- Accessing the CLI through the Console Port, page 2-2
- Accessing the CLI through Telnet, page 2-3

Accessing the CLI through the Console Port

To access the switch CLI through the console port, you must connect a console terminal to the console port through an EIA/TIA-232 (RS-232) cable.

Note For complete information on how to connect to the supervisor engine console port, refer to the hardware documentation for your switch.

To access the switch through the console port, perform this task:

	Task	Command
Step 1	Initiate a connection from the terminal to the switch console prompt and press **Return**.	–
Step 2	At the prompt, enter the system password. The Console> prompt appears, indicating that you have accessed the CLI in normal mode.	–
Step 3	If necessary, enter privileged mode (you must enter privileged mode to change the switch configuration).	**enable**
Step 4	Enter the necessary commands to complete the desired tasks.	–
Step 5	When finished, exit the session.	**exit**

After accessing the switch through the console port, you see this display:

```
Cisco Systems Console
Enter password:
Console>
```

Accessing the CLI through Telnet

Before you can open a Telnet session to the switch, you must first set the IP address for the switch. For information about setting the IP address, see the "Assigning the In-Band (sc0 and sc1) Interface IP Address" section on page 3-7. Up to eight simultaneous Telnet sessions are supported. Telnet sessions disconnect automatically after remaining idle for a set time period.

To access the switch CLI from a remote host using Telnet, perform this task:

	Task	Command
Step 1	From the remote host, enter the **telnet** command and the name or IP address of the switch that you want to access.	**telnet** {*hostname* | *ip_addr*}
Step 2	At the prompt, enter the password for the CLI. If no password has been configured, press **Return**.	–
Step 3	Enter the necessary commands to complete your desired tasks.	–
Step 4	When finished, exit the Telnet session.	**exit**

This example shows how to open a Telnet session to the switch:

```
unix_host% telnet Catalyst_1
Trying 172.16.10.10...
Connected to Catalyst_1.
Escape character is '^]'.

Cisco Systems Console
Enter password:
Catalyst_1>
```

Accessing the MSFC from the Switch

These sections describe how to access the Multilayer Switch Feature Card (MSFC) from a directly connected console port or from a Telnet session:

- Accessing the MSFC from the Console Port, page 2-3
- Accessing the MSFC from a Telnet Session, page 2-4

See the "MSFC Command-Line Interface" section on page 2-8.

Accessing the MSFC from the Console Port

You can enter the **switch console** command to access the MSFC from the switch CLI that is directly connected to the supervisor engine console port. To exit from the MSFC CLI and return to the switch CLI, press **Ctrl-C** three times at the Router> prompt.

To access the MSFC from the switch CLI, perform this task:

Task	Command
Access the MSFC from the switch CLI.	**switch console** [*mod*][1]

1. The *mod* argument specifies the module number of the MSFC. A module number of 15 indicates that the MSFC is installed on the supervisor engine in slot 1. A module number of 16 indicates that the MSFC is installed on the supervisor engine in slot 2. With the Supervisor Engine 720, the *mod* argument specifies the module number of the MSFC3. A module number of 15 indicates that the MSFC3 is installed on the Supervisor Engine 720 in slot 5 (6- or 9-slot switches) or slot 7 (13-slot switches). A module number of 16 indicates that the MSFC3 is installed on the Supervisor Engine 720 in slot 6 (6- or 9-slot switches) or slot 8 (13-slot switches).

Note If no module number is specified, the console will switch to the MSFC on the active supervisor engine.

Note To access the Cisco IOS CLI on the standby MSFC, connect to the console port of the standby supervisor engine.

This example shows how to access the active MSFC from the switch CLI of the active supervisor engine and how to exit the MSFC CLI and return to the switch CLI:

```
Console> (enable) switch console 15
Trying Router-15...
Connected to Router-15.
Type ^C^C^C to switch back...
Router> ^C^C^C
Console> (enable)
```

Accessing the MSFC from a Telnet Session

You can enter the **session** *mod* command to access the MSFC from the switch CLI using a Telnet session. To exit from the MSFC CLI back to the switch CLI, enter **^]** or the **exit** command at the Router> prompt.

Note The *mod* argument specifies the module number of the MSFC. A module number of 15 indicates that the MSFC is installed on the supervisor engine in slot 1. A module number of 16 indicates that the MSFC is installed on the supervisor engine in slot 2. With the Supervisor Engine 720, the *mod* argument specifies the module number of the MSFC3. A module number of 15 indicates that the MSFC3 is installed on the Supervisor Engine 720 in slot 5 (6- or 9-slot switches) or slot 7 (13-slot switches). A module number of 16 indicates that the MSFC3 is installed on the Supervisor Engine 720 in slot 6 (6- or 9-slot switches) or slot 8 (13-slot switches).

This example shows how to access the MSFC from the switch CLI and how to exit the MSFC CLI and return to the switch CLI:

```
Console> (enable) session 15
Trying Router-15...
Connected to Router-15.
Escape character is '^]'.
Router> exit
Console> (enable)
```

Working With the Command-Line Interface

These sections describe how to work with the switch CLI:

For additional information about the CLI, refer to the Command-Line Interfaces chapter in the *Catalyst 6500 Series Switch Command Reference.*

Switch CLI Command Modes

The switch CLI supports two modes of operation: normal and privileged. Both modes are password protected. Enter normal-mode commands for everyday system monitoring. Enter privileged-mode commands to configure the system and perform basic troubleshooting.

After you log in, the system enters normal mode automatically, which gives you access to normal-mode commands only. You can access privileged mode by entering the **enable** command followed by the privileged-mode password. To return to normal mode, enter the **disable** command at the prompt.

This example shows how to enter privileged mode:

```
Console> enable
Enter Password: <password>
Console> (enable)
```

Designating Modules, Ports, and VLANs on the Command Line

Switch commands are not case sensitive. You can abbreviate commands and parameters as long as they contain enough letters to be distinguished from any other currently available commands or parameters.

Catalyst 6500 series switches are multimodule systems. Commands that you enter from the CLI might apply to the entire system or to a specific module, port, or VLAN.

Modules, ports, and VLANs are numbered starting with 1. The supervisor engine is module 1, residing in slot 1. If your switch has a redundant supervisor engine, the supervisor engines reside in slots 1 and 2. The Supervisor Engine 720 is module 5, residing in slot 5 (6- or 9-slot switches) or module 7, residing in slot 7 (13-slot switches). With redundant Supervisor Engine 720s, the Supervisor Engine 720s reside in slots 5 and 6 (6- or 9-slot switches) and slots 7 and 8 (13-slot switches).

To designate a specific module, use the module number.

Port 1 is always the left-most port. To designate a specific port on a specific module, the command syntax is *mod/port.* For example, **3/1** denotes module 3, port 1. In some commands, such as **set trunk** and **set port channel**, you can enter lists of ports.

To specify a range of ports, use a comma-separated list (do not insert spaces) to specify individual ports or use a hyphen (-) between the port numbers to specify a range of ports. Hyphens take precedence over commas.

Table 2-1 shows examples of how to designate ports and port ranges.

Table 2-1 Designating Ports and Port Ranges

Example	Function
`2/1`	Specifies port 1 on module 2.
`3/4-8`	Specifies ports 4, 5, 6, 7, and 8 on module 3.
`5/2,5/4,6/10`	Specifies ports 2 and 4 on module 5 and port 10 on module 6.
`3/1-2,4/8`	Specifies ports 1 and 2 on module 3 and port 8 on module 4.

VLANs are identified using the VLAN ID, which is a single number that is associated with the VLAN. To specify a list of VLANs, use a comma-separated list (do not insert spaces) to specify individual VLANs or a hyphen (-) between the VLAN numbers to specify a range of VLANs.

Table 2-2 shows examples of how to designate VLANs and VLAN ranges.

Table 2-2 Designating VLANs and VLAN Ranges

Example	Function
`10`	Specifies VLAN 10.
`5,10,15`	Specifies VLANs 5, 10, and 15.
`10-50,500`	Specifies VLANs 10 through 50, inclusive, and VLAN 500.

Designating MAC Addresses, IP Addresses, and IP Aliases

Some commands require a MAC address, IP address, or IP alias, which must be designated in a standard format. The MAC address format must be six hexadecimal numbers separated by hyphens, as shown in the following example:

```
00-00-0c-24-d2-fe
```

The IP address format is 32 bits, written as 4 octets separated by periods (dotted decimal format) that are made up of a network section, an optional subnet section, and a host section, as shown in the following example:

```
126.2.54.1
```

If you have configured IP aliases on the switch, you can use IP aliases in place of the dotted decimal IP address. This is true for most commands that use an IP address, except for commands that define the IP address or IP alias. For information on using IP aliases, see the "Defining IP Aliases on the Switch" section on page 22-7.

If DNS is configured on the switch, you can use DNS host names in place of IP addresses. For information on configuring DNS, see Chapter 30, "Configuring DNS."

Command Line Editing

You can scroll through the last 20 commands that are stored in the history buffer, and enter or edit the command at the prompt. Table 2-3 lists the keyboard shortcuts to use when entering and editing switch commands.

Table 2-3 Command-Line Editing Keyboard Shortcuts

Keystroke	Function
Ctrl-A	Jumps to the first character of the command line.
Ctrl-B or the left arrow key	Moves the cursor back one character.
Ctrl-C	Escapes and terminates prompts and tasks.
Ctrl-D	Deletes the character at the cursor.
Ctrl-E	Jumps to the end of the current command line.
Ctrl-F or the right arrow key[1]	Moves the cursor forward one character.
Ctrl-K	Deletes from the cursor to the end of the command line.
Ctrl-L; Ctrl-R	Repeats the current command line on a new line.
Ctrl-N or the down arrow key[1]	Enters the next command line in the history buffer.
Ctrl-P or the up arrow key[1]	Enters the previous command line in the history buffer.
Ctrl-U; Ctrl-X	Deletes from the cursor to the beginning of the command line.
Ctrl-W	Deletes the last word typed.
Esc B	Moves the cursor back one word.
Esc D	Deletes from the cursor to the end of the word.
Esc F	Moves the cursor forward one word.
Delete key or Backspace key	Erases the mistake when entering a command; reenter the command after using this key.

1. The arrow keys function only on ANSI-compatible terminals such as VT100s.

History Substitution

The history buffer stores the last 20 commands that you entered during a terminal session. History substitution allows you to access these commands without retyping them by using special abbreviated commands. Table 2-4 lists the history substitution commands.

Table 2-4 History Substitution Commands

Command	Function
Repeating recent commands:	
!!	Repeat the most recent command.
!-nn	Repeat the *nn*th most recent command.
!n	Repeat the command *n*.
!aaa	Repeat the command beginning with the string *aaa*.
!?aaa	Repeat the command containing the string *aaa*.
To modify and repeat the most recent command:	
^aaa^bbb	Replace the string *aaa* with the string *bbb* in the most recent command.
To add a string to the end of a previous command and repeat it:	
!!aaa	Add the string *aaa* to the end of the most recent command.
!n aaa	Add the string *aaa* to the end of the command *n*.

Table 2-4 History Substitution Commands (continued)

Command	Function
!aaa bbb	Add the string *bbb* to the end of the command beginning with the string *aaa*.
!?aaa bbb	Add the string *bbb* to the end of the command containing the string *aaa*.

Accessing Command Help

Enter **help** or **?** in normal or privileged mode to see the commands that are available in those modes. On selected commands, entering **help** or **?** after a command provides additional information, such as a command usage description. Command usage, the help menu, and when appropriate, parameter ranges are provided if you enter a command using the wrong number of arguments or inappropriate arguments. Additionally, appending **help** or **?** to a command category displays a list of commands in that category.

MSFC Command-Line Interface

These sections describe the MSFC CLI:

- Cisco IOS Command Modes, page 2-8
- Cisco IOS Command-Line Interface, page 2-10

Note In addition to the methods that are described in the "Accessing the MSFC from the Switch" section on page 2-3, you can configure Cisco IOS software to support direct Telnet access to the MSFC. Refer to "Configuring Authentication" in the *Cisco IOS Security Configuration Guide* at this URL: http://www.cisco.com/en/US/docs/ios/12_1/security/configuration/guide/scdathen.html

Cisco IOS Command Modes

The Cisco IOS user interface is divided into many different modes. The commands that are available to you depend on which mode you are currently in. To get a list of the commands in a given mode, type a question mark (?) at the system prompt. For more information, see the "Getting a List of Cisco IOS Commands and Syntax" section on page 2-10.

When you start a session on the switch, you begin in user mode, which is often called user EXEC mode. Only a limited subset of the commands are available in EXEC mode. To have access to all commands, you must enter privileged EXEC mode. Normally, you must type in a password to access privileged EXEC mode. From privileged EXEC mode, you can type in any EXEC command or access global configuration mode. Most of the EXEC commands are one-time commands, such as **show** commands, which show the current configuration status, and **clear** commands, which clear counters or interfaces. The EXEC commands are not saved across reboots of the switch.

The configuration modes allow you to make changes to the running configuration. If you later save the configuration, these commands are stored across switch reboots. You must start at global configuration mode. From global configuration mode, you can enter interface configuration mode, subinterface configuration mode, and a variety of protocol-specific modes.

ROM-monitor mode is a separate mode that is used when the switch cannot boot properly. For example, the switch might enter ROM-monitor mode if it does not find a valid system image when it is booting, or if its configuration file is corrupted at startup. For more information, see the "ROM-Monitor Command-Line Interface" section on page 2-1.

Table 2-5 lists and describes the most commonly used Cisco IOS modes.

Table 2-5 Frequently Used Cisco IOS Command Modes

Mode	Description of Use	How to Access	Prompt
User EXEC	Connect to remote devices, change terminal settings on a temporary basis, perform basic tests, and display system information.	Log in.	`Router>`
Privileged EXEC (enable)	Set operating parameters. The privileged command set includes the commands in user EXEC mode as well as the **configure** command. Use this command to access the other command modes.	From the user EXEC mode, enter the **enable** command and the enable password.	`Router#`
Global configuration	Configure features that affect the system as a whole.	From the privileged EXEC mode, enter the **configure terminal** command.	`Router(config)#`
Interface configuration	Many features are enabled for a particular interface. Interface commands enable or modify the operation of a Gigabit Ethernet or Fast Ethernet interface.	From global configuration mode, enter the **interface** *type location* command.	`Router(config-if)#`
Console configuration	From the directly connected console or the virtual terminal used with Telnet, use this configuration mode to configure the console interface.	From global configuration mode, enter the **line console 0** command.	`Router(config-line)#`

The Cisco IOS command interpreter, called the EXEC, interprets and executes the commands that you enter. You can abbreviate commands and keywords by entering just enough characters to make the command unique from other commands. For example, you can abbreviate the **show** command to **sh** and the **configure terminal** command to **config t**.

When you type **exit**, the switch backs out one level. To exit configuration mode completely and return to privileged EXEC mode, press **Ctrl-Z**.

Getting a List of Cisco IOS Commands and Syntax

In any command mode, you can get a list of available commands by entering a question mark (?).

```
Router> ?
```

To obtain a list of commands that begin with a particular character sequence, type in those characters followed by the question mark (?). Do not include a space. This form of help is called word help, because it completes a word for you.

```
Router# co?
configure
```

To list keywords or arguments, enter a question mark in place of a keyword or argument. Include a space before the question mark. This form of help is called command syntax help, because it reminds you which keywords or arguments are applicable based on the command, keywords, and arguments that you have already entered.

```
Router# configure ?
   memory                Configure from NV memory
   network               Configure from a TFTP network host
   overwrite-network     Overwrite NV memory from TFTP network host
   terminal              Configure from the terminal
```

To redisplay a command that you previously entered, press the up-arrow key or **Ctrl-P**. You can continue to press the up-arrow key to see the last 20 commands that you entered.

Tip If you are having trouble entering a command, check the system prompt, and enter the question mark (?) for a list of available commands. You might be in the wrong command mode or using incorrect syntax.

Press **Ctrl-Z** in any mode to return to privileged EXEC mode. Enter **exit** to return to the previous mode.

Cisco IOS Command-Line Interface

These sections describe basic Cisco IOS configuration tasks that you need to understand before you configure routing:

- Accessing Cisco IOS Configuration Mode, page 2-10
- Viewing and Saving the Cisco IOS Configuration, page 2-11
- Bringing Up an MSFC Interface, page 2-11

Accessing Cisco IOS Configuration Mode

To access the Cisco IOS configuration mode, perform this task:

Note Enter the **switch console** command to access the MSFC from the switch CLI when directly connected to the supervisor engine console port. To access the MSFC from a Telnet session, see the "Accessing the MSFC from a Telnet Session" section on page 2-4.

	Task	Command
Step 1	If you are in the switch CLI, enter the MSFC CLI.	Console> **switch console** [*mod*]
Step 2	At the EXEC prompt, enter enable mode.	Router> **enable**
Step 3	At the privileged EXEC prompt, enter global configuration mode.	Router# **configure terminal**
Step 4	Enter the commands to configure routing.	(Refer to the appropriate configuration tasks later in this chapter.)
Step 5	Exit configuration mode.	Router(config)# **Ctrl-Z**

Viewing and Saving the Cisco IOS Configuration

To view and save the configuration after you make changes, perform this task:

	Task	Command
Step 1	View the current operating configuration at the privileged EXEC prompt.	Router# **show running-config**
Step 2	View the configuration in NVRAM.	Router# **show startup-config**
Step 3	Save the current configuration to NVRAM.	Router# **copy running-config startup-config**

Bringing Up an MSFC Interface

In some cases, an MSFC interface might be administratively shut down. You can check the status of an interface using the **show interface** command.

Note In a redundant supervisor engine setup, if an interface on one MSFC is shut down, the matching VLAN interface on the redundant MSFC will stop forwarding packets. Therefore, you should manually shut down the matching interface on the redundant MSFC.

To bring up an MSFC interface that is administratively shut down, perform this task in privileged mode:

	Task	Command
Step 1	Specify the interface to bring up.	Router(config)# **interface** *interface_type interface_num*
Step 2	Bring the interface up.	Router(config-if)# **no shutdown**
Step 3	Exit configuration mode.	Router(config-if)# **Ctrl-Z**

CHAPTER 3

Configuring the Switch IP Address and Default Gateway

This chapter describes how to configure the IP address, subnet mask, and default gateway on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Understanding How the Switch Management Interfaces Work

Catalyst 6500 series switches have three configurable IP management interfaces, the in-band (sc0 and sc1) interfaces and the out-of-band management Serial Line Internet Protocol (SLIP) (sl0) interface.

The in-band (sc0 and sc1) management interfaces are connected to the switching fabric and participate in all of the functions of a normal switch port, such as spanning tree, Cisco Discovery Protocol (CDP), VLAN membership, and so forth. The out-of-band management interface (sl0) is not connected to the switching fabric and does not participate in any of these functions.

When you configure the IP address, subnet mask, broadcast address, and VLAN membership of the sc0 and sc1 interfaces, you can access the switch through Telnet or Simple Network Management Protocol (SNMP). When you configure the SLIP (sl0) interface, you can open a point-to-point connection to the switch through the console port from a workstation.

All IP traffic that is generated by the switch itself (for example, a Telnet session that is opened from the switch to a host) is forwarded according to the entries in the switch IP routing table. For intersubnetwork communication to occur, you must configure at least one default gateway for the sc0 or sc1 interfaces. The switch IP routing table is used to forward traffic originating on the switch only; the routing table is not used for forwarding traffic that is sent by the devices that are connected to the switch.

Understanding How Automatic IP Configuration Works

These sections describe how the switch can obtain its IP configuration automatically:

- Automatic IP Configuration Overview, page 3-2
- Understanding DHCP, page 3-3
- Understanding BOOTP and RARP, page 3-4

Note These sections apply only to the sc0 interface. The automatic IP configuration features do not apply to the sc1 or sl0 interfaces.

Automatic IP Configuration Overview

The switch can obtain its IP configuration automatically using one of the following protocols:

- Bootstrap Protocol (BOOTP)
- Dynamic Host Configuration Protocol (DHCP)
- Reverse Address Resolution Protocol (RARP)

The switch makes BOOTP, DHCP, and RARP requests only if the sc0 interface IP address is set to 0.0.0.0 when the switch boots up. This address is the default for a new switch or a switch whose configuration file has been cleared using the **clear config all** command. BOOTP, DHCP, and RARP requests are only broadcast out the sc0 interface.

Note If the CONFIG_FILE environment variable is set, all configuration files are processed before the switch determines whether to broadcast BOOTP, DHCP, and RARP requests. For more information about the CONFIG_FILE environment variable, see Chapter 25, "Modifying the Switch Boot Configuration."

Understanding DHCP

There are three methods for obtaining an IP address from the DHCP server:

- Manual allocation—The network administrator maps the switch MAC address to an IP address at the DHCP server.
- Automatic allocation—The switch obtains an IP address when it first contacts the DHCP server. The address is permanently assigned to the switch.
- Dynamic allocation—The switch obtains a “leased” IP address for a specified period of time. The IP address is revoked at the end of this period, and the switch surrenders the address. The switch must request another IP address.

In addition to the sc0 interface IP address, the switch can obtain the subnet mask, broadcast address, and default gateway address. DHCP-learned values are not used if user-configured values are present.

The switch broadcasts a DHCPDISCOVER message 1 to 10 seconds after all of the switch ports are online. The switch always requests an infinite lease time in the DHCPDISCOVER message.

If a DHCP or Bootstrap Protocol (BOOTP) server responds to the request, the switch takes appropriate action. If a DHCPOFFER message is received from a DCHP server, the switch processes all the supported options that are contained in the message. Table 3-1 shows the supported DHCP options. Other options that are specified in the DHCPOFFER message are ignored.

Table 3-1 Supported DHCP Options

Code	Option
1	Subnet mask
2	Time offset
3	Router
6	Domain name server
12	Host name
15	Domain name
28	Broadcast address
33	Static route
42	NTP servers
51	IP address lease time
52	Option overload
61	Client-identifier
66	TFTP server name

If a BOOTP response is received from a BOOTP server, the switch sets the in-band (sc0) interface IP address to the address that is specified in the BOOTP response.

If no DHCPOFFER message or BOOTP response is received in reply, the switch rebroadcasts the request using an exponential backoff algorithm (the amount of time between requests increases exponentially). If no response is received after 10 minutes, the sc0 interface IP address remains set to 0.0.0.0 (if the BOOTP and RARP requests also fail).

If you reset or power cycle a switch with a DHCP- or BOOTP-obtained IP address, the information that is learned from DHCP or BOOTP is retained. At bootup, the switch attempts to renew the lease on the IP address. If no reply is received, the switch retains the current IP address.

Understanding BOOTP and RARP

With BOOTP and RARP, you map the switch MAC address to an IP address on the BOOTP or RARP server. The switch retrieves its IP address from the server automatically when it boots up.

The switch broadcasts 10 BOOTP and RARP requests after all of the switch ports are online. If a response is received, the switch sets the in-band (sc0) interface IP address to the address that is specified in the response.

If no reply is received, the sc0 interface IP address remains set to 0.0.0.0 (if the DHCP requests also fail).

If you reset or power cycle a switch with a BOOTP or RARP-obtained IP address, the information that is learned from BOOTP or RARP is retained.

Preparing to Configure the IP Address and Default Gateway

Before you configure the switch IP address and default gateway, obtain the following information, as appropriate:

- IP address for the switch (sc0 and sc1 interfaces only)
- Subnet mask/number of subnet bits (sc0 and sc1 interfaces only)
- (Optional) Broadcast address (sc0 and sc1 interfaces only)
- VLAN membership (sc0 and sc1 interfaces only)
- SLIP and SLIP destination addresses (sl0 interface only)
- Interface connection type
  - In-band (sc0 and sc1) interfaces: Configure these interfaces when assigning an IP address, subnet mask, and VLAN to the in-band management interface on the switch.
  - SLIP (sl0) interface: Configure this interface when setting up a point-to-point SLIP connection between a terminal and the switch.

Booting the MSFC for the First Time

Two Multilayer Switch Feature Card (MSFC) images are provided on the MSFC bootflash: a boot loader image and a system image. The boot loader image is a limited function system image that has network interface code and end-host protocol code. The system image is the main Cisco IOS software image with full multiprotocol routing support.

As shipped, the MSFC is configured to boot the boot loader image first, which then boots the system image from the bootflash. However, if a Flash PC card is available on the supervisor engine, we recommend that you store all new system images (upgrades) on the supervisor engine Flash PC card instead of the bootflash on the MSFC. The boot loader image *must* stay on the MSFC bootflash.

Caution *Do not erase* the boot loader image; this image must always remain as the first image on the MSFC bootflash as it is always used as the first image to boot.

Note Before you use a system image that is stored on the supervisor engine Flash PC card, set the BOOTLDR environment variable. In privileged mode, enter the **boot bootldr bootflash:***boot_loader_image* command.

To store the system image on the supervisor Flash PC card, change the configuration on the MSFC to boot the MSFC from the appropriate image on the Flash PC card by adding the following command to the MSFC configuration:

boot sup-slot0:*system_image*

In this example, *system_image* is the name of the desired image on the supervisor Flash PC card.

Note To boot a system image that is stored on the supervisor engine Flash PC card, at least one VLAN interface must be configured and active.

By following this recommendation, you do not need to store new system images on the bootflash. If required, you can update the system image on the bootflash from an image on the supervisor engine Flash PC card by entering the following commands:

delete bootflash:*old_system_image*
squeeze bootflash:
copy sup-slot0:*new_system_image* **bootflash:**

Booting from a Melody Compact Flash Adapter Card

Catalyst software release 8.7(1) supports Melody Compact Flash memory, replacing the traditional bootflash memory on Supervisor Engine 720. When a Melody adapter card is detected by the Supervisor engine, the bootdisk file system is loaded instead of the traditional bootflash file system. The configuration commands that list bootflash as an option will list the bootdisk if the Melody adapter card is present. The default location of the crashinfo file is bootdisk instead of bootflash.

The following system messages are displayed when you boot the switch with no Compact Flash memory present or there is faulty Compact Flash memory on the Melody adapter card:

```
2007 Dec 14 05:41:14 %SYS-1-SYS_CF_MSG: No CompactFlash found on adapter card
2007 Dec 14 05:48:07 %SYS-1-SYS_CF_MSG: Faulty CompactFlash found on adapter card
```

If you are using a software version earlier than 8.7(1), which does not have knowledge of the Melody adapter card but the card is present in the system, the following error messages display when bootflash is accessed:

```
Console> (enable) show flash
error = -24
Open device bootflash failed (bad device info block)
Console> (enable) dir bootflash:
error = -24
Open device bootflash failed (bad device info block)
Console> (enable)
```

The following restrictions must be considered when you use bootdisk:

- The ROMMON must be upgraded to 8.4(2) or a later release to boot the switch from the Melody Compact Flash memory card.
- The deleted files cannot be recovered; the **squeeze** command is not supported.

- Standby and active supervisor engines must have the same file system. If the standby supervisor engine has a different file system, it is moved into ROMMON with syslog and nvlog messages when it becomes active.
- The maximum capacity of the Melody Compact Flash memory card can be 1 GB.
- Nonstandard memory cards and cards with varying read and write speeds are not supported.

Default IP Address and Default Gateway Configuration

Table 3-2 shows the default IP address and default gateway configuration.

Table 3-2 Switch IP Address and Default Gateway Default Configuration

Feature	Default Value
In-band (sc0) interface	• IP address, subnet mask, and broadcast address set to 0.0.0.0 • Assigned to VLAN 1
In-band (sc1) interface	• IP address, subnet mask, and broadcast address set to 0.0.0.0 • Assigned to VLAN 2
Default gateway address	Set to 0.0.0.0 with a metric of 0
SLIP[1] (sl0) interface	• IP address and SLIP destination address set to 0.0.0.0 • SLIP for the console port is not active (set to **detach**)

1. SLIP = Serial Line Internet Protocol

Features Supported by the sc0 and sc1 In-Band Interfaces

Table 3-3 lists the features that are supported by the sc0 and sc1 in-band interfaces.

Table 3-3 Feature Support for sc0 and sc1 In-Band Interfaces

sc0 Interface	sc1 Interface
Downloading images	Downloading images
Ping	Ping
Telnet	Telnet
SNMP	SNMP
Default gateway support	Default gateway support
BOOTP	—
DHCP	—
RARP	—

Assigning the In-Band (sc0 and sc1) Interface IP Address

Before you can use Telnet to access the switch or use SNMP to manage the switch, you must assign an IP address to one of the in-band (sc0 or sc1) logical interfaces.

Tip Use the **set interface** {**sc0** | **sc1**} **0.0.0.0** command to set (*clear)* the sc1 or sc0 interfaces back to their default address of 0.0.0.0.

Tip If you configure two inband interfaces, sc0 and sc1, the switch is directly accessible from two different VLANs at the same time.

You can specify the subnet mask (*netmask*) using the number of subnet bits or using the subnet mask in dotted decimal format.

To set the IP address and VLAN membership of the in-band (sc0 or sc1) management interface, perform this task in privileged mode (in this example, the sc0 interface is configured):

	Task	Command
Step 1	Assign an IP address, subnet mask (or number of subnet bits), and (optional) broadcast address to an in-band (sc0 or sc1) interface.	**set interface** {**sc0** | **sc1**} [*ip_addr* [*netmask* [*broadcast*]]] or **set interface** {**sc0** | **sc1**} [*ip_addr*/*netmask* [*broadcast*]]
Step 2	Assign the in-band interface to the proper VLAN (make sure that the VLAN is associated with the network to which the IP address belongs).	**set interface** {**sc0** | **sc1**} [*vlan*]
Step 3	If necessary, bring the interface up.	**set interface** {**sc0** | **sc1**} **up**
Step 4	Verify the interface configuration.	**show interface**

This example shows how to assign an IP address, specify the number of subnet bits, and specify the VLAN assignment for the in-band sc0 interface:

```
Console> (enable) set interface sc0 172.20.52.124/29
Interface sc0 IP address and netmask set.
Console> (enable) set interface sc0 5
Interface sc0 vlan set.
Console> (enable)
```

This example shows how to specify the VLAN assignment, assign an IP address, specify the subnet mask in dotted decimal format, and verify the configuration. In this example, the sc0 interface is configured (the sc1 and sl0 interfaces have not been configured):

```
Console> (enable) set interface sc0 5 172.20.52.124/255.255.255.248
Interface sc0 vlan set, IP address and netmask set.
Console> (enable) show interface
sl0: flags=51<UP,POINTOPOINT,RUNNING>
        slip 0.0.0.0 dest 0.0.0.0
sc0: flags=63<UP,BROADCAST,RUNNING>
        vlan 5 inet 172.20.52.124 netmask 255.255.255.248 broadcast 172.20.52.17
sc1: flags=62<DOWN,BROADCAST,RUNNING>
```

```
        vlan 0 inet 0.0.0.0 netmask 0.0.0.0 broadcast 0.0.0.0
Console> (enable)
```

Configuring the Default Gateways

The supervisor engine sends IP packets that are destined for other IP subnets to the default gateway (typically, a router interface in the same network or subnet as the switch IP address). The switch does not use the IP routing table to forward traffic from connected devices; the switch forwards only IP traffic that is generated by the switch itself (for example, Telnet, TFTP, and ping).

Note In some cases, you might want to configure static IP routes in addition to default gateways. For information on configuring static routes, see the "Configuring Static Routes on the Switch" section on page 22-8.

You can define up to three default IP gateways. Use the **primary** keyword to make a gateway the primary gateway. If you do not specify a primary default gateway, the first gateway that is configured is the primary gateway. If you designate more than one gateway as primary, the last primary gateway that is configured is the primary default gateway.

The switch sends all off-network IP traffic to the primary default gateway. If connectivity to the primary gateway is lost, the switch attempts to use the backup gateways in the order that they were configured. The switch sends periodic ping messages to determine whether each default gateway is up or down. If connectivity to the primary gateway is restored, the switch resumes sending traffic to the primary gateway.

Note The system automatically associates routes and gateways to the appropriate sc0 or sc1 in-band interface.

To configure one or more default gateways, perform this task in privileged mode:

	Task	Command
Step 1	Configure a default IP gateway address for the switch.	**set ip route default** *gateway* [*metric*] [**primary**]
Step 2	(Optional) Configure additional default gateways for the switch.	**set ip route default** *gateway* [*metric*] [**primary**]
Step 3	Verify that the default gateways appear correctly in the IP routing table.	**show ip route**

To remove the default gateway entries, perform one of these tasks in privileged mode:

Task	Command
Clear an individual default gateway entry.	**clear ip route default** *gateway*
Clear all default gateways and static routes.	**clear ip route all**

This example shows how to configure three default gateways on the switch and verify the default gateway configuration:

```
Console> (enable) set ip route default 10.1.1.10
Route added.
Console> (enable) set ip route default 10.1.1.20
Route added.
Console> (enable) set ip route default 10.1.1.1 primary
Route added.
Console> (enable) show ip route
Fragmentation   Redirect   Unreachable
-------------   --------   -----------
enabled         enabled    enabled

The primary gateway: 10.1.1.1
Destination      Gateway          RouteMask    Flags   Use       Interface
---------------  ---------------  ----------   -----   --------  ---------
default          10.1.1.1         0x0          UG      6           sc0
default          10.1.1.20        0x0          G       0           sc0
default          10.1.1.10        0x0          G       0           sc0
10.0.0.0         10.1.1.100       0xff000000   U       75          sc0
default          default          0xff000000   UH      0           sl0
Console> (enable)
```

Configuring the SLIP (sl0) Interface on the Console Port

Use the SLIP (sl0) interface for point-to-point SLIP connections between the switch and an IP host.

Caution You *must* use the console port for the SLIP connection. When the SLIP connection is enabled and SLIP is attached on the console port, an EIA/TIA-232 terminal cannot connect through the console port. If you are connected to the switch CLI through the console port and you enter the **slip attach** command, you will lose the console port connection. Use Telnet to access the switch, enter privileged mode, and enter the **slip detach** command to restore the console port connection.

To enable and attach SLIP on the console port, perform this task:

	Task	Command
Step 1	Access the switch from a remote host with Telnet.	**telnet** {*host_name* | *ip_addr*}
Step 2	Enter privileged mode on the switch.	**enable**
Step 3	Set the console port SLIP address and the destination address of the attached host.	**set interface sl0** *slip_addr dest_addr*
Step 4	Verify the SLIP interface configuration.	**show interface**
Step 5	Enable SLIP for the console port.	**slip attach**

To disable SLIP on the console port, perform this task:

	Task	Command
Step 1	Access the switch from a remote host with Telnet.	**telnet** {*host_name* | *ip_addr*}
Step 2	Enter privileged mode on the switch.	**enable**
Step 3	Disable SLIP for the console port.	**slip detach**

This example shows how to configure SLIP on the console port and verify the configuration:

```
sparc20% telnet 172.20.52.38
Trying 172.20.52.38 ...
Connected to 172.20.52.38.
Escape character is '^]'.

Cisco Systems, Inc. Console

Enter password:
Console> enable

Enter password:
Console> (enable) set interface sl0 10.1.1.1 10.1.1.2
Interface sl0 slip and destination address set.
Console> (enable) show interface
sl0: flags=51<UP,POINTOPOINT,RUNNING>
        slip 10.1.1.1 dest 10.1.1.2
sc0: flags=63<UP,BROADCAST,RUNNING>
        vlan 522 inet 172.20.52.38 netmask 255.255.255.240 broadcast 172.20.52.7
Console> (enable) slip attach
Console Port now running SLIP.

Console> (enable) slip detach
SLIP detached on Console port.
Console> (enable)
```

Using BOOTP, DHCP, or RARP to Obtain an IP Address

Note For complete information on how the switch uses BOOTP, DHCP, or RARP to obtain its IP configuration, see the "Understanding How Automatic IP Configuration Works" section on page 3-2.

To use BOOTP, DHCP, or RARP to obtain an IP address for the switch, perform this task:

	Task	Command
Step 1	Make sure that there is a DHCP, BOOTP, or RARP server on the network.	–
Step 2	Obtain the last address in the MAC address range for module 1 (the supervisor engine). This address is displayed under the MAC-Address(es) heading. (With DHCP, this step is necessary only if using the manual allocation method.)	**show module**
Step 3	Add an entry for each switch in the DHCP, BOOTP, or RARP server configuration, mapping the MAC address of the switch to the IP configuration information for the switch. (With DHCP, this step is necessary only if using the manual or automatic allocation methods.)	–
Step 4	Set the sc0 interface IP address to 0.0.0.0.	**set interface sc0 0.0.0.0**

	Task	Command
Step 5	Reset the switch. The switch broadcasts DHCP and RARP requests only when the switch boots up.	**reset system**
Step 6	When the switch reboots, confirm that the sc0 interface IP address, subnet mask, and broadcast address are set correctly.	**show interface**
Step 7	For DHCP, confirm that other options (such as the default gateway address) are set correctly.	**show ip route**

This example shows the switch broadcasting a DHCP request, receiving a DHCP offer, and configuring the IP address and other IP parameters according to the contents of the DHCP offer:

```
Console> (enable)
Sending RARP request with address 00:90:0c:5a:8f:ff
Sending DHCP packet with address: 00:90:0c:5a:8f:ff
dhcpoffer
Sending DHCP packet with address: 00:90:0c:5a:8f:ff
Timezone set to '', offset from UTC is 7 hours 58 minutes
Timezone set to '', offset from UTC is 7 hours 58 minutes
172.16.30.32 added to DNS server table as primary server.
172.16.31.32 added to DNS server table as backup server.
172.16.32.32 added to DNS server table as backup server.
NTP server 172.16.25.253 added
NTP server 172.16.25.252 added
%MGMT-5-DHCP_S:Assigned IP address 172.20.25.244 from DHCP Server 172.20.25.254
Console> (enable) show interface
sl0: flags=51<UP,POINTOPOINT,RUNNING>
        slip 0.0.0.0 dest 0.0.0.0
sc0: flags=63<UP,BROADCAST,RUNNING>
        vlan 1 inet 172.20.25.244 netmask 255.255.255.0 broadcast 172.20.25.255
dhcp server: 172.20.25.254
Console>
```

Renewing and Releasing a DHCP-Assigned IP Address

If you are using DHCP for IP address assignment, you can perform either of these DHCP-related tasks:

- Renew the lease on a DHCP-assigned IP address
- Release the lease on a DHCP-assigned IP address

To renew or release a DHCP-assigned IP address on the in-band (sc0) management interface, perform one of these tasks in privileged mode:

Task	Command
Renew the lease on a DHCP-assigned IP address.	**set interface sc0 dhcp renew**
Release the lease on a DHCP-assigned IP address.	**set interface sc0 dhcp release**

This example shows how to renew the lease on a DHCP-assigned IP address:

```
Console> (enable) set interface sc0 dhcp renew
Renewing IP address...
Console> (enable) Sending DHCP packet with address: 00:90:0c:5a:8f:ff
```

```
<...output truncated...>
```

This example shows how to release the lease on a DHCP-assigned IP address:

```
Console> (enable) set interface sc0 dhcp release
Releasing IP address...
Console> (enable) Sending DHCP packet with address: 00:90:0c:5a:8f:ff
Done

Console> (enable)
```

CHAPTER 4

Configuring Ethernet, Fast Ethernet, Gigabit Ethernet, and 10-Gigabit Ethernet Switching

This chapter describes how to use the command-line interface (CLI) to configure Ethernet, Fast Ethernet, Gigabit Ethernet, and 10-Gigabit Ethernet switching on the Catalyst 6500 series switches. The configuration tasks in this chapter apply to Ethernet, Fast Ethernet, Gigabit Ethernet, and 10-Gigabit Ethernet switching modules, as well as to the uplink ports on the supervisor engine.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Understanding How Ethernet Works

Catalyst 6500 series switches support simultaneous, parallel connections between Ethernet segments. Switched connections between Ethernet segments last only for the duration of the packet. New connections can be made between different segments for the next packet.

Catalyst 6500 series switches solve congestion problems that are caused by high-bandwidth devices and a large number of users by assigning each device (for example, a server) to its own 10-, 100-, 1000-, or 10000-Mbps segment. Because each Ethernet port on the switch represents a separate Ethernet segment, the servers that are in a properly configured switched environment achieve full access to the bandwidth.

Because collisions are a major bottleneck in the Ethernet networks, an effective solution is full-duplex communication, which is an option for any 10- or 100-Mbps port on a Catalyst 6500 series switch (the Gigabit Ethernet and 10-Gigabit Ethernet ports are always full duplex). Typically, Ethernet operates in half-duplex mode, which means that stations can either receive or transmit. In full-duplex mode, two stations can transmit and receive at the same time. When packets can flow in both directions simultaneously, effective Ethernet bandwidth doubles to 20 Mbps for 10-Mbps ports and to 200 Mbps for Fast Ethernet ports. The Gigabit Ethernet and 10-Gigabit Ethernet ports on the Catalyst 6500 series switches are full duplex only (2-Gbps and 20-Gbps effective bandwidth, respectively).

These sections describe Ethernet:

Switching Frames Between Segments

Each Ethernet port on a Catalyst 6500 series switch can connect to a single workstation or server or to a hub through which workstations or servers connect to the network.

Ports on a typical Ethernet hub all connect to a common backplane within the hub, and the bandwidth of the network is shared by all devices that are attached to the hub. If two stations establish a session that uses a significant level of bandwidth, the network performance of all other stations that are attached to the hub is degraded.

To reduce degradation, the switch treats each port as an individual segment. When the stations on different ports need to communicate, the switch forwards the frames from one port to the other port at wire speed to ensure that each session receives full bandwidth.

To switch frames between ports efficiently, the switch maintains an address table. When a frame enters the switch, it associates the MAC address of the sending station with the port on which it was received.

Building the Address Table

Catalyst 6500 series switches build the address table by using the source address of the received frames. When the switch receives a frame for a destination address that is not listed in its address table, it floods the frame to all ports of the same VLAN except for the port that received the frame. When the destination station replies, the switch adds its relevant source address and port ID to the address table. The switch then forwards subsequent frames to a single port without flooding to all ports.

The address table can store at least 32,000 address entries without flooding any entries. The switch uses an aging mechanism, which is defined by a configurable aging timer, so if an address remains inactive for a specified number of seconds, it is removed from the address table.

Understanding Port Negotiation

Note The **set port negotiation** command is supported on Gigabit Ethernet ports only; it is not supported on the WS-X6316-GE-TX and WS-X6516-GE-TX modules. If a port does not support this command, this message appears: “Feature not supported on Port N/N,” where N/N is the module and the port number.

Note You cannot configure port negotiation on 1000BASE-TX (copper) Gigabit Ethernet ports in this release. If you insert a 1000BASE-TX GBIC in the port that was previously configured as negotiation disabled, the negotiation-disabled setting is ignored and the port operates in negotiation-enabled mode.

Note Port negotiation does not involve negotiating port speed. You cannot disable port negotiation with the **set port speed** command.

Port negotiation exchanges flow-control parameters, remote fault information, and duplex information. Configure port negotiation with the **set port negotiation** command. Port negotiation is enabled by default.

Note When you enable port negotiation on the 16-port 10/100/1000BASE-T Ethernet modules, the system autonegotiates flow control only.

The ports on both ends of a link must have the same setting. The link will not come up if the ports at each end of the link are set inconsistently (port negotiation is enabled on one port and is disabled on the other port).

Table 4-1 shows the four possible port negotiation configurations and the resulting link status for each configuration.

Table 4-1 Port Negotiation Configuration and Possible Link Status

Port Negotiation State		Link Status	
Near End[1]	Far End[2]	Near End	Far End
Off	Off	Up	Up
On	On	Up	Up
Off	On	Up	Down
On	Off	Down	Up

1. Near End refers to the local port.
2. Far End refers to the port at the other end of the link.

Default Ethernet, Fast Ethernet, Gigabit Ethernet, and 10-Gigabit Ethernet Configuration

Table 4-2 shows the Ethernet, Fast Ethernet, Gigabit Ethernet, and 10-Gigabit Ethernet default configuration.

Table 4-2 Ethernet Default Configuration

Feature	Default Value
Port enable state	All ports are enabled
Port name	None

Table 4-2 Ethernet Default Configuration (continued)

Feature	Default Value
Duplex mode	• Half duplex for 10-Mbps Ethernet ports • Autonegotiate speed and duplex for 10/100-Mbps Fast Ethernet ports • Autonegotiate duplex for 100-Mbps Fast Ethernet ports • Full duplex for 1000-Mbps Gigabit Ethernet ports • Full duplex for 10000-Mbps Gigabit Ethernet ports
Flow control (10-Gigabit Ethernet)	Flow control set to on for receive (Rx) and off for transmit (Tx)[1]
Flow control (Gigabit Ethernet)	Flow control set to off for receive (Rx) and desired for transmit (Tx)
Flow control (other Ethernet)	Flow control set to off for receive (Rx); transmit (Tx) not supported
Spanning Tree Protocol (STP)	Enabled for VLAN 1
Native VLAN	VLAN 1
Port VLAN cost	• 100 for 10-Mbps Ethernet ports • 19 for 10/100-Mbps Fast Ethernet ports • 19 for 100-Mbps Fast Ethernet ports • 4 for 1000-Mbps Gigabit Ethernet ports • 1 for 10000-Mbps Gigabit Ethernet ports
EtherChannel	Disabled on all Ethernet ports
Jumbo frames	Disabled on all Ethernet ports

1. On WS-X6502-10-Gigabit Ethernet ports, flow control on the receive side is always on and cannot be set to off. On WS-X6704, WS-X6708-10-Gigabit Ethernet ports and Supervisor Engine 720-10 Gigabit Ethernet uplink ports flowcontrol for both send and receive side can be set to off.

Setting the Port Configuration

These sections describe how to configure Ethernet, Fast Ethernet, Gigabit Ethernet, and 10-Gigabit Ethernet switching on the Catalyst 6500 series switches:

Configuring Supervisor Engine 720 Ports

Supervisor Engine 720, port 1 has a small form-factor pluggable (SFP) connector and no unique configuration options.

Note Cisco WS-X6408A-GBIC, which is an 8-port Gigabit Ethernet interface module for the Catalyst 6500 Series switches, is supported on Supervisor Engine 720.

Supervisor Engine 720, port 2 has an RJ-45 connector and an SFP connector (default). To use the RJ-45 connector, you must change the configuration.

To configure port 2 on Supervisor Engine 720, perform this task in privileged mode:

Task	Command
Configure port 2 on Supervisor Engine 720.	**set port media-type** *mod/port* {**rj45** | **sfp**}

This example shows how to configure port 2 on Supervisor Engine 720 to use the RJ-45 connector:

```
Console> (enable) set port media-type 5/2 rj45
Port 5/2 media type set to RJ-45.
Console> (enable)
```

Setting the Port Name

You can set the port names on Ethernet, Fast Ethernet, Gigabit Ethernet, and 10-Gigabit Ethernet switching modules to facilitate switch administration.

To set the port name, perform this task in privileged mode:

	Task	Command
Step 1	Set a port name.	**set port name** *mod/port* [*name_string*]
Step 2	Verify that the port name is configured.	**show port** [*mod*[*/port*]]

This example shows how to set the name for ports 1/1 and 1/2 and verify that the port names are configured correctly:

```
Console> (enable) set port name 1/1 Router Connection
Port 1/1 name set.
Console> (enable) set port name 1/2 Server Link
Port 1/2 name set.
```

```
Console> (enable) show port 1
Port  Name               Status     Vlan         Duplex Speed Type
----- ------------------ ---------- ----------   ------ ----- ------------
 1/1  Router Connection  connected  trunk          full  1000 1000BaseSX
 1/2  Server Link        connected  trunk          full  1000 1000BaseSX
.
.
.
Last-Time-Cleared
--------------------------
Wed Jun 16 1999, 16:25:57
Console> (enable)
```

Setting the Port Speed

You can configure the port speed on 10/100-Mbps Ethernet switching modules. Use the **auto** keyword to autonegotiate the port's speed and duplex mode with the neighboring port.

Note If the port speed is set to **auto** on a 10/100-Mbps Ethernet port, both speed and duplex are autonegotiated.

Use the **auto-10-100** keyword on ports that support speeds of 10/100/1000 Mbps. Using the **auto-10-100** keyword makes the port behave the same as a 10/100-Mbps port that has the speed set to **auto**. The speed and duplex are negotiated (the 1000-Mbps speed does not take part in the negotiation).

To set the port speed of an Ethernet port, perform this task in privileged mode:

	Task	Command
Step 1	Set the port speed of an Ethernet port.	**set port speed** *mod/port* {**10** | **100** | **1000** | **auto** | **auto-10-100**}
Step 2	Verify that the speed of the port is configured correctly.	**show port** [*mod*[*/port*]]

This example shows how to set the port speed to 100 Mbps on port 2/2:

```
Console> (enable) set port speed 2/2 100
Port 2/2 speed set to 100 Mbps.
Console> (enable)
```

This example shows how to make port 2/1 autonegotiate speed and duplex with the neighboring port:

```
Console> (enable) set port speed 2/1 auto
Port 2/1 speed set to auto-sensing mode.
Console> (enable)
```

Setting the Port Duplex Mode

You can set the port duplex mode to full or half duplex for Ethernet and Fast Ethernet ports.

Note Gigabit Ethernet and 10-Gigabit Ethernet are full duplex only. You cannot change the duplex mode on Gigabit Ethernet and 10-Gigabit Ethernet ports.

Note If the port speed is set to **auto** on a 10/100-Mbps Ethernet port, both speed and duplex are autonegotiated. You cannot change the duplex mode of autonegotiation ports.

To set the duplex mode of a port, perform this task in privileged mode:

	Task	Command
Step 1	Set the duplex mode of a port.	**set port duplex** *mod/port* {**full** | **half**}
Step 2	Verify that the duplex mode of the port is configured correctly.	**show port** [*mod*[*/port*]]

This example shows how to set the duplex mode to half duplex on port 2/1:

```
Console> (enable) set port duplex 2/1 half
Port 2/1 set to half-duplex.
Console> (enable)
```

Enabling or Disabling Auto-MDI/MDIX

With auto-MDI/MDIX you can use either a straight or crossover cable, and the module will automatically detect and adjust for the cable type. Auto-MDI/MDIX works with the speed set to auto/1000 Mbps but not with the speed set to 10 Mbps or 100 Mbps. The link will come up with either a straight or crossover cable if the speed is set to auto/1000 using the **set port speed** *mod/port* **auto** command or the **set port speed** *mod/port* **1000** command. The link comes up even if the speed is autonegotiated at 10 Mbps or 100 Mbps in **auto** mode. However, if you enter the **set port speed** *mod/port* **10** command or the **set port speed** *mod/port* **100** command, the link fails to come up if the wrong cable is used.

Auto-MDI/MDIX has always been enabled on the following modules:

- WS-X6548-RJ-45, WS-X6548-RJ-21, WS-X6148-GE-TX, WS-X6548-GE-TX

  Auto-MDI/MDIX works in 10-, 100-, and 1000-Mbps modes with autonegotiated and fixed speeds.
- WS-X6516-GE-TX

  Auto-MDI/MDIX works with the speed set to auto/1000 Mbps but not with the speed set to 10 Mbps or 100 Mbps.
- WS-X6316-GE-TX

With software release 8.2(1) and later releases, auto-MDIX is also enabled on the following modules:

- WS-X6748-GE-TX, Supervisor Engine 720 port 2 (RJ-45)

  Auto-MDI/MDIX works with the speed set to auto/1000 but not with the speed set to 10 Mbps or 100 Mbps
- WS-X6148X2-RJ-45, WS-X6148X2-45AF

  Auto-MDI/MDIX works with the speed set to auto but not with the speed set to 10 Mbps or 100 Mbps.

Note Auto-MDI/MDIX is not supported on any other 10/100-Mbps Ethernet modules or GBIC, SFP, and XENPAK ports.

With software release 8.3(1) and later releases, the **set port auto-mdix** *mod/port* {**enable** | **disable**} command is introduced to disable auto-MDI/MDIX on all the modules that currently have this feature enabled by default. Use the **show port auto-mdix** [*mod*[/*port*]] command to display auto-MDI/MDIX settings.

Configuring IEEE 802.3x Flow Control

Gigabit Ethernet and 10-Gigabit Ethernet ports on the Catalyst 6500 series switches use flow control to inhibit the transmission of packets to the port for a period of time; other Ethernet ports use flow control to respond to flow-control requests.

If a Gigabit Ethernet or a 10-Gigabit Ethernet port receive buffer becomes full, the port transmits a "pause" packet that tells remote ports to delay sending more packets for a specified period of time. All Ethernet ports (10000 Mbps, 1000 Mbps, 100 Mbps, and 10 Mbps) can receive and act upon "pause" packets from other devices.

Enter the **set port flow control** command to configure flow control on ports. Table 4-3 lists the **set port flowcontrol** command keywords and describes their functions.

Table 4-3 Ethernet-Flow Control Keyword Functions

Keywords	Function
receive on[1]	The port uses flow control dictated by the neighboring port.
receive desired	The port uses flow control if the neighboring port uses it and does not use flow control if the neighboring port does not use it.
receive off	The port does not use flow control, regardless of whether flow control is requested by the neighboring port.
send on[2]	The port sends flow-control frames to the neighboring port.
send desired[2]	The port sends flow-control frames to the port if the neighboring port asks to use flow control.
send off[2]	The port does not send flow-control frames to the neighboring port.

1. On WS-X6502-10-Gigabit Ethernet ports, flow control on the receive side is always on and cannot be set to off. On WS-X6704, WS-X6708-10-Gigabit Ethernet ports and Supervisor Engine 720-10 Gigabit Ethernet uplink ports flowcontrol for both send and receive side can be set to off.
2. Supported only on Gigabit Ethernet and 10-Gigabit Ethernet ports.

To configure flow control, perform this task in privileged mode:

	Task	Command
Step 1	Set the flow-control parameters.	**set port flowcontrol** *mod/port* {**receive** | **send**} {**off** | **on** | **desired**}
Step 2	Verify the flow-control configuration.	**show port flowcontrol**

This example shows how to turn on transmit and receive flow control and verify the flow-control configuration:

```
Console> (enable) set port flowcontrol 3/1 send on
Port 3/1 will send flowcontrol to far end.
```

```
Console> (enable) set port flowcontrol 3/1 receive on
Port 3/1 will require far end to send flow control
Console> (enable) show port flowcontrol
Port  Send-Flowcontrol  Receive-Flowcntl  RxPause  TxPause
      Admin   Oper      Admin   Oper
----- ----------------  ----------------  -------  -------
 3/1  on      disagree  on      disagree  0        0
 3/2  off     off       off     off       0        0
 3/3  desired on        desired off       10       10
Console> (enable)
```

Enabling and Disabling Port Negotiation

To enable port negotiation, perform this task in privileged mode:

	Task	Command
Step 1	Enable port negotiation.	**set port negotiation** *mod/port* **enable**
Step 2	Verify the port negotiation configuration.	**show port negotiation** [*mod/port*]

This example shows how to enable port negotiation and verify the configuration:

```
Console> (enable) set port negotiation 2/1 enable
Port 2/1 negotiation enabled
Console> (enable) show port negotiation 2/1
Port   Link Negotiation
-----  ----------------
 2/1   enabled
Console> (enable)
```

To disable port negotiation, perform this task in privileged mode:

	Task	Command
Step 1	Disable port negotiation.	**set port negotiation** *mod/port* **disable**
Step 2	Verify the port negotiation configuration.	**show port negotiation** [*mod/port*]

This example shows how to disable port negotiation and verify the configuration:

```
Console> (enable) set port negotiation 2/1 disable
Port 2/1 negotiation disabled
Console> (enable) show port negotiation 2/1
Port   Link Negotiation
-----  ----------------
 2/1   disabled
Console> (enable)
```

Changing the Default Port Enable State

Note Changing the default port enable state applies to all port types, not just Ethernet.

When you enter the **clear config all** command or in the event of a configuration loss, all ports collapse into VLAN 1. This situation might cause a security and network instability problem. Entering the **set default portstatus** command puts all ports into a disable state and blocks the traffic flowing through the ports during a configuration loss. You can then manually configure the ports back to the enable state.

The default port status configuration is stored on the chassis. The configuration is tied to a chassis and not to the supervisor engine. The **clear config all** command uses this setting to determine whether ports should be enabled or disabled when returning to default configuration. The **clear config all** command does not change the default port status setting on the chassis. The output of the **show config** command shows the current default port status configuration.

To change the port enable state, perform this task in privileged mode:

	Task	Command
Step 1	Change the port enable state.	**set default portstatus** {**enable** | **disable**}
Step 2	Display the port enable state.	**show default**

This example shows how to change the default port enable state from enabled to disabled:

```
Console> (enable) set default portstatus disable
Default port status set to disable.
Console> (enable)
```

This example shows how to display the port enable state:

```
Console> (enable) show default
portstatus: disable
Console> (enable)
```

Setting the Port Debounce Timer

You can set the port debounce timer on a per-port basis for Ethernet, Fast Ethernet, Gigabit Ethernet, and 10-Gigabit Ethernet ports. When you set the port debounce timer, the switch delays notifying the main processor of a link change that can decrease traffic loss due to network reconfiguration.

Caution Enabling the port debounce timer causes link up and link down detections to be delayed, resulting in loss of data traffic during the debouncing period. This situation might affect the convergence and reconvergence of various Layer 2 and Layer 3 protocols.

Table 4-4 lists the time delay that occurs before the switch notifies the main processor of a link change before and after the switch enables the debounce timer.

Table 4-4 Port Debounce Timer Delay Time

Port Type	Debounce Timer Disabled	Debounce Timer Enabled
10BASE-FL ports	300 milliseconds	3100 milliseconds
10/100BASE-TX ports	300 milliseconds	3100 milliseconds
100BASE-FX ports	300 milliseconds	3100 milliseconds
10/100/1000BASE-TX ports	300 milliseconds	3100 milliseconds
1000BASE-TX ports	300 milliseconds	3100 milliseconds

Table 4-4 Port Debounce Timer Delay Time (continued)

Port Type	Debounce Timer Disabled	Debounce Timer Enabled
Fiber Gigabit Ethernet ports	10 milliseconds	100 milliseconds
10-Gigabit Ethernet ports	10 milliseconds	100 milliseconds

To set the debounce timer on a port, perform this task in privileged mode:

	Task	Command
Step 1	Enable the debounce timer for a port.	**set port debounce** *mod num/port num* {**enable** | **disable**}
Step 2	Verify that the debounce timer of the port is configured correctly.	**show port debounce** [*mod* | *mod_num/port_num*]

This example shows how to enable the debounce timer on port 2/1:

```
Console> (enable) set port debounce 2/1 enable
Debounce is enabled on port 2/1
Warning: Enabling port debounce causes Link Up/Down detections to be delayed.
It results in loss of data traffic during debouncing period, which might
affect the convergence/reconvergence of various Layer 2 and Layer 3 protocols.
Use with caution.
Console> (enable)
```

This example shows how to display the per-port debounce timer settings:

```
Console> (enable) show port debounce
Port   Debounce link timer
-----  ---------------
 2/1   enable
 2/2   disable
Console> (enable)
```

Modifying the Port Debounce Timer Setting

Note Modifying the port debounce timer setting is possible only on fiber Gigabit Ethernet ports.

You can increase the port debounce timer value in increments of 100 up to 5000 milliseconds. You do not need to enable the debounce timer on the port before adjusting the timer value. Specifying any timer value that is greater than the default value in the disabled state enables the debounce timer.

To modify the port debounce timer setting on a port, perform this task in privileged mode:

	Task	Command
Step 1	Modify the port debounce timer setting.	**set port debounce** *mod num/port num* **delay** *time*
Step 2	Verify that the port debounce timer setting has been modified.	**show port debounce** [*mod* | *mod_num/port_num*]

This example shows how to modify the port debounce timer setting on port 2/1:

```
Console> (enable) set port debounce 2/1 delay 500
Debounce time for port 2/1 set to 500 ms.
Warning:Enabling port debounce causes Link Up/Down detections to be delayed.
It results in loss of data traffic during debouncing period, which might
affect the convergence/reconvergence of various Layer 2 and Layer 3 protocols.
Use with caution.
Console> (enable)
```

This example shows how to display the per-port debounce timer setting on port 2/1:

```
Console> (enable) show port debounce 2/1
Port   Debounce link timer
-----  ------------------
 2/1   enabled (500 ms)
Console> (enable)
```

Configuring a Timeout Period for Ports in errdisable State

A port is in errdisable state if it is enabled in NVRAM but is disabled at runtime by any process. For example, if UniDirectional Link Detection (UDLD) detects a unidirectional link, the port shuts down at runtime. However, because the NVRAM configuration for the port is enabled (you have not disabled the port), the port status is shown as errdisable.

If a port goes into errdisable state, it is reenabled automatically after a selected time interval. With the timeout enhancement, you can manually prevent a port from being enabled by setting the errdisable timeout for that port to disable using the **set port errdisable-timeout** *mod/port* **disable** command.

A global timer is maintained for all ports. At every *t* seconds, where *t* is the user-configurable timeout, a process checks to see if any ports are in errdisable state. If there are ports in errdisable state, only those ports that have errdisable timeout set (enabled) are reenabled through SCP messages.

By default, all errdisabled ports are reenabled when the global timer times out.

A port enters errdisable state for the following reasons (these reasons appear as configuration options within the **set errdisable-timeout enable** command):

- ARP inspection
- Broadcast suppression
- BPDU port-guard
- CAM monitor
- Channel misconfiguration
- Crossbar failure
- Duplex mismatch
- Layer 2 protocol tunnel misconfiguration
- Layer 2 protocol tunnel threshold exceeded
- Layer 2 protocol tunnel CDP threshold exceeded
- Layer 2 protocol tunnel STP threshold exceeded
- Layer 2 protocol tunnel VTP threshold exceeded
- Link errors RX threshold exceeded
- Link errors TX threshold exceeded

- UDLD
- Other (reasons other than the above)
- All (apply errdisable timeout for all of the above reasons)

You can enable or disable errdisable timeout for each of the above listed reasons. If you specify "other," all ports that are errdisabled by causes *other* than the reasons listed are enabled for errdisable timeout. If you specify "all," all ports that are errdisabled for any reason are enabled for errdisable timeout.

The errdisable feature is disabled by default. The default interval for enabling a port is 300 seconds. The allowable interval range is 30 to 86400 seconds (30 seconds to 24 hours).

To enable and configure the timeout period for ports in the errdisable state, perform these tasks in privileged mode:

Task	Command
Prevent a port from being reenabled at timeout after it goes into the errdisable state.	**set port errdisable-timeout** *mod/port* **disable**
Enable errdisable timeout for the BPDU guard causes.	**set errdisable-timeout enable bpdu-guard**
Enable errdisable timeout for all causes.	**set errdisable-timeout enable all**
Set the errdisable timeout interval.	**set errdisable-timeout interval** *interval*
Display the errdisable timeout configuration.	**show errdisable-timeout**

This example shows how to prevent port 3/3 from being reenabled at timeout after it goes into the errdisable state:

```
Console> (enable) set port errdisable-timeout 3/3 disable
Successfully disabled errdisable-timeout for port 3/3.
Console> (enable)
```

This example shows how to enable errdisable timeout for BPDU guard causes:

```
Console> (enable) set errdisable-timeout enable bpdu-guard
Successfully enabled errdisable-timeout for bpdu-guard.
Console> (enable)
```

This example shows how to enable errdisable timeout for all causes:

```
Console> (enable) set errdisable-timeout enable all
Successfully enabled errdisable-timeout for all.
Console> (enable)
```

This example shows how to set the errdisable timeout interval to 450 seconds:

```
Console> (enable) set errdisable-timeout interval 450
Successfully set errdisable timeout to 450 seconds.
Console> (enable)
```

This example shows how to display the errdisable timeout configuration:

```
Console> (enable) show errdisable-timeout
ErrDisable Reason                    Timeout Status
------------------------------------ --------------
arp-inspection                       enable
bcast-suppression                    enable
bpdu-guard                           enable
cam-monitor                          enable
channel-misconfig                    enable
crossbar-fallback                    enable
duplex-mismatch                      enable
gl2pt-ingress-loop                   enable
gl2pt-threshold-exceed               enable
gl2pt-cdp-threshold-exceed           enable
gl2pt-stp-threshold-exceed           enable
gl2pt-vtp-threshold-exceed           enable
link-rxcrc                           enable
link-txcrc                           enable
udld                                 enable
other                                enable

Interval: 450 seconds

Port  ErrDisable Reason    Port ErrDisableTimeout  Action on Timeout
----  -------------------  ----------------------  -----------------
Console> (enable)
```

Configuring Automatic Module Shutdown

When you enable automatic module shutdown, you can manage your network connectivity issues. A module that frequently resets itself can disrupt traffic load balancing. By enabling the automatic module shutdown, you can disable a module that continually resets due to any hardware or software problems and limit the number of times that the module resets itself before shutting down completely.

You can also shut down a module manually using the **set module disable** or the **set module power down** commands.

After the module shuts down, you must reenable the module manually.

By default, automatic module shutdown is disabled. When you enable automatic module shutdown, the default is that the module can reset itself three times within two minutes.

You must configure these two parameters before an automatic shutdown can occur:

- Frequency—Allows you to specify the threshold value for an automatic module shutdown. When the number of resets reaches the value that is assigned to this option, the Ethernet module can perform an automatic shutdown.
- Period—Allows you to specify the time period in which the number of resets must occur. The period is measured from one of these conditions:
  - When the switch first comes up
  - When the supervisor engine performs a switchover
  - When the Ethernet module is powered up
  - When the module's autoshut counters are cleared

When the frequency threshold is reached and occurs within the defined period, the Ethernet module automatically shuts down and this sample syslog message is displayed:

```
%SYS-5-MOD_AUTOSHUT: Module 2 shutdown automatically, reset 4 times in last 5
minutes due to inband failure
```

When the frequency threshold is reached and occurs outside the defined period, the module does not automatically shut down and this sample syslog message is displayed:

```
%%SYS-4-MOD_AUTOSHUT_SLOW:Module 1 reset frequency exceeded threshold but over 46
mins. Hence NOT powering down module
```

The run-time variable states for the Ethernet module do not synchronize with the standby supervisor engine. The output of the **show autoshut** command on a standby supervisor engine does not track with the number of resets or the reasons for the resets. If the module is powered down by the **autoshut** command, the output stays the same.

You do not have to enable an automatic module shutdown in order to track the number of resets. You can track resets even if you do not enable an automatic module shutdown.

The runtime counters are cleared only for these conditions:

- When you enter the **clear autoshut** command
- When the switch resets
- At module power up
- At supervisor engine switchover

Note An automatic module shutdown is supported on Ethernet modules only.

To enable and configure an automatic module shutdown, perform one of these tasks in privileged mode:

Task	Command
Enable an automatic module shutdown on a module.	**set module autoshut enable** *mod num*
Disable an automatic module shutdown on a module.	**set module autoshut disable** *mod num*
Set the threshold of the number of times that the module can reset itself.	**set autoshut frequency** *num*
Set the period (in minutes) over which the frequency is valid.	**set autoshut period** *minutes*
Clear the run-time counters on a specific module.	**clear autoshut counters** *mod num*
Reset the autoshut frequency to the default setting.	**clear autoshut frequency**
Reset the autoshut period to the default setting.	**clear autoshut period**
Display the automatic module shutdown configuration and current status information.	**show autoshut**

This example shows how to enable an automatic module shutdown on a module:

```
Console> (enable) set module autoshut enable 2
```

This example shows how to disable an automatic module shutdown on a module:

```
Console> (enable) set module autoshut disable 2
```

This example shows how to set the threshold of the number of times that the module can reset itself:

```
Console> (enable) set autoshut frequency 4
```

This example shows how to set the period (in minutes) over which the frequency is valid:

```
Console> (enable) set autoshut period 3
```

This example shows how to clear the run-time counters on a specific module:

```
Console> (enable) clear autoshut counters 3
Automatic shutdown counters cleared for module 3
Console> (enable)
```

This example shows how to reset the automatic module shutdown frequency to the default setting:

```
Console> (enable) clear autoshut frequency
```

This example shows how to reset the automatic module shutdown period to the default setting:

```
Console> (enable) clear autoshut period
```

This example shows how to display the automatic module shutdown configuration and current status information:

```
Console> (enable) show autoshut
AutoShut Frequency:     3 times
AutoShut Period:        5 minutes

Mod Autoshut Current  Number Reason for last Time of last reset
num status   status   resets reset
--- -------- -------- ------ ---------------- -------------------------
1   NA       ok       -      -                -
2   enabled  shutdown 4       inband failure  Mon Jul 14 2003, 22:55:45
3   disabled ok       0      None             -
4   enabled  ok       1      scp failure      Mon Jul 14 2003, 21:03:17
Console> (enable)
```

Configuring Port Error Detection

Note All ports in an EtherChannel should have the same port error detection settings.

To enable or disable port error detection on a port, perform this task in privileged mode (the default port setting for inerrors, RXCRC, and TXCRC is disabled):

	Task	Command
Step 1	Enable or disable port error detection on a port.	**set port errordetection** *mod/port* {**inerrors** | **rxcrc** | **txcrc**} {**disable** | **enable**}
Step 2	Verify the port configuration.	**show port errordetection** [*mod* | *mod/port*]

This example shows how to enable RXCRC port error detection on port 3/1:

```
Console> (enable) set port errordetection 3/1 rxcrc enable
Port(s)  3/1 set to errordetection rxcrc enable.
Console> (enable) show port errordetection 3/1
Port   Rxcrc    Txcrc
-----  -------- --------
 3/1   enabled  disabled
Console> (enable)
```

Configuring Redundant Flex Links

The redundant flex links provide an alternative solution to the Spanning Tree Protocol (STP) by allowing users to disable STP and still provide link-level redundancy. The redundant flex links provide a link backup capability with rapid switchover redundancy. Flex link redundancy allows you to specify two ports to form the redundant link capability. You configure one port as the active port and the other port as the backup or *peer* port. The active port is in the forwarding state while the backup port is in the blocking state and does not allow traffic to pass through.

When the active port of the flex links experiences a failure, the MAC addresses are flushed and flooded. The backup port of the flex links learns the MAC addresses and restores connectivity. The failover convergence time depends on the number of VLANs and the number of MAC addresses. You cannot enable STP on the flex-link ports but you can run STP on other ports in the switch.

Tip We recommend that you use redundant flex links for configurations that have multiple Layer 2 access switches with common VLANs that are connected to a Layer 2 concentrator switch through two uplink ports.

These sections describe how to configure redundant flex links on the Catalyst 6500 series switches:

- Redundant Flex Link Configuration Guidelines and Restrictions, page 4-17
- Specifying the Active and Backup Ports for the Flex Links, page 4-18
- Displaying the Port Configuration of the Flex Links, page 4-18
- Clearing the Port Configuration of the Flex Links, page 4-18

Redundant Flex Link Configuration Guidelines and Restrictions

This section describes the guidelines and restrictions for configuring redundant flex links:

- The maximum number of flex-link pairs (one active port and one backup port) is 16 per switch.
- Flex-link ports cannot be part of an EtherChannel.
- STP—Flex-link ports do not join STP operations. Flex-link ports do not generate STP bridge protocol data units (BPDUs) and they drop all received BPDUs.
- Switched Port Analyzer (SPAN)—SPAN works with flex-link ports.
- VLAN Trunk Protocol (VTP)—As VTP pruning requires working with STP, it does not work on flex-link ports.
- Internet Group Management Protocol (IGMP)—IGMP works with flex-link ports.
- Dynamic Trunking Protocol (DTP)—DTP can run on a flex-link port.

- Redundant flex links are for simple access topologies (two uplinks from a leaf node). You need to make sure that there is a loop-free path from the wiring closet to the access network. Unlike STP, the flex-link port is not designed to detect loops.
- Deploying STP in the core network while running flex-link redundancy on the edge is an acceptable configuration.
- The flex links converge faster only if the directly connected link fails. Any other failure in the network sees no improvement from the flex-link fast convergence.

Specifying the Active and Backup Ports for the Flex Links

To specify an active port and a backup port (peer) for the flex links, perform this task in privileged mode:

Task	Command
Specify an active port and a backup port (peer) for the flex links.	**set port flexlink** *mod/port* **peer** *mod/port*

This example shows how to specify port 3/48 as the flex-link active port and port 3/47 as the flex-link backup (peer) port:

```
Console> (enable) set port flexlink 3/48 peer 3/47
Flexlink is successfully set on the port 3/48 and 3/47
Console> (enable)
```

Displaying the Port Configuration of the Flex Links

To display information about the flex-link port configuration, perform this task in normal mode:

Task	Command
Display information about the flex-link port configuration.	**show port flexlink** [*mod* | *mod/port*]

This example shows how to display information about all the flex-link ports that are configured on the switch:

```
Console> (enable) show port flexlink
Port   State       Peer port  State
-----  ----------  ---------  ----------
 3/47  linkdown    3/48       active
 3/48  active      3/47       linkdown
Console> (enable)
```

Clearing the Port Configuration of the Flex Links

To clear the port configuration of the flex links, perform this task in privileged mode:

Task	Command
Clear the port configuration of the flex links.	**clear port flexlink** *mod/port* [**peer** *mod/port*]

This example shows how to clear port 3/48 as the flex-link active port and port 3/47 as the flex-link backup (peer) port:

```
Console> (enable) clear port flexlink 3/48 peer 3/47
Port 3/48 and 3/47 flexlink pair cleared
Console> (enable)
```

Configuring Jumbo Frames

These sections describe how to configure jumbo frames:

- Configuring Jumbo Frames on the Supervisor Engine, page 4-19
- Configuring Jumbo Frames on the MSFC2, page 4-20

Configuring Jumbo Frames on the Supervisor Engine

When you enable jumbo frames on a port, the port can switch large (or *jumbo*) frames. This feature is useful in optimizing server-to-server performance. The default maximum transmission unit (MTU) frame size is 1548 bytes for all Ethernet ports. By enabling jumbo frames on a port, the MTU size is increased to 9216 bytes.

To enable jumbo frames on a per-port basis, follow these guidelines:

Note The WS-X6148 and WS-X6548 GE-TX modules do not support jumbo frames.

- Jumbo frames are supported on the following:
  - All Ethernet ports
  - Trunk ports
  - EtherChannels
  - sc0 interface (jumbo frames are passed through the sc0 interface as a nonconfigurable default; no CLI configuration is necessary)
- These switching modules support a maximum ingress frame size of 8092 bytes:
  - WS-6516-GE-TX when operating at 100 Mbps. At 10 Mbps and 1000 Mbps, the module supports the jumbo frame default of 9216 bytes.
  - WS-X6148-RJ-45, WS-X6148-RJ-45V, WS-X6148-RJ21, and WS-X6148-RJ21V.
  - WS-X6248-RJ-45, WS-X6248A-RJ-45, WS-X6248-TEL, and WS-X6248A-TEL.
  - WS-X6348-RJ-45, WS-X6348-RJ45V, WS-X6348-RJ-21, and WX-X6348-RJ21V.

  When jumbo frame support is configured, these modules drop ingress frames that are larger than 8092 bytes.
- The WS-X6548-RJ-21 and WS-X6548-RJ-45 modules use different hardware at the PHY level and support the full jumbo frame default value of 9216 bytes.
- Jumbo frames are supported on all Optical Services Modules (OSMs).
- Jumbo frames are not supported on ATM modules (WS-X6101-OC12-SMF/MMF).

- The Multilayer Switching Feature Card 2 (MSFC2) supports jumbo frame routing with Cisco IOS Release 12.1(2)E and later releases.
- The Multilayer Switching Feature Card (MSFC) and the Multilayer Switch Module (MSM) do not support jumbo frame routing; if jumbo frames are sent to these routers, router performance is significantly degraded.

Note Occasionally, you might see a "Jumbo frames inconsistent state" message for a port or multiple ports after entering the **show port jumbo** command. If you see this message, enter the **set port jumbo** command to reenable the ports.

To enable jumbo frames on an Ethernet port, perform this task in privileged mode:

	Task	Command
Step 1	Enable jumbo frames.	**set port jumbo** *mod*/*port* **enable**
Step 2	Verify the port configuration.	**show port jumbo**

This example shows how to enable jumbo frames on a port and verify the configuration:

```
Console> (enable) set port jumbo 2/1 enable
Jumbo frames enabled on port 2/1
Console> (enable) show port jumbo
Jumbo frames MTU size is 9216 bytes
Jumbo frames enabled on port(s) 2/1
```

To disable jumbo frames on an Ethernet port, perform this task in privileged mode:

	Task	Command
Step 1	Disable jumbo frames.	**set port jumbo** *mod*/*port* **disable**
Step 2	Verify the port configuration.	**show port jumbo**

This example shows how to disable jumbo frames on a port:

```
Console> (enable) set port jumbo 2/1 disable
Jumbo frames disabled on port 2/1
Console> (enable)
```

Configuring Jumbo Frames on the MSFC2

With an MSFC2, you can configure the MTU size on VLAN interfaces to support jumbo frame routing.

Jumbo frames support only a single larger-than-default MTU size on the switch. Configuring a VLAN interface with an MTU size that is greater than the default automatically configures all other VLAN interfaces that have an MTU size that is greater than the default to the newly configured size. The VLAN interfaces that have not been changed from the default are not affected.

To configure the MTU value, perform this task:

	Task	Command
Step 1	Access VLAN interface configuration mode.	Router(config)# **interface vlan** *vlan_ID*
Step 2	Set the MTU size. The valid values are from 64 to 17952 bytes[1].	Router(config-if)# **mtu** *mtu_size*
Step 3	Verify the configuration.	Router# **show interface vlan 111**

1. Set the MTU size no larger than 9216, which is the size that is supported by the supervisor engine.

This example shows how to set the MTU size on a VLAN interface and verify the configuration:

```
Router(config)# interface vlan 111
Router(config-if)# mtu 9216
Router(config-if)# end
Router# show interface vlan 111
.
.
.
MTU 9216 bytes, BW 1000000 Kbit, DLY 10 usec,
.
.
.
Router#
```

Checking Connectivity

Use the **ping** and **traceroute** commands to test connectivity.

To check connectivity out a port, perform this task in privileged mode:

	Task	Command
Step 1	Ping a remote host that is located out the port that you want to test.	**ping** [**-s**] *host* [*packet_size*] [*packet_count*]
Step 2	Trace the hop-by-hop route of packets from the switch to a remote host that is located out the port that you want to test.	**traceroute** *host*
Step 3	If the host is unresponsive, check the IP address and default gateway that are configured on the switch.	**show interface** **show ip route**

This example shows how to ping a remote host and trace the hop-by-hop path of packets through the network using **traceroute**:

```
Console> (enable) ping somehost
somehost is alive
Console> (enable) traceroute somehost
traceroute to somehost.company.com (10.1.2.3), 30 hops max, 40 byte packets
 1 engineering-1.company.com (173.31.192.206) 2 ms 1 ms 1 ms
 2 engineering-2.company.com (173.31.196.204) 2 ms 3 ms 2 ms
 3 gateway_a.company.com (173.16.1.201) 6 ms 3 ms 3 ms
 4 somehost.company.com (10.1.2.3) 3 ms * 2 ms
Console> (enable)
```

CHAPTER 5

Configuring Ethernet VLAN Trunks

This chapter describes how to configure Ethernet VLAN trunks on the Catalyst 6500 series switches.

Note For complete information on configuring VLANs, see Chapter 11, "Configuring VLANs."

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Understanding How VLAN Trunks Work

These sections describe how VLAN trunks work on the Catalyst 6500 series switches:

Trunking Overview

A trunk is a point-to-point link between one or more Ethernet switch ports and another networking device such as a router or a switch. Trunks carry the traffic of multiple VLANs over a single link and allow you to extend VLANs across an entire network.

Two trunking encapsulations are available on all Ethernet ports:

- Inter-Switch Link (ISL)—ISL is a Cisco-proprietary trunking encapsulation
- IEEE 802.1Q—802.1Q is an industry-standard trunking encapsulation

You can configure a trunk on a single Ethernet port or on an EtherChannel bundle. For more information about EtherChannel, see Chapter 6, "Configuring EtherChannel."

Ethernet trunk ports support five different trunking modes (see Table 5-1). In addition, you can specify whether the trunk will use ISL encapsulation, 802.1Q encapsulation, or whether the encapsulation type will be autonegotiated.

For trunking to be autonegotiated, the ports must be in the same VLAN Trunking Protocol (VTP) domain. However, you can use the **on** or **nonegotiate** mode to force a port to become a trunk, even if it is in a different domain. For more information on VTP domains, see Chapter 10, "Configuring VTP."

Trunk negotiation is managed by the Dynamic Trunking Protocol (DTP). DTP supports autonegotiation of both ISL and 802.1Q trunks.

Trunking Modes and Encapsulation Type

Note For a complete list of modules that do not support ISL encapsulation, refer to the *Catalyst 6500 Series Release Notes* at this URL:

http://www.cisco.com/en/US/products/hw/switches/ps708/prod_release_notes_list.html

Table 5-1 lists the trunking modes that are used with the **set trunk** command and describes how they function on Fast Ethernet, Gigabit Ethernet, and 10-Gigabit Ethernet ports.

Table 5-1 Ethernet Trunking Modes

Mode	Function
on	Puts the port into permanent trunking mode and negotiates to convert the link into a trunk link. The port becomes a trunk port even if the neighboring port does not agree to the change.
off	Puts the port into permanent nontrunking mode and negotiates to convert the link into a nontrunk link. The port becomes a nontrunk port even if the neighboring port does not agree to the change.
desirable	Makes the port actively attempt to convert the link to a trunk link. The port becomes a trunk port if the neighboring port is set to **on**, **desirable**, or **auto** mode.
auto	Makes the port willing to convert the link to a trunk link. The port becomes a trunk port if the neighboring port is set to **on** or **desirable** mode. This is the default mode for all Ethernet ports.
nonegotiate	Puts the port into permanent trunking mode but prevents the port from generating DTP frames. You must configure the neighboring port manually as a trunk port to establish a trunk link.

Table 5-2 lists the encapsulation types that are used with the **set trunk** command and describes how they function on Ethernet ports. You can enter the **show port capabilities** command to determine which encapsulation types that a particular port supports.

Table 5-2 Ethernet Trunk Encapsulation Types

Encapsulation	Function
isl	Specifies ISL encapsulation on the trunk link.
dot1q	Specifies 802.1Q encapsulation on the trunk link.
negotiate	Specifies that the port negotiate with the neighboring port to become an ISL (preferred) or 802.1Q trunk, depending on the configuration and capabilities of the neighboring port.

The trunking mode, the trunk encapsulation type, and the hardware capabilities of the two connected ports determine whether a trunk link comes up and the type of trunk the link becomes. Table 5-3 shows the result of the possible trunking configurations.

Table 5-3 Results of Possible Fast Gigabit Ethernet, and 10-Gigabit Ethernet Trunk Configurations

Neighbor Port Trunk Mode and Trunk Encapsulation	Local Port Trunk Mode and Trunk Encapsulation								
	off **isl or dot1q**	**on** **isl**	**desirable** **isl**	**auto** **isl**	**on** **dot1q**	**desirable** **dot1q**	**auto** **dot1q**	**desirable** **negotiate**	**auto** **negotiate**
off **isl or dot1q**	Local: Nontrunk Neighbor: Nontrunk	Local: ISL trunk Neighbor: Nontrunk	Local: Nontrunk Neighbor: Nontrunk	Local: Nontrunk Neighbor: Nontrunk	Local: 1Q trunk Neighbor: Nontrunk	Local: Nontrunk Neighbor: Nontrunk	Local: Nontrunk Neighbor: Nontrunk	Local: Nontrunk Neighbor: Nontrunk	Local: Nontrunk Neighbor: Nontrunk
on **isl**	Local: Nontrunk Neighbor: ISL trunk	Local: ISL trunk Neighbor: ISL trunk	Local: ISL trunk Neighbor: ISL trunk	Local: ISL trunk Neighbor: ISL trunk	Local: 1Q trunk[1] Neighbor: ISL trunk[1]	Local: Nontrunk Neighbor: ISL trunk	Local: Nontrunk Neighbor: ISL trunk	Local: ISL trunk Neighbor: ISL trunk	Local: ISL trunk Neighbor: ISL trunk
desirable **isl**	Local: Nontrunk Neighbor: Nontrunk	Local: ISL trunk Neighbor: ISL trunk	Local: ISL trunk Neighbor: ISL trunk	Local: ISL trunk Neighbor: ISL trunk	Local: 1Q trunk Neighbor: Nontrunk	Local: Nontrunk Neighbor: Nontrunk	Local: Nontrunk Neighbor: Nontrunk	Local: ISL trunk Neighbor: ISL trunk	Local: ISL trunk Neighbor: ISL trunk
auto **isl**	Local: Nontrunk Neighbor: Nontrunk	Local: ISL trunk Neighbor: ISL trunk	Local: ISL trunk Neighbor: ISL trunk	Local: Nontrunk Neighbor: Nontrunk	Local: 1Q trunk Neighbor: Nontrunk	Local: Nontrunk Neighbor: Nontrunk	Local: Nontrunk Neighbor: Nontrunk	Local: ISL trunk Neighbor: ISL trunk	Local: Nontrunk Neighbor: Nontrunk
on **dot1q**	Local: Nontrunk Neighbor: 1Q trunk	Local: ISL trunk[1] Neighbor: 1Q trunk[1]	Local: Nontrunk Neighbor: 1Q trunk	Local: Nontrunk Neighbor: 1Q trunk	Local: 1Q trunk Neighbor: 1Q trunk	Local: 1Q trunk Neighbor: 1Q trunk	Local: 1Q trunk Neighbor: 1Q trunk	Local: 1Q trunk Neighbor: 1Q trunk	Local: 1Q trunk Neighbor: 1Q trunk
desirable **dot1q**	Local: Nontrunk Neighbor: Nontrunk	Local: ISL trunk Neighbor: Nontrunk	Local: Nontrunk Neighbor: Nontrunk	Local: Nontrunk Neighbor: Nontrunk	Local: 1Q trunk Neighbor: 1Q trunk	Local: 1Q trunk Neighbor: 1Q trunk	Local: 1Q trunk Neighbor: 1Q trunk	Local: 1Q trunk Neighbor: 1Q trunk	Local: 1Q trunk Neighbor: 1Q trunk

Table 5-3 Results of Possible Fast Gigabit Ethernet, and 10-Gigabit Ethernet Trunk Configurations (continued)

Neighbor Port Trunk Mode and Trunk Encapsulation	Local Port Trunk Mode and Trunk Encapsulation								
	off isl or dot1q	**on isl**	**desirable isl**	**auto isl**	**on dot1q**	**desirable dot1q**	**auto dot1q**	**desirable negotiate**	**auto negotiate**
auto dot1q	Local: Nontrunk Neighbor: Nontrunk	Local: ISL trunk Neighbor: Nontrunk	Local: Nontrunk Neighbor: Nontrunk	Local: Nontrunk Neighbor: Nontrunk	Local: 1Q trunk Neighbor: 1Q trunk	Local: 1Q trunk Neighbor: 1Q trunk	Local: Nontrunk Neighbor: Nontrunk	Local: 1Q trunk Neighbor: 1Q trunk	Local: Nontrunk Neighbor: Nontrunk
desirable negotiate	Local: Nontrunk Neighbor: Nontrunk	Local: ISL trunk Neighbor: ISL trunk	Local: ISL trunk Neighbor: ISL trunk	Local: ISL trunk Neighbor: ISL trunk	Local: 1Q trunk Neighbor: 1Q trunk	Local: 1Q trunk Neighbor: 1Q trunk	Local: 1Q trunk Neighbor: 1Q trunk	Local: ISL trunk Neighbor: ISL trunk	Local: ISL trunk Neighbor: ISL trunk
auto negotiate	Local: Nontrunk Neighbor: Nontrunk	Local: ISL trunk Neighbor: ISL trunk	Local: ISL trunk Neighbor: ISL trunk	Local: Nontrunk Neighbor: Nontrunk	Local: 1Q trunk Neighbor: 1Q trunk	Local: 1Q trunk Neighbor: 1Q trunk	Local: Nontrunk Neighbor: Nontrunk	Local: ISL trunk Neighbor: ISL trunk	Local: Nontrunk Neighbor: Nontrunk

1. Using this configuration can result in spanning tree loops and is not recommended.

Note DTP is a point-to-point protocol. However, some internetworking devices might forward DTP frames improperly. To avoid this problem, ensure that trunking is turned **off** on ports that are connected to non-switch devices if you do not intend to trunk across those links. When manually enabling trunking on a link to a Cisco router, enter the **nonegotiate** keyword to cause the port to become a trunk but not generate DTP frames.

802.1Q Trunk Configuration Guidelines and Restrictions

The following configuration guidelines and restrictions apply when using 802.1Q trunks and impose some limitations on the trunking strategy for a network:

- When connecting Cisco switches through an 802.1Q trunk, make sure that the native VLAN for an 802.1Q trunk is the same on both ends of the trunk link. If the native VLAN on one end of the trunk is different from the native VLAN on the other end, spanning-tree loops might result.
- Disabling spanning tree on the native VLAN of an 802.1Q trunk without disabling spanning tree on every VLAN in the network can cause spanning-tree loops. We recommend that you leave spanning tree enabled on the native VLAN of an 802.1Q trunk. If this is not possible, disable spanning tree on every VLAN in the network. Make sure that your network is free of physical loops before disabling spanning tree.
- When you connect two Cisco switches through 802.1Q trunks, the switches exchange spanning-tree BPDUs on each VLAN that is allowed on the trunks. The BPDUs on the native VLAN of the trunk are sent untagged to the reserved IEEE 802.1D spanning-tree multicast MAC address (01-80-C2-00-00-00). The BPDUs on all other VLANs on the trunk are sent tagged to the reserved Cisco Shared Spanning Tree (SSTP) multicast MAC address (01-00-0c-cc-cc-cd).

- Non-Cisco 802.1Q switches maintain only a single instance of spanning tree (the Mono Spanning Tree, or MST) that defines the spanning-tree topology for all VLANs. When you connect a Cisco switch to a non-Cisco switch through an 802.1Q trunk, the MST of the non-Cisco switch and the native VLAN spanning tree of the Cisco switch combine to form a single spanning-tree topology known as the Common Spanning Tree (CST).

  When you connect a Cisco switch to a non-Cisco switch, the CST is always on VLAN 1. The Cisco switch sends an untagged IEEE BDPU (01-80-C2-00-00-00) on VLAN 1 for the CST. On the native VLAN, the Cisco switch sends an untagged Cisco BPDU (01-00-0C-CC-CC-CC) which the non-Cisco switch forwards but does not act on (the IEEE BPDU is not forwarded on the native VLAN).
- Because Cisco switches transmit BPDUs to the SSTP multicast MAC address on VLANs other than the native VLAN of the trunk, non-Cisco switches do not recognize these frames as BPDUs and flood them on all ports in the corresponding VLAN. Other Cisco switches that are connected to the non-Cisco 802.1Q cloud receive these flooded BPDUs. This situation allows Cisco switches to maintain a per-VLAN spanning-tree topology across a cloud of non-Cisco 802.1Q switches. The non-Cisco 802.1Q cloud separating the Cisco switches is treated as a single broadcast segment between all switches that are connected to the non-Cisco 802.1Q cloud through the 802.1Q trunks.
- Make sure that the native VLAN is the same on *all* of the 802.1Q trunks connecting the Cisco switches to the non-Cisco 802.1Q cloud.
- If you are connecting multiple Cisco switches to a non-Cisco 802.1Q cloud, all of the connections *must* be through 802.1Q trunks. You *cannot* connect Cisco switches to a non-Cisco 802.1Q cloud through ISL trunks or through access ports because the switch will place the ISL trunk port or access port into the spanning tree "port inconsistent" state and no traffic will pass through the port.

Default Trunk Configuration

Table 5-4 shows the default Ethernet trunk configuration.

Table 5-4 Default Ethernet Trunk Configuration

Feature	Default Configuration
Trunk mode	**auto**
Trunk encapsulation	**negotiate**
Allowed VLAN range	VLANs 1–1005, 1025–4094[1]

1. With software release 8.3(1) and later releases, instead of reserved VLANs, we now have only user and internal VLANs. VLAN manager no longer permanently sets aside VLANs for features that require them; they are now dynamically assigned as needed. The entire VLAN range (1 to 4094) is now available for user (and internal) VLANs.

Configuring a Trunk Link

These sections describe how to configure a trunk link on Ethernet ports and how to define the allowed VLAN range on a trunk:

- Configuring an ISL Trunk, page 5-6
- Configuring an 802.1Q Trunk, page 5-7
- Configuring an ISL/802.1Q Negotiating Trunk Port, page 5-8

Configuring an ISL Trunk

To configure an ISL trunk, perform this task in privileged mode:

	Task	Command
Step 1	Configure an ISL trunk.	**set trunk** *mod/port* [**on** | **off** | **desirable** | **auto** | **nonegotiate**] **isl**
Step 2	Verify the trunking configuration.	**show trunk** [*mod/port*]

This example shows how to configure a port as a trunk and verify the trunk configuration. This example assumes that the neighboring port is in **auto** mode:

```
Console> (enable) set trunk 1/1 on
Port(s) 1/1 trunk mode set to on.
Console> (enable) 06/16/1998,22:16:39:DTP-5:Port 1/1 has become isl trunk
06/16/1998,22:16:40:PAGP-5:Port 1/1 left bridge port 1/1.
06/16/1998,22:16:40:PAGP-5:Port 1/1 joined bridge port 1/1.
Console> (enable) show trunk
Port      Mode         Encapsulation  Status        Native vlan
--------  -----------  -------------  ------------  -----------
 1/1      on           isl            trunking      1
Port      Vlans allowed on trunk
--------  ---------------------------------------------------------------------
 1/1      1-1005, 1025-4094

Port      Vlans allowed and active in management domain
--------  ---------------------------------------------------------------------
 1/1      1,521-524
Port      Vlans in spanning tree forwarding state and not pruned
--------  ---------------------------------------------------------------------
 1/1
Console> (enable)
```

This example shows how to place a port in **desirable** mode and verify the trunk configuration. This example assumes that the neighboring port is in **auto** mode:

```
Console> (enable) set trunk 1/2 desirable
Port(s) 1/2 trunk mode set to desirable.
Console> (enable) 06/16/1998,22:20:16:DTP-5:Port 1/2 has become isl trunk
06/16/1998,22:20:16:PAGP-5:Port 1/2 left bridge port 1/2.
06/16/1998,22:20:16:PAGP-5:Port 1/2 joined bridge port 1/2.
```

```
Console> (enable) show trunk 1/2
Port      Mode         Encapsulation  Status        Native vlan
--------  -----------  -------------  ------------  -----------
 1/2      desirable    isl            trunking      1
Port      Vlans allowed on trunk
--------  ---------------------------------------------------------------------
 1/2      1-1005, 1025-4094
Port      Vlans allowed and active in management domain
--------  ---------------------------------------------------------------------
 1/2      1,521-524
Port      Vlans in spanning tree forwarding state and not pruned
--------  ---------------------------------------------------------------------
 1/2
Console> (enable)
```

Configuring an 802.1Q Trunk

To configure an 802.1Q trunk, perform this task in privileged mode:

	Task	Command
Step 1	Configure an 802.1Q trunk.	**set trunk** *mod/port* [**on** | **off** | **desirable** | **auto** | **nonegotiate**] **dot1q**
Step 2	Verify the trunking configuration.	**show trunk** [*mod/port*]

This example shows how to configure an 802.1Q trunk and verify the trunk configuration:

```
Console> (enable) set trunk 2/9 desirable dot1q
Port(s) 2/9 trunk mode set to desirable.
Port(s) 2/9 trunk type set to dot1q.
Console> (enable) 07/02/1998,18:22:25:DTP-5:Port 2/9 has become dot1q trunk

Console> (enable) show trunk
Port      Mode         Encapsulation  Status        Native vlan
--------  -----------  -------------  ------------  -----------
 2/9      desirable    dot1q          trunking      1

Port      Vlans allowed on trunk
--------  ---------------------------------------------------------------------
 2/9      1-1005, 1025-4094

Port      Vlans allowed and active in management domain
--------  ---------------------------------------------------------------------
 2/9      1,5,10-32,101-120,150,200,250,300,400,500,600,700,800,900,1000

Port      Vlans in spanning tree forwarding state and not pruned
--------  ---------------------------------------------------------------------
 2/9      5,10-32,101-120,150,200,250,300,400,500,600,700,800,900,1000
Console> (enable)
```

Configuring an ISL/802.1Q Negotiating Trunk Port

To configure a trunk port to negotiate the trunk encapsulation type (either ISL or 802.1Q), perform this task in privileged mode:

	Task	Command
Step 1	Configure a port to negotiate the trunk encapsulation type.	**set trunk** *mod/port* [**on** | **off** | **desirable** | **auto** | **nonegotiate**] **negotiate**
Step 2	Verify the trunking configuration.	**show trunk** [*mod/port*]

This example shows how to configure a port to negotiate the encapsulation type and verify the trunk configuration. This example assumes that the neighboring port is in **auto** mode with encapsulation set to **isl** or **negotiate**.

```
Console> (enable) set trunk 4/11 desirable negotiate
Port(s) 4/11 trunk mode set to desirable.
Port(s) 4/11 trunk type set to negotiate.
Console> (enable) show trunk 4/11
Port      Mode         Encapsulation  Status        Native vlan
--------  -----------  -------------  ------------  -----------
 4/11     desirable    n-isl          trunking      1

Port      Vlans allowed on trunk
--------  ---------------------------------------------------------------------
 4/11     1-1005,1025-4094

Port      Vlans allowed and active in management domain
--------  ---------------------------------------------------------------------
 4/11     1,5,10-32,55,101-120,998-1000

Port      Vlans in spanning tree forwarding state and not pruned
--------  ---------------------------------------------------------------------
 4/11     1,5,10-32,55,101-120,998-1000
Console> (enable)
```

Defining the Allowed VLANs on a Trunk

When you configure a trunk port, all VLANs are added to the allowed VLANs list for that trunk. However, you can remove VLANs from the allowed list to prevent traffic for those VLANs from passing over the trunk.

Note When you first configure a port as a trunk, entering the **set trunk** command always adds all VLANs to the allowed VLANs list for the trunk, even if you specify a VLAN range (any specified VLAN range is ignored). To modify the allowed VLANs list, use a combination of the **clear trunk** and **set trunk** commands to specify the allowed VLANs.

In software releases prior to software release 8.3(1), to define the allowed VLANs list for a trunk port, perform this task in privileged mode:

	Task	Command
Step 1	Remove VLANs from the allowed VLANs list for a trunk.	**clear trunk** *mod/port vlans*
Step 2	(Optional) Add specific VLANs to the allowed VLANs list for a trunk.	**set trunk** *mod/port vlans*
Step 3	Verify the allowed VLANs list for the trunk.	**show trunk** [*mod/port*]

This example shows how to define the allowed VLANs list to allow VLANs 1–100, VLANs 500–1005, and VLAN 2500 on trunk port 1/1 and verify the allowed VLAN list for the trunk:

```
Console> (enable) clear trunk 1/1 101-499
Removing Vlan(s) 101-499 from allowed list.
Port 1/1 allowed vlans modified to 1-100,500-1005.
Console> (enable) set trunk 1/1 2500
Adding vlans 2500 to allowed list.
Port(s) 1/1 allowed vlans modified to 1-100,500-1005,2500.
Console> (enable) show trunk 1/1
Port      Mode         Encapsulation  Status        Native vlan
--------  -----------  -------------  ------------  -----------
 1/1      desirable    isl            trunking      1
Port      Vlans allowed on trunk
--------  ---------------------------------------------------------------------
 1/1      1-100, 500-1005,2500
Port      Vlans allowed and active in management domain
--------  ---------------------------------------------------------------------
 1/1      1,521-524
Port      Vlans in spanning tree forwarding state and not pruned
--------  ---------------------------------------------------------------------
 1/1      1,521-524
Console> (enable)
```

In software release 8.3(1) and later releases, if you want to configure a trunk but do not want to allow any VLANs on the trunk, enter the **none** keyword as follows:

```
Console> (enable) set trunk 7/1 on none dot1q
Removing Vlan(s) 1-4094 from allowed list.
Port 7/1 allowed vlans modified to none.
Port(s) 7/1 trunk mode set to on.
Port(s) 7/1 trunk type set to dot1q.
Console> (enable)
```

Disabling a Trunk Port

To turn off trunking on a port, perform this task in privileged mode:

	Task	Command
Step 1	Turn off trunking on a port.	**set trunk** *mod/port* **off**
Step 2	Verify the trunking configuration.	**show trunk** [*mod/port*]

To return a port to the default trunk type and mode for that port type, perform this task in privileged mode:

	Task	Command
Step 1	Return the port to the default trunking type and mode for that port type.	**clear trunk** *mod/port*
Step 2	Verify the trunking configuration.	**show trunk** [*mod/port*]

Disabling VLAN 1 on Trunks

On the Catalyst 6500 series switches, VLAN 1 is enabled by default to allow control protocols to transmit and receive packets across the network topology. However, when VLAN 1 is enabled on trunk links in a large complex network, the impact of broadcast storms increases. Because spanning tree applies to the entire network, spanning-tree loops might increase when you enable VLAN 1 on all trunk links. To prevent this scenario, you can disable VLAN 1 on trunk interfaces.

When you disable VLAN 1 on a trunk interface, no user traffic is transmitted and received across that trunk interface, but the supervisor engine continues to transmit and receive packets from control protocols such as Cisco Discovery Protocol (CDP), VTP, Port Aggregation Protocol (PAgP), and DTP.

When a trunk port with VLAN 1 disabled becomes a nontrunk port, it is added to the native VLAN. If the native VLAN is VLAN 1, the port is enabled and added to VLAN 1.

To disable VLAN 1 on a trunk interface, perform this task in privileged mode:

	Task	Command
Step 1	Disable VLAN 1 on the trunk interface.	**clear trunk** *mod/port* [*vlan-range*]
Step 2	Verify the allowed VLAN list for the trunk.	**show trunk** [*mod/port*]

This example shows how to disable VLAN 1 on a trunk link and verify the configuration:

```
Console> (enable) clear trunk 8/1 1
Removing Vlan(s) 1 from allowed list.
Port  8/1 allowed vlans modified to 2-1005.
Console> (enable) show trunk 8/1
Port      Mode         Encapsulation  Status        Native vlan
--------  -----------  -------------  ------------  -----------
 8/1      on           isl            trunking      1

Port      Vlans allowed on trunk
--------  ---------------------------------------------------------------------
 8/1      2-1005, 1025-4094

Port      Vlans allowed and active in management domain
--------  ---------------------------------------------------------------------
 8/1      2-6,10,20,50,100,152,200,300,400,500,521,524,570,776,801-802,850,917,9
99,1003,1005

Port      Vlans in spanning tree forwarding state and not pruned
--------  ---------------------------------------------------------------------
 8/1      2-6,10,20,50,100,152,200,300,400,500,521,524,570,776,802,850,917,999,1
003,1005
Console> (enable) show config
```

Enabling 802.1Q Tagging of Native VLAN Traffic

The **set dot1q-all-tagged enable** command is a global command that configures a switch to forward all frames from 802.1Q trunks with 802.1Q tagging in the native VLAN, and admit only 802.1Q tagged frames on 802.1Q trunks, dropping any untagged traffic, including untagged traffic in the native VLAN. You can enter this command on any switch that needs to support 802.1Q tunneling with 802.1Q trunks.

To configure the switch to forward all 802.1Q tagged frames on 802.1Q trunks, perform this task in privileged mode:

	Task	Command
Step 1	Enable the switch to forward all 802.1Q tagged frames.	**set dot1q-all-tagged** [**enable** | **disable**]
Step 2	Verify the configuration.	**show dot1q-all-tagged**

This example shows how to enable the switch to forward all 802.1Q traffic and verify the configuration:

```
Console> (enable) set dot1q-all-tagged enable
Dot1q-all-tagged feature enabled globally.
Console> (enable) show dot1q-all-tagged
Dot1q-all-tagged feature globally enabled.
Console> (enable)
```

Disabling 802.1Q Tagging on Specific Ports

The **set port dot1q-all-tagged** *mod/port* **enable | disable** command allows you to disable 802.1Q tagging on specific ports. Enter the **set port dot1q-all-tagged disable** command to selectively disable 802.1Q tagging on ports that connect to the devices that do not support 802.1Q tagged traffic. If you enable or disable 802.1Q tagging on an EtherChannel port, the configuration is applied to all ports in the channel.

Note If you did not enter the global **set dot1q-all-tagged enable** command, the default group is never tagged and the per-port setting has no effect.

If you entered the global **set dot1q-all-tagged enable** command, the per-port setting controls whether frames are tagged.

Note The **set port dot1q-all-tagged** *mod/port* **enable | disable** command is not supported on the ports on the MSFC or ports on the WS-X6101 OC-12 ATM modules.

To disable the forwarding of 802.1Q tagged frames on specific ports, perform this task in privileged mode:

	Task	Command
Step 1	Enable or disable the forwarding of 802.1Q tagged frames on specific ports or on all ports.	**set port dot1q-all-tagged** *mod/port* **enable** | **disable**
Step 2	Verify the configuration.	**show port dot1q-all-tagged**

This example shows how to disable the forwarding of 802.1Q tagged frames on port 3/2 and verify the configuration:

```
Console> (enable) set port dot1q-all-tagged 3/2 disable
Packets on native vlan will not be tagged on port  3/2.
Console> (enable) show port dot1q-all-tagged

Dot1q-all-tagged feature globally enabled.
Port    Dot1q-all-tagged mode
----    ---------------------
 2/1    enable
 2/2    enable
 3/1    enable
 3/2    disable
 3/3    enable
 3/4    enable
 3/5    enable
<output truncated>
```

Specifying a Custom 802.1Q EtherType Field

Note A custom 802.1Q EtherType field is supported only on the following modules: Supervisor Engine 2, Supervisor Engine 720, and Supervisor Engine 32 uplink ports, WS-X6516-GBIC, WS-X6516A-GBIC, WS-X6516-GE-TX, WS-X6148-GE-TX, WS-X6148V-GE-TX, WS-X6548-GE-TX, WS-X6548V-GE-TX, WS-X6748-GE-TX, WS-X6724-SFP, WS-X6704-10GE, and WS-X6501-10GEX4.

Note A custom 802.1Q EtherType field is not supported on EtherChannels. If you configure a port with a custom 802.1Q EtherType field, the port cannot join a channel. If a channel is already configured, you cannot change the 802.1Q EtherType on any of the channel ports.

Note On the WS-X6516A-GBIC, WS-X6516-GBIC, and WS-X6548-GE-TX modules, if you configure a port with a custom 802.1Q EtherType in the port groups 1 through 8 or 9 through 16, all the ports in the group are configured with the custom 802.1Q EtherType. On the WS-X6516-GE-TX module, if you configure a port with a custom 802.1Q EtherType in the port groups 1 through 4, 5 through 8, 9 through 12, or 13 through 16, all the ports in the group are configured with the custom 802.1Q EtherType.

Note You can use a custom 802.1Q EtherType field on trunk ports, 802.1Q access ports, and 802.1Q/802.1p multi-VLAN access ports. Additionally, you should configure the custom EtherType value the same on both ends of a link.

By specifying a custom EtherType field, your network can support Cisco and non-Cisco switches that do not use the standard 0x8100 EtherType to identify 802.1Q-tagged frames. When you specify a custom EtherType field, you can identify 802.1Q tagged frames and switch the frames to a specified VLAN. The two bytes immediately following the EtherType are interpreted as a standard 802.1Q tag. Specify the value of the two-byte EtherType field in hexadecimal. The **default** value is 8100.

To specify a custom 802.1Q EtherType value in the 802.1Q tag, perform this task in privileged mode:

	Task	Command
Step 1	Specify a custom EtherType field for a port.	**set port dot1q-ethertype** *mod/port* {*value* | **default**}
Step 2	Verify the configuration.	**show port dot1q-ethertype** [*mod* | *mod/port*]

This example shows how to set the 802.1Q EtherType to 0x1234 on port 2/1 and verify the configuration:

```
Console> (enable) set port dot1q-ethertype 2/1 1234
All the group ports 2/1-2 associated with port 2/1 will be modified.
Do you want to continue (y/n) [n]?y
Dot1q Ethertype value set to 0x1234 on ports 2/1-2.
Console> (enable)

Console> (enable) show port dot1q-ethertype 2/1
Port    Dot1q ethertype value
----    ---------------------
 2/1    1234
Console> (enable)
```

Returning a Custom 802.1Q EtherType Field to the Standard EtherType

The **set port dot1q-ethertype** *mod/port* {*value* | **default**} command is the only command that is required to return the custom 802.1Q EtherType field to the standard EtherType field (0x8100).

To return the custom EtherType field to the default value (0x8100), perform this task in privileged mode:

	Task	Command
Step 1	Return the EtherType field to the standard value (0x8100) for a port.	**set port dot1q-ethertype** *mod/port* **default**
Step 2	Verify the configuration.	**show port dot1q-ethertype** [*mod* | *mod/port*]

This example shows how to return the 802.1Q EtherType field to the standard EtherType field (0x8100) on port 2/1 and verify the configuration:

```
Console> (enable) set port dot1q-ethertype 2/1 default
All the group ports 2/1-2 associated with port 2/1 will be modified.
Do you want to continue (y/n) [n]?y
Dot1q Ethertype value set to 0x8100 on ports 2/1-2.
Console> (enable)

Console> (enable) show port dot1q-ethertype 2/1
Port    Dot1q ethertype value
----    ---------------------
 2/1    8100
Console> (enable)
```

Example VLAN Trunk Configurations

This section contains example VLAN trunk configurations:

ISL Trunk Configuration Example

This example shows how to configure an ISL trunk between two switches and limit the allowed VLANs on the trunk to VLAN 1 and VLANs 520–530.

In this example, port 1/1 on Switch 1 is connected to a Fast Ethernet port on another switch. Both ports are in their default state, with the trunk mode set to **auto** (for more information, see the "Default Trunk Configuration" section on page 5-5).

To configure an ISL trunk between two switches and limit the allowed VLANs on the trunk to VLAN 1 and VLANs 520–530, perform these steps:

Step 1 Configure port 1/1 on Switch 1 as an ISL trunk port by entering the **set trunk** command. By specifying the **desirable** keyword, the trunk is automatically negotiated with the neighboring port (port 1/2 on Switch 2). ISL encapsulation is assumed based on the hardware type.

```
Switch1> (enable) set trunk 1/1 desirable
Port(s) 1/1 trunk mode set to desirable.
Switch1> (enable) 06/18/1998,12:20:23:DTP-5:Port 1/1 has become isl trunk
06/18/1998,12:20:23:PAGP-5:Port 1/1 left bridge port 1/1.
06/18/1998,12:20:23:PAGP-5:Port 1/1 joined bridge port 1/1.
Switch1> (enable)
```

Step 2 Check the configuration by entering the **show trunk** command. The Status field in the screen output indicates that port 1/1 is trunking.

```
Switch1> (enable) show trunk 1/1
Port      Mode         Encapsulation  Status        Native vlan
--------  -----------  -------------  ------------  -----------
 1/1      desirable    isl            trunking      1
Port      Vlans allowed on trunk
--------  ---------------------------------------------------------------------
 1/1      1-1005, 1025-4094
Port      Vlans allowed and active in management domain
--------  ---------------------------------------------------------------------
 1/1      1,521-524
Port      Vlans in spanning tree forwarding state and not pruned
--------  ---------------------------------------------------------------------
 1/1
Switch1> (enable)
```

Step 3 Define the allowed VLAN list for the trunk by entering the **clear trunk** command to remove the VLANs that should not pass traffic over the trunk link.

```
Switch1> (enable) clear trunk 1/1 2-519
Removing Vlan(s) 2-519 from allowed list.
Port 1/1 allowed vlans modified to 1,520-1005.
Switch1> (enable) clear trunk 1/1 531-1005
Removing Vlan(s) 531-1005 from allowed list.
Port 1/1 allowed vlans modified to 1,520-530.
Switch1> (enable) show trunk 1/1
Port      Mode         Encapsulation  Status        Native vlan
--------  -----------  -------------  ------------  -----------
 1/1      desirable    isl            trunking      1
Port      Vlans allowed on trunk
--------  ---------------------------------------------------------------------
 1/1      1,520-530
Port      Vlans allowed and active in management domain
--------  ---------------------------------------------------------------------
 1/1      1,521-524
Port      Vlans in spanning tree forwarding state and not pruned
--------  ---------------------------------------------------------------------
 1/1      1,521-524
Switch1> (enable)
```

Step 4 Verify connectivity across the trunk by entering the **ping** command.

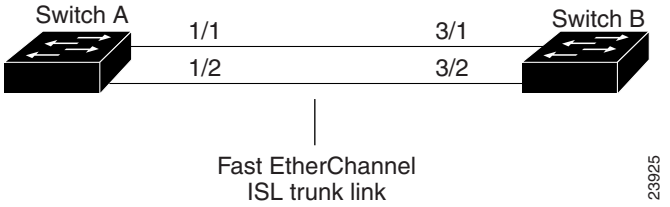
```
Switch1> (enable) ping switch2
switch2 is alive
Switch1> (enable)
```

ISL Trunk Over EtherChannel Link Example

This example shows how to configure an ISL trunk over an EtherChannel link between two switches.

Figure 5-1 shows two switches that are connected through two 100BASE-TX Fast Ethernet ports.

Figure 5-1 ISL Trunk Over Fast EtherChannel Link



To configure the switches to form a two-port EtherChannel bundle and then configure the EtherChannel bundle as an ISL trunk link, perform these steps:

Step 1 Confirm the channeling and trunking status of the switches by entering the **show port channel** and **show trunk** commands.

```
Switch_A> (enable) show port channel
No ports channelling
Switch_A> (enable) show trunk
No ports trunking.
Switch_A> (enable)

Switch_B> (enable) show port channel
No ports channelling
Switch_B> (enable) show trunk
No ports trunking.
Switch_B> (enable)
```

Step 2 Configure the ports on Switch A to negotiate an EtherChannel bundle with the neighboring switch by entering the **set port channel** command. This example assumes that the neighboring ports on Switch B are in EtherChannel **auto** mode. The system logging messages provide information about the formation of the EtherChannel bundle.

```
Switch_A> (enable) set port channel 1/1-2 desirable
Port(s) 1/1-2 channel mode set to desirable.
Switch_A> (enable) %PAGP-5-PORTFROMSTP:Port 1/1 left bridge port 1/1
%PAGP-5-PORTFROMSTP:Port 1/2 left bridge port 1/2
%PAGP-5-PORTFROMSTP:Port 1/2 left bridge port 1/2
%PAGP-5-PORTTOSTP:Port 1/1 joined bridge port 1/1-2
%PAGP-5-PORTTOSTP:Port 1/2 joined bridge port 1/1-2

Switch_B> (enable) %PAGP-5-PORTFROMSTP:Port 3/1 left bridge port 3/1
%PAGP-5-PORTFROMSTP:Port 3/2 left bridge port 3/2
%PAGP-5-PORTFROMSTP:Port 3/2 left bridge port 3/2
%PAGP-5-PORTTOSTP:Port 3/1 joined bridge port 3/1-2
%PAGP-5-PORTTOSTP:Port 3/2 joined bridge port 3/1-2
```

Step 3 After the EtherChannel bundle is negotiated, verify the configuration by entering the **show port channel** command.

```
Switch_A> (enable) show port channel
Port  Status     Channel   Channel     Neighbor                  Neighbor
                 mode      status      device                    port
----- ---------- --------- ----------- ------------------------- ----------
 1/1  connected  desirable channel     WS-C5000    009979082(Sw  3/1
 1/2  connected  desirable channel     WS-C5000    009979082(Sw  3/2
----- ---------- --------- ----------- ------------------------- ----------
Switch_A> (enable)

Switch_B> (enable) show port channel
Port  Status     Channel   Channel     Neighbor                  Neighbor
```

```
                 mode      status      device                    port
----- ---------- --------- ----------- ------------------------- ----------
 3/1  connected  auto      channel     WS-C5500    069003103(Sw  1/1
 3/2  connected  auto      channel     WS-C5500    069003103(Sw  1/2
----- ---------- --------- ----------- ------------------------- ----------
Switch_B> (enable)
```

Step 4 Configure one of the ports in the EtherChannel bundle to negotiate an ISL trunk by entering the **set trunk** command.

The configuration is applied to all of the ports in the bundle. This example assumes that the neighboring ports on Switch B are configured to use **isl** or **negotiate** encapsulation and are in **auto** trunk mode. The system logging messages provide information about the formation of the ISL trunk.

```
Switch_A> (enable) set trunk 1/1 desirable isl
Port(s) 1/1-2 trunk mode set to desirable.
Port(s) 1/1-2 trunk type set to isl.
Switch_A> (enable) %DTP-5-TRUNKPORTON:Port 1/1 has become isl trunk
%DTP-5-TRUNKPORTON:Port 1/2 has become isl trunk
%PAGP-5-PORTFROMSTP:Port 1/1 left bridge port 1/1-2
%PAGP-5-PORTFROMSTP:Port 1/2 left bridge port 1/1-2
%PAGP-5-PORTTOSTP:Port 1/1 joined bridge port 1/1-2
%PAGP-5-PORTTOSTP:Port 1/2 joined bridge port 1/1-2

Switch_B> (enable) %DTP-5-TRUNKPORTON:Port 3/1 has become isl trunk
%DTP-5-TRUNKPORTON:Port 3/2 has become isl trunk
%PAGP-5-PORTFROMSTP:Port 3/1 left bridge port 3/1-2
%PAGP-5-PORTFROMSTP:Port 3/2 left bridge port 3/1-2
%PAGP-5-PORTTOSTP:Port 3/1 joined bridge port 3/1-2
%PAGP-5-PORTTOSTP:Port 3/2 joined bridge port 3/1-2
```

Step 5 After the ISL trunk link is negotiated, verify the configuration by entering the **show trunk** command.

```
Switch_A> (enable) show trunk
Port      Mode         Encapsulation  Status        Native vlan
--------  -----------  -------------  ------------  -----------
 1/1      desirable    isl            trunking      1
 1/2      desirable    isl            trunking      1

Port      Vlans allowed on trunk
--------  ---------------------------------------------------------------------
 1/1      1-1005, 1025-4094
 1/2      1-1005, 1025-4094

Port      Vlans allowed and active in management domain
--------  ---------------------------------------------------------------------
 1/1      1-5,10,20,50,152,200,300,400,500,521-524,570,850,917,999
 1/2      1-5,10,20,50,152,200,300,400,500,521-524,570,850,917,999

Port      Vlans in spanning tree forwarding state and not pruned
--------  ---------------------------------------------------------------------
 1/1      1-5,10,20,50,152,200,300,400,500,521-524,570,850,917,999
 1/2      1-5,10,20,50,152,200,300,400,500,521-524,570,850,917,999
Switch_A> (enable)

Switch_B> (enable) show trunk
Port      Mode         Encapsulation  Status        Native vlan
--------  -----------  -------------  ------------  -----------
 3/1      auto         isl            trunking      1
 3/2      auto         isl            trunking      1

Port      Vlans allowed on trunk
--------  ---------------------------------------------------------------------
 3/1      1-1005, 1025-4094
```

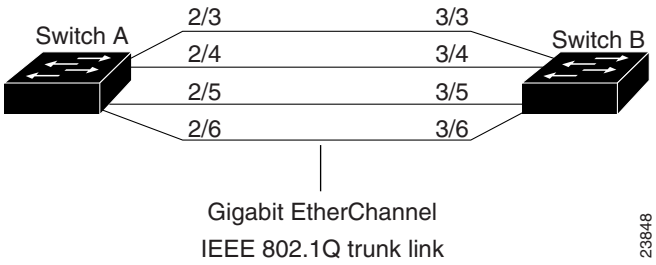
```
 3/2      1-1005, 1025-4094

Port      Vlans allowed and active in management domain
--------  ---------------------------------------------------------------------
 3/1      1-5,10,20,50,152,200,300,400,500,521-524,570,850,917,999
 3/2      1-5,10,20,50,152,200,300,400,500,521-524,570,850,917,999
Port      Vlans in spanning tree forwarding state and not pruned
--------  ---------------------------------------------------------------------
 3/1      1-5,10,20,50,152,200,300,400,500,521-524,570,801,850,917,999
 3/2      1-5,10,20,50,152,200,300,400,500,521-524,570,801,850,917,999
Switch_B> (enable)
```

802.1Q Trunk Over EtherChannel Link Example

This example shows how to configure an 802.1Q trunk over an EtherChannel link between two switches.

Figure 5-2 shows two switches that are connected through four 1000BASE-SX Gigabit Ethernet ports.

Figure 5-2 802.1Q Trunk Over EtherChannel Link



To configure the switches to form a four-port EtherChannel bundle and then configure the EtherChannel bundle as an 802.1Q trunk link, perform these steps:

Step 1 Make sure that all ports on both Switch A and Switch B are assigned to the same VLAN by entering the **set vlan** command. This VLAN is used as the 802.1Q native VLAN for the trunk. In this example, all ports are configured as members of VLAN 1.

```
Switch_A> (enable) set vlan 1 2/3-6
VLAN  Mod/Ports
---- ----------------------
1     2/1-6

Switch_A> (enable)

Switch_B> (enable) set vlan 1 3/3-6
VLAN  Mod/Ports
---- ----------------------
1     3/1-6

Switch_B> (enable)
```

Step 2 Confirm the channeling and trunking status of the switches by entering the **show port channel** and **show trunk** commands.

```
Switch_A> (enable) show port channel
No ports channelling
```

```
Switch_A> (enable) show trunk
No ports trunking.
Switch_A> (enable)

Switch_B> (enable) show port channel
No ports channelling
Switch_B> (enable) show trunk
No ports trunking.
Switch_B> (enable)
```

Step 3 Configure the ports on Switch A to negotiate an EtherChannel bundle with the neighboring switch by entering the **set port channel** command. This example assumes that the neighboring ports on Switch B are in EtherChannel **auto** mode. The system logging messages provide information about the formation of the EtherChannel bundle.

```
Switch_A> (enable) set port channel 2/3-6 desirable
Port(s) 2/3-6 channel mode set to desirable.
Switch_A> (enable) %PAGP-5-PORTFROMSTP:Port 2/3 left bridge port 2/3
%PAGP-5-PORTFROMSTP:Port 2/4 left bridge port 2/4
%PAGP-5-PORTFROMSTP:Port 2/5 left bridge port 2/5
%PAGP-5-PORTFROMSTP:Port 2/6 left bridge port 2/6
%PAGP-5-PORTFROMSTP:Port 2/4 left bridge port 2/4
%PAGP-5-PORTFROMSTP:Port 2/5 left bridge port 2/5
%PAGP-5-PORTFROMSTP:Port 2/6 left bridge port 2/6
%PAGP-5-PORTFROMSTP:Port 2/3 left bridge port 2/3
%PAGP-5-PORTTOSTP:Port 2/3 joined bridge port 2/3-6
%PAGP-5-PORTTOSTP:Port 2/4 joined bridge port 2/3-6
%PAGP-5-PORTTOSTP:Port 2/5 joined bridge port 2/3-6
%PAGP-5-PORTTOSTP:Port 2/6 joined bridge port 2/3-6

Switch_B> (enable) %PAGP-5-PORTFROMSTP:Port 3/3 left bridge port 3/3
%PAGP-5-PORTFROMSTP:Port 3/4 left bridge port 3/4
%PAGP-5-PORTFROMSTP:Port 3/5 left bridge port 3/5
%PAGP-5-PORTFROMSTP:Port 3/6 left bridge port 3/6
%PAGP-5-PORTFROMSTP:Port 3/4 left bridge port 3/4
%PAGP-5-PORTFROMSTP:Port 3/5 left bridge port 3/5
%PAGP-5-PORTFROMSTP:Port 3/6 left bridge port 3/6
%PAGP-5-PORTFROMSTP:Port 3/3 left bridge port 3/3
%PAGP-5-PORTTOSTP:Port 3/3 joined bridge port 3/3-6
%PAGP-5-PORTTOSTP:Port 3/4 joined bridge port 3/3-6
%PAGP-5-PORTTOSTP:Port 3/5 joined bridge port 3/3-6
%PAGP-5-PORTTOSTP:Port 3/6 joined bridge port 3/3-6
```

Step 4 After the EtherChannel bundle is negotiated, verify the configuration by entering the **show port channel** command.

```
Switch_A> (enable) show port channel
Port  Status     Channel   Channel     Neighbor                  Neighbor
                 mode      status      device                    port
----- ---------- --------- ----------- ------------------------- ----------
 2/3  connected  desirable channel     WS-C4003    JAB023806(Sw  2/3
 2/4  connected  desirable channel     WS-C4003    JAB023806(Sw  2/4
 2/5  connected  desirable channel     WS-C4003    JAB023806(Sw  2/5
 2/6  connected  desirable channel     WS-C4003    JAB023806(Sw  2/6
----- ---------- --------- ----------- ------------------------- ----------
Switch_A> (enable)

Switch_B> (enable) show port channel
Port  Status     Channel   Channel     Neighbor                  Neighbor
                 mode      status      device                    port
----- ---------- --------- ----------- ------------------------- ----------
 3/3  connected  auto      channel     WS-C4003    JAB023806(Sw  2/3
 3/4  connected  auto      channel     WS-C4003    JAB023806(Sw  2/4
```

```
 3/5  connected  auto      channel     WS-C4003    JAB023806(Sw  2/5
 3/6  connected  auto      channel     WS-C4003    JAB023806(Sw  2/6
----- ---------- --------- ----------- ------------------------- ----------
Switch_B> (enable)
```

Step 5 Configure one of the ports in the EtherChannel bundle to negotiate an 802.1Q trunk by entering the **set trunk** command. The configuration is applied to all of the ports in the bundle. This example assumes that the neighboring ports on Switch B are configured to use **dot1q** or **negotiate** encapsulation and are in **auto** trunk mode. The system logging messages provide information about the formation of the 802.1Q trunk.

```
Switch_A> (enable) set trunk 2/3 desirable dot1q
Port(s) 2/3-6 trunk mode set to desirable.
Port(s) 2/3-6 trunk type set to dot1q.
Switch_A> (enable) %DTP-5-TRUNKPORTON:Port 2/3 has become dot1q trunk
%DTP-5-TRUNKPORTON:Port 2/4 has become dot1q trunk
%PAGP-5-PORTFROMSTP:Port 2/3 left bridge port 2/3-6
%DTP-5-TRUNKPORTON:Port 2/5 has become dot1q trunk
%PAGP-5-PORTFROMSTP:Port 2/4 left bridge port 2/3-6
%PAGP-5-PORTFROMSTP:Port 2/5 left bridge port 2/3-6
%DTP-5-TRUNKPORTON:Port 2/6 has become dot1q trunk
%PAGP-5-PORTFROMSTP:Port 2/6 left bridge port 2/3-6
%PAGP-5-PORTFROMSTP:Port 2/3 left bridge port 2/3
%PAGP-5-PORTTOSTP:Port 2/3 joined bridge port 2/3-6
%PAGP-5-PORTTOSTP:Port 2/4 joined bridge port 2/3-6
%PAGP-5-PORTTOSTP:Port 2/5 joined bridge port 2/3-6
%PAGP-5-PORTTOSTP:Port 2/6 joined bridge port 2/3-6
Switch_B> (enable) %DTP-5-TRUNKPORTON:Port 3/3 has become dot1q trunk
%DTP-5-TRUNKPORTON:Port 3/4 has become dot1q trunk
%PAGP-5-PORTFROMSTP:Port 3/3 left bridge port 3/3-6
%PAGP-5-PORTFROMSTP:Port 3/4 left bridge port 3/3-6
%PAGP-5-PORTFROMSTP:Port 3/5 left bridge port 3/3-6
%PAGP-5-PORTFROMSTP:Port 3/6 left bridge port 3/3-6
%DTP-5-TRUNKPORTON:Port 3/5 has become dot1q trunk
%DTP-5-TRUNKPORTON:Port 3/6 has become dot1q trunk
%PAGP-5-PORTFROMSTP:Port 3/5 left bridge port 3/3-6
%PAGP-5-PORTFROMSTP:Port 3/6 left bridge port 3/3-6
%PAGP-5-PORTTOSTP:Port 3/3 joined bridge port 3/3-6
%PAGP-5-PORTTOSTP:Port 3/4 joined bridge port 3/3-6
%PAGP-5-PORTTOSTP:Port 3/5 joined bridge port 3/3-6
%PAGP-5-PORTTOSTP:Port 3/6 joined bridge port 3/3-6
```

Step 6 After the 802.1Q trunk link is negotiated, verify the configuration by entering the **show trunk** command.

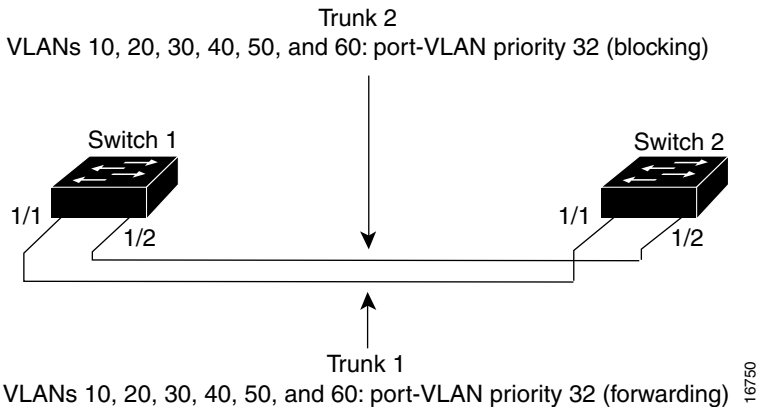
```
Switch_A> (enable) show trunk
Port      Mode         Encapsulation  Status        Native vlan
--------  -----------  -------------  ------------  -----------
 2/3      desirable    dot1q          trunking      1
 2/4      desirable    dot1q          trunking      1
 2/5      desirable    dot1q          trunking      1
 2/6      desirable    dot1q          trunking      1

Port      Vlans allowed on trunk
--------  ---------------------------------------------------------------------
 2/3      1-1005, 1025-4094
 2/4      1-1005, 1025-4094
 2/5      1-1005, 1025-4094
 2/6      1-1005, 1025-4094

Port      Vlans allowed and active in management domain
--------  ---------------------------------------------------------------------
 2/3      1-5,10,20,50,152,200,300,400,500,521-524,570,850,917,999
 2/4      1-5,10,20,50,152,200,300,400,500,521-524,570,850,917,999
 2/5      1-5,10,20,50,152,200,300,400,500,521-524,570,850,917,999
 2/6      1-5,10,20,50,152,200,300,400,500,521-524,570,850,917,999

Port      Vlans in spanning tree forwarding state and not pruned
--------  ---------------------------------------------------------------------
 2/3
 2/4
 2/5
 2/6
Switch_A> (enable)

Switch_B> (enable) show trunk
Port      Mode         Encapsulation  Status        Native vlan
--------  -----------  -------------  ------------  -----------
 3/3      auto         dot1q          trunking      1
 3/4      auto         dot1q          trunking      1
 3/5      auto         dot1q          trunking      1
 3/6      auto         dot1q          trunking      1

Port      Vlans allowed on trunk
--------  ---------------------------------------------------------------------
 3/3      1-1005, 1025-4094
 3/4      1-1005, 1025-4094
 3/5      1-1005, 1025-4094
 3/6      1-1005, 1025-4094

Port      Vlans allowed and active in management domain
--------  ---------------------------------------------------------------------
 3/3      1-5,10,20,50,152,200,300,400,500,521-524,570,850,917,999
 3/4      1-5,10,20,50,152,200,300,400,500,521-524,570,850,917,999
 3/5      1-5,10,20,50,152,200,300,400,500,521-524,570,850,917,999
 3/6      1-5,10,20,50,152,200,300,400,500,521-524,570,850,917,999

Port      Vlans in spanning tree forwarding state and not pruned
--------  ---------------------------------------------------------------------
 3/3      1-5,10,20,50,152,200,300,400,500,521-524,570,850,917,999
 3/4      1-5,10,20,50,152,200,300,400,500,521-524,570,850,917,999
 3/5      1-5,10,20,50,152,200,300,400,500,521-524,570,850,917,999
 3/6      1-5,10,20,50,152,200,300,400,500,521-524,570,850,917,999
Switch_B> (enable)
```

Load-Sharing VLAN Traffic Over Parallel Trunks Example

Using spanning-tree port-VLAN priorities, you can load-share VLAN traffic over parallel trunk ports so that traffic from some VLANs travels over one trunk, while traffic from other VLANs travels over the other trunk. This configuration allows traffic to be carried over both trunks simultaneously (instead of keeping one trunk in blocking mode), which reduces the total traffic that is carried over each trunk while still maintaining a fault-tolerant configuration.

Figure 5-3 shows a parallel trunk configuration between two switches using the Fast Ethernet uplink ports on the supervisor engine.

Figure 5-3 Parallel Trunk Configuration Before Configuring VLAN-Traffic Load Sharing



By default, the port-VLAN priority for both trunks is equal (a value of 32). STP blocks port 1/2 (Trunk 2) for each VLAN on Switch 1 to prevent forwarding loops. Trunk 2 is not used to forward traffic unless Trunk 1 fails.

To configure the switches so that traffic from multiple VLANs is load balanced over the parallel trunks, perform these steps:

Step 1 Configure a VTP domain on both Switch 1 and Switch 2 by entering the **set vtp** command so that the VLAN information that is configured on Switch 1 is learned by Switch 2. Make sure that Switch 1 is a VTP server. You can configure Switch 2 as a VTP client or as a VTP server.

```
Switch_1> (enable) set vtp domain BigCorp mode server
VTP domain BigCorp modified
Switch_1> (enable)

Switch_2> (enable) set vtp domain BigCorp mode server
VTP domain BigCorp modified
Switch_2> (enable)
```

Step 2 Create the VLANs on Switch 1 by entering the **set vlan** command. In this example, you see VLANs 10, 20, 30, 40, 50, and 60.

```
Switch_1> (enable) set vlan 10
Vlan 10 configuration successful
Switch_1> (enable) set vlan 20
Vlan 20 configuration successful
Switch_1> (enable) set vlan 30
Vlan 30 configuration successful
Switch_1> (enable) set vlan 40
Vlan 40 configuration successful
```

```
Switch_1> (enable) set vlan 50
Vlan 50 configuration successful
Switch_1> (enable) set vlan 60
Vlan 60 configuration successful
Switch_1> (enable)
```

Step 3 Verify the VTP and VLAN configuration on Switch 1 by entering the **show vtp domain** and **show vlan** commands.

```
Switch_1> (enable) show vtp domain
Domain Name                    Domain Index VTP Version Local Mode  Password
------------------------------ ------------ ----------- ----------- ----------
BigCorp                        1            2           server      -

Vlan-count Max-vlan-storage Config Revision Notifications
---------- ---------------- --------------- -------------
11         1023             13              disabled

Last Updater    V2 Mode  Pruning  PruneEligible on Vlans
--------------- -------- -------- -------------------------
172.20.52.10    disabled enabled  2-1000
Switch_1> (enable) show vlan
VLAN Name                           Status    Mod/Ports, Vlans
---- ------------------------------ --------- ----------------------------
1    default                        active    1/1-2
                                               2/1-12
                                               5/1-2
10   VLAN0010                       active
20   VLAN0020                       active
30   VLAN0030                       active
40   VLAN0040                       active
50   VLAN0050                       active
60   VLAN0060                       active
1002 fddi-default                   active
1003 token-ring-default             active
1004 fddinet-default                active
1005 trnet-default                  active

Switch_1> (enable)
```

Step 4 Configure the supervisor engine uplinks on Switch 1 as ISL trunk ports by entering the **set trunk** command. Specifying the **desirable** mode on the Switch 1 ports causes the ports on Switch 2 to negotiate to become trunk links (assuming that the Switch 2 uplinks are in the default **auto** mode).

```
Switch_1> (enable) set trunk 1/1 desirable
Port(s) 1/1 trunk mode set to desirable.
Switch_1> (enable) 04/21/1998,03:05:05:DISL-5:Port 1/1 has become isl trunk

Switch_1> (enable) set trunk 1/2 desirable
Port(s) 1/2 trunk mode set to desirable.
Switch_1> (enable) 04/21/1998,03:05:13:DISL-5:Port 1/2 has become isl trunk
```

Step 5 Verify that the trunk links are up by entering the **show trunk** command.

```
Switch_1> (enable) show trunk 1
Port      Mode         Encapsulation  Status        Native vlan
--------  -----------  -------------  ------------  -----------
 1/1      desirable    isl            trunking      1
 1/2      desirable    isl            trunking      1
```

```
Port      Vlans allowed on trunk
--------  ---------------------------------------------------------------------
 1/1      1-1005, 1025-4094
 1/2      1-1005, 1025-4094

Port      Vlans allowed and active in management domain
--------  ---------------------------------------------------------------------
 1/1      1,10,20,30,40,50,60
 1/2      1,10,20,30,40,50,60

Port      Vlans in spanning tree forwarding state and not pruned
--------  ---------------------------------------------------------------------
 1/1
 1/2
Switch_1> (enable)
```

Step 6 Note that when the trunk links come up, VTP passes the VTP and VLAN configuration to Switch 2. Verify that Switch 2 has learned the VLAN configuration by entering the **show vlan** command on Switch 2.

```
Switch_2> (enable) show vlan
VLAN Name                             Status    Mod/Ports, Vlans
---- -------------------------------- --------- -----------------------------
1    default                          active
10   VLAN0010                         active
20   VLAN0020                         active
30   VLAN0030                         active
40   VLAN0040                         active
50   VLAN0050                         active
60   VLAN0060                         active
1002 fddi-default                     active
1003 token-ring-default               active
1004 fddinet-default                  active
1005 trnet-default                    active
.
.
.
Switch_2> (enable)
```

Step 7 Note that spanning tree takes 1 to 2 minutes to converge. After the network stabilizes, check the spanning-tree state of each trunk port on Switch 1 by entering the **show spantree** command.

Trunk 1 is forwarding for all VLANs. Trunk 2 is blocking for all VLANs. On Switch 2, both trunks are forwarding for all VLANs, but no traffic passes over Trunk 2 because port 1/2 on Switch 1 is blocking.

```
Switch_1> (enable) show spantree 1/1
Port      Vlan  Port-State     Cost   Priority  Fast-Start  Group-method
--------- ----  -------------  -----  --------  ----------  ------------
 1/1      1     forwarding        19        32   disabled
 1/1      10    forwarding        19        32   disabled
 1/1      20    forwarding        19        32   disabled
 1/1      30    forwarding        19        32   disabled
 1/1      40    forwarding        19        32   disabled
 1/1      50    forwarding        19        32   disabled
 1/1      60    forwarding        19        32   disabled
 1/1      1003  not-connected     19        32   disabled
 1/1      1005  not-connected     19         4   disabled
```

```
Switch_1> (enable) show spantree 1/2
Port      Vlan  Port-State     Cost   Priority  Fast-Start  Group-method
--------- ----  -------------  -----  --------  ----------  ------------
 1/2      1     blocking          19        32   disabled
 1/2      10    blocking          19        32   disabled
 1/2      20    blocking          19        32   disabled
 1/2      30    blocking          19        32   disabled
 1/2      40    blocking          19        32   disabled
 1/2      50    blocking          19        32   disabled
 1/2      60    blocking          19        32   disabled
 1/2      1003  not-connected     19        32   disabled
 1/2      1005  not-connected     19         4   disabled
Switch_1> (enable)
```

Step 8 Divide the configured VLANs into two groups. You might want traffic from half of the VLANs to go over one trunk link and half over the other, or if one VLAN has heavier traffic than the others, you can forward traffic from that VLAN over one trunk and traffic from the other VLANs over the other trunk link.

Note In the following steps, VLANs 10, 20, and 30 (Group 1) are forwarded over Trunk 1, and VLANs 40, 50, and 60 (Group 2) are forwarded over Trunk 2.

Step 9 On Switch 1, change the port-VLAN priority for the Group 1 VLANs on Trunk 1 (port 1/1) to an integer value lower than the default of 32 by entering the **set spantree portvlanpri** command.

```
Switch_1> (enable) set spantree portvlanpri 1/1 1 10
Port 1/1 vlans 1-9,11-1004 using portpri 32.
Port 1/1 vlans 10 using portpri 1.
Port 1/1 vlans 1005 using portpri 4.
Switch_1> (enable) set spantree portvlanpri 1/1 1 20
Port 1/1 vlans 1-9,11-19,21-1004 using portpri 32.
Port 1/1 vlans 10,20 using portpri 1.
Port 1/1 vlans 1005 using portpri 4.
Switch_1> (enable) set spantree portvlanpri 1/1 1 30
Port 1/1 vlans 1-9,11-19,21-29,31-1004 using portpri 32.
Port 1/1 vlans 10,20,30 using portpri 1.
Port 1/1 vlans 1005 using portpri 4.
Switch_1> (enable)
```

Step 10 On Switch 1, change the port-VLAN priority for the Group 2 VLANs on Trunk 2 (port 1/2) to an integer value lower than the default of 32 by entering the **set spantree portvlanpri** command.

```
Switch_1> (enable) set spantree portvlanpri 1/2 1 40
Port 1/2 vlans 1-39,41-1004 using portpri 32.
Port 1/2 vlans 40 using portpri 1.
Port 1/2 vlans 1005 using portpri 4.
Switch_1> (enable) set spantree portvlanpri 1/2 1 50
Port 1/2 vlans 1-39,41-49,51-1004 using portpri 32.
Port 1/2 vlans 40,50 using portpri 1.
Port 1/2 vlans 1005 using portpri 4.
Switch_1> (enable) set spantree portvlanpri 1/2 1 60
Port 1/2 vlans 1-39,41-49,51-59,61-1004 using portpri 32.
Port 1/2 vlans 40,50,60 using portpri 1.
Port 1/2 vlans 1005 using portpri 4.
Switch_1> (enable)
```

Step 11 On Switch 2, change the port-VLAN priority for the Group 1 VLANs on Trunk 1 (port 1/1) to the same value that you configured for those VLANs on Switch 1 by entering the **set spantree portvlanpri** command.

Caution The port-VLAN priority for each VLAN must be equal on both ends of the link.

```
Switch_2> (enable) set spantree portvlanpri 1/1 1 10
Port 1/1 vlans 1-9,11-1004 using portpri 32.
Port 1/1 vlans 10 using portpri 1.
Port 1/1 vlans 1005 using portpri 4.
Switch_2> (enable) set spantree portvlanpri 1/1 1 20
Port 1/1 vlans 1-9,11-19,21-1004 using portpri 32.
Port 1/1 vlans 10,20 using portpri 1.
Port 1/1 vlans 1005 using portpri 4.
Switch_2> (enable) set spantree portvlanpri 1/1 1 30
Port 1/1 vlans 1-9,11-19,21-29,31-1004 using portpri 32.
Port 1/1 vlans 10,20,30 using portpri 1.
Port 1/1 vlans 1005 using portpri 4.
Switch_2> (enable)
```

Step 12 On Switch 2, change the port-VLAN priority for the Group 2 VLANs on Trunk 2 (port 1/2) to the same value that you configured for those VLANs on Switch 1 by entering the **set spantree portvlanpri** command.

```
Switch_2> (enable) set spantree portvlanpri 1/2 1 40
Port 1/2 vlans 1-39,41-1004 using portpri 32.
Port 1/2 vlans 40 using portpri 1.
Port 1/2 vlans 1005 using portpri 4.
Switch_2> (enable) set spantree portvlanpri 1/2 1 50
Port 1/2 vlans 1-39,41-49,51-1004 using portpri 32.
Port 1/2 vlans 40,50 using portpri 1.
Port 1/2 vlans 1005 using portpri 4.
Switch_2> (enable) set spantree portvlanpri 1/2 1 60
Port 1/2 vlans 1-39,41-49,51-59,61-1004 using portpri 32.
Port 1/2 vlans 40,50,60 using portpri 1.
Port 1/2 vlans 1005 using portpri 4.
Switch_2> (enable)
```

Note When you configure the port-VLAN priorities on both ends of the link, the spanning tree converges to use the new configuration.

Step 13 Check the spanning-tree port states on Switch 1 by entering the **show spantree** command. The Group 1 VLANs should forward on Trunk 1 and block on Trunk 2. The Group 2 VLANs should block on Trunk 1 and forward on Trunk 2.

```
Switch_1> (enable) show spantree 1/1
Port      Vlan  Port-State     Cost   Priority  Fast-Start  Group-method
--------- ----  -------------  -----  --------  ----------  ------------
 1/1      1     forwarding        19        32   disabled
 1/1      10    forwarding        19         1   disabled
 1/1      20    forwarding        19         1   disabled
 1/1      30    forwarding        19         1   disabled
 1/1      40    blocking          19        32   disabled
 1/1      50    blocking          19        32   disabled
 1/1      60    blocking          19        32   disabled
 1/1      1003  not-connected     19        32   disabled
 1/1      1005  not-connected     19         4   disabled
```

```
Switch_1> (enable) show spantree 1/2
Port      Vlan  Port-State     Cost   Priority  Fast-Start  Group-method
--------- ----  -------------  -----  --------  ----------  ------------
 1/2      1     blocking          19        32   disabled
 1/2      10    blocking          19        32   disabled
 1/2      20    blocking          19        32   disabled
 1/2      30    blocking          19        32   disabled
 1/2      40    forwarding        19         1   disabled
 1/2      50    forwarding        19         1   disabled
 1/2      60    forwarding        19         1   disabled
 1/2      1003  not-connected     19        32   disabled
 1/2      1005  not-connected     19         4   disabled
Switch_1> (enable)
```

Figure 5-4 shows the network after you configure VLAN traffic load sharing.

Figure 5-4 Parallel Trunk Configuration After Configuring VLAN-Traffic Load Sharing

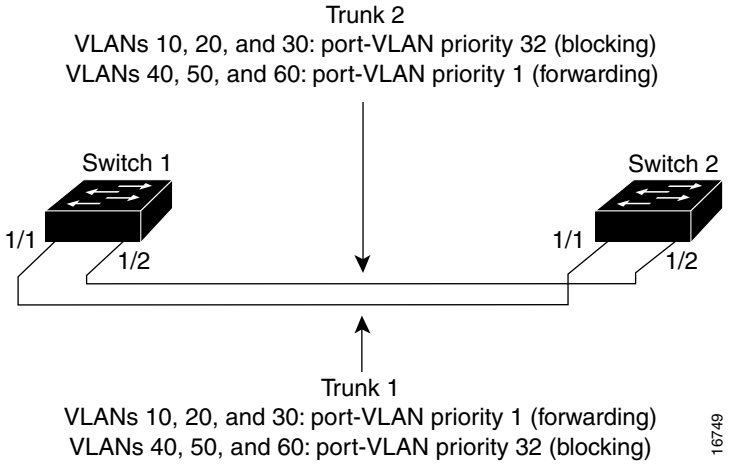


Figure 5-4 shows that both trunks are used when the network is operating normally; if one trunk link fails, the other trunk link acts as an alternate forwarding path for the traffic previously traveling over the failed link.

If Trunk 1 fails in the network that is shown in Figure 5-4, STP reconverges to use Trunk 2 to forward traffic from all the VLANs, as shown as follows:

```
Switch_1> (enable) 04/21/1998,03:15:40:DISL-5:Port 1/1 has become non-trunk

Switch_1> (enable) show spantree 1/1
Port      Vlan  Port-State     Cost   Priority  Fast-Start  Group-method
--------- ----  -------------  -----  --------  ----------  ------------
 1/1      1     not-connected     19        32   disabled
```

```
Switch_1> (enable) show spantree 1/2
Port      Vlan  Port-State     Cost   Priority  Fast-Start  Group-method
--------- ----  -------------  -----  --------  ----------  ------------
 1/2      1     learning          19        32   disabled
 1/2      10    learning          19        32   disabled
 1/2      20    learning          19        32   disabled
 1/2      30    learning          19        32   disabled
 1/2      40    forwarding        19         1   disabled
 1/2      50    forwarding        19         1   disabled
 1/2      60    forwarding        19         1   disabled
 1/2      1003  not-connected     19        32   disabled
 1/2      1005  not-connected     19         4   disabled
Switch_1> (enable) show spantree 1/2
Port      Vlan  Port-State     Cost   Priority  Fast-Start  Group-method
--------- ----  -------------  -----  --------  ----------  ------------
 1/2      1     forwarding        19        32   disabled
 1/2      10    forwarding        19        32   disabled
 1/2      20    forwarding        19        32   disabled
 1/2      30    forwarding        19        32   disabled
 1/2      40    forwarding        19         1   disabled
 1/2      50    forwarding        19         1   disabled
 1/2      60    forwarding        19         1   disabled
 1/2      1003  not-connected     19        32   disabled
 1/2      1005  not-connected     19         4   disabled
Switch_1> (enable)
```

CHAPTER 6

Configuring EtherChannel

This chapter describes how to use the command-line interface (CLI) to configure EtherChannel on the Catalyst 6500 series switches. The configuration tasks in this chapter apply to Ethernet, Fast Ethernet, Gigabit Ethernet, and 10-Gigabit Ethernet switching modules and the uplink ports on the supervisor engine.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Note With software release 8.4(1) and later releases, you can configure EtherChannel error handling to provide for the automatic failover of traffic to another port in the EtherChannel when one of the ports in the channel exceeds a configurable error threshold. For more information, see the "Configuring EtherChannel/Link Error Handling" section on page 20-24.

Note You can use the commands in the following sections on all Ethernet ports in the Catalyst 6500 series switches.

Understanding How EtherChannel Works

EtherChannel aggregates the bandwidth of up to eight compatibly configured ports into a single logical link. Catalyst 6500 series switches support a maximum of 128 EtherChannels. All Ethernet ports on all modules, including those on a standby supervisor engine, support EtherChannel with no requirement that ports be contiguous or on the same module. All ports in each EtherChannel must be the same speed.

Note With software release 6.3(1) and later releases, due to the port ID handling by the spanning-tree feature, the maximum supported number of EtherChannels is 126 for a 6- or 9-slot chassis and 63 for a 13-slot chassis.

Note The network device to which a Catalyst 6500 series switch is connected may impose its own limits on the number of ports in an EtherChannel.

If a link within an EtherChannel fails, the traffic that was previously carried over the failed link switches to the remaining links within the EtherChannel. Inbound broadcast and multicast packets on one link in an EtherChannel are blocked from returning on any other link of the EtherChannel.

You can configure EtherChannels as trunks. After a channel is formed, configuring any port in the channel as a trunk applies the configuration to all ports in the channel. Identically configured trunk ports can be configured as an EtherChannel.

Understanding How EtherChannel Frame Distribution Works

EtherChannel distributes frames across the links in a channel by reducing part of the binary pattern formed from the addresses in the frame to a numerical value that selects one of the links in the channel.

EtherChannel frame distribution is based on a Cisco-proprietary hashing algorithm. The algorithm is deterministic; given the same addresses and session information, you always hash to the same port in the channel, preventing out-of-order packet delivery.

The address may be a source, a destination, or a combination of two IP addresses, two MAC addresses, or two TCP/UDP port numbers depending on the policy that is adopted through the **ip**, **mac**, **session**, and **ip-vlan-session** options of the **set port channel all distribution** command. See the "Configuring EtherChannel Load Balancing" section on page 6-11 for detailed information.

Note The **set port channel all distribution session** command is supported on Supervisor Engine 2, Supervisor Engine 720, and Supervisor Engine 32 only. The **set port channel all distribution ip-vlan-session** command is supported on Supervisor Engine 720 and Supervisor Engine 32 only.

EtherChannel frame distribution is not configurable on all supervisor engines. Enter the **show module** command on a supervisor engine to determine if EtherChannel frame distribution is configurable on your switch. If the display shows the "Sub-Type" to be "L2 Switching Engine I WS-F6020," then EtherChannel frame distribution is not configurable on your Catalyst 6500 series switch; the switch uses source and destination Media Access Control (MAC) addresses.

EtherChannel frame distribution is configurable with all other switching engines. The default is to use source and destination IP addresses.

Port Aggregation Control Protocol and Link Aggregation Control Protocol

Port Aggregation Control Protocol (PAgP) and Link Aggregation Control Protocol (LACP) are two different protocols that allow ports with similar characteristics to form a channel through dynamic negotiation with adjoining switches. PAgP is a Cisco-proprietary protocol that can be run only on Cisco switches and those switches that are released by licensed vendors. LACP, which is defined in IEEE 802.3ad, allows Cisco switches to manage Ethernet channeling with devices that conform to the 802.3ad specification.

Note MAC address notification settings are ignored on PAgP and LACP EtherChannel ports.

To use PAgP, see the "Understanding How the Port Aggregation Protocol Works" section on page 6-5. To use LACP, see the "Understanding How the Link Aggregation Control Protocol Works" section on page 6-13.

EtherChannel Configuration Guidelines

If improperly configured, some EtherChannel ports are disabled automatically to avoid network loops and other problems.

Note Except where specifically differentiated, these guidelines apply to both PAgP and LACP.

These sections provide the guidelines for EtherChannel configuration:

Port Configuration Guidelines

This section describes the guidelines for port configuration:

- You can have a maximum of eight compatibly configured ports per EtherChannel; the ports do not have to be contiguous or on the same module.
- All ports in an EtherChannel must be placed in the same administrative group. Enter the **show channel group** command to verify the administrative group, and enter the **set port channel** *mod/ports admin_group* command if necessary to assign ports to the same administrative group.
- All ports in an EtherChannel must use the same protocol; you cannot run two protocols on one module.
- PAgP and LACP are not compatible; both ends of a channel must use the same protocol.

Note You can configure the switch manually with PAgP on one side and LACP on the other side in the **on** mode.

- You can change the protocol at any time, but this change causes all existing EtherChannels to reset to the default channel mode for the new protocol.
- Configure all ports in an EtherChannel to operate at the same speed and duplex mode (full duplex only for LACP mode).
- Enable all ports in an EtherChannel. If you disable a port in an EtherChannel, it is treated as a link failure and its traffic is transferred to one of the remaining ports in the EtherChannel.
- A port cannot belong to more than one channel group at the same time.
- Ports with different port path costs, set by the **set spantree portcost** command, can form an EtherChannel as long as they are otherwise compatibly configured. Setting different port path costs does not, by itself, make ports incompatible for the formation of an EtherChannel.
- PAgP and LACP manage channels differently. When all the ports in a channel get disabled, PAgP removes them from its internal channels list; the **show** commands do not display the channel. With LACP, when all the ports in a channel get disabled, LACP does not remove the channel; the **show** commands continue to display the channel even though all its ports are down. To determine if a channel is actively sending and receiving traffic with LACP, enter the **show port** command to see if the link is up or down.
- LACP does not support half-duplex links. If a port is in active/passive mode and becomes half duplex, the port is suspended (and a syslog message is generated). The port is shown as "connected" when you enter the **show port** command and as "not connected" when you enter the **show spantree** command. This discrepancy occurs because the port is physically connected but never joined spanning tree. To get the port to join spanning tree, either set the duplex to full or set the channel mode to off for that port.

  With software release 7.3(1) and later releases, the LACP behavior for the half-duplex links has changed and the affected ports are no longer suspended. Instead of suspending a port, the LACP PDU transmission (if any) is suppressed. If the port is part of a channel, the port is detached from the channel but still functions as a nonchannel port. A syslog message is generated when this condition occurs. The normal LACP behavior is reenabled automatically when you set the link to full duplex.

VLAN and Trunk Configuration Guidelines

This section describes the guidelines for VLAN and trunk-related configuration:

- Assign all ports in an EtherChannel to the same VLAN, or configure them as trunk ports.
- If you configure the EtherChannel as a trunk, configure the same trunk mode on all the ports in the EtherChannel. Configuring ports in an EtherChannel in different trunk modes can have unexpected results.
- An EtherChannel supports the same allowed range of VLANs on all the ports in a trunking EtherChannel. If the allowed range of VLANs is not the same for a port list, the ports do not form an EtherChannel even when set to the **auto** or **desirable** mode with the **set port channel** command.
- Do not configure the ports in an EtherChannel as dynamic VLAN ports. Doing so can adversely affect switch performance.
- Ports with different VLAN cost configurations cannot form a channel.

Interaction with Other Features Guidelines

This section describes the guidelines for the EtherChannel's interaction with other features:

- An EtherChannel does not form with ports that have different GARP VLAN Registration Protocol (GVRP), GARP Multicast Registration Protocol (GMRP), and QoS configurations.
- An EtherChannel does not form with ports when you enable port security. You cannot enable port security for ports in an EtherChannel.
- An EtherChannel does not form if one of the ports is a SPAN destination port.
- An EtherChannel does not form if protocol filtering is set differently on the ports.
- Cisco Discovery Protocol (CDP) runs on the physical port even after the port is added to a channel.
- VLAN Trunking Protocol (VTP) and Dual Ring Protocol (DRiP) run on the channel.
- During fast switchover to the standby supervisor engine, all channeling ports are cleared on the standby supervisor engine channeling configuration and state, and the links are pulled down temporarily to cause partner ports to reset. All ports are reset to the nonchanneling state.
- Ports with different dot1q port types cannot form a channel.
- Ports with different jumbo frame configurations cannot form a channel.
- Ports with different dynamic configurations cannot form a channel.
- During high-availability switchover to the standby supervisor engine, all channeling ports remain operational. Ports are reset only if there are events missing during the switchover.

Note With software release 6.3(1) and later releases, a PAgP-configured EtherChannel is preserved even if it contains only one port (this situation does not apply to LACP-configured EtherChannels). In software releases prior to 6.3(1), traffic was disrupted when you removed a 1-port channel from spanning tree and then added it to spanning tree as an individual port.

Note With software release 6.3(1) and later releases, due to the port ID handling by the spanning-tree feature, the maximum number of EtherChannels is 126 for a 6- or 9-slot chassis and 63 for a 13-slot chassis.

Understanding How the Port Aggregation Protocol Works

Note Use the information in these sections if you are configuring EtherChannel using PAgP. If you are using LACP, see the "Understanding How the Link Aggregation Control Protocol Works" section on page 6-13.

These sections describe PAgP:

PAgP Modes

PAgP facilitates the automatic creation of EtherChannels by exchanging packets between Ethernet ports. PAgP packets are exchanged only between ports in **auto** and **desirable** modes. Ports that are configured in **on** or **off** mode do not exchange PAgP packets. The protocol learns the capabilities of port groups dynamically and informs the other ports. After PAgP identifies correctly matched EtherChannel links, it groups the ports into an EtherChannel. The EtherChannel is then added to the spanning tree as a single bridge port.

EtherChannel includes four user-configurable modes: **on**, **off**, **auto**, and **desirable**. Only **auto** and **desirable** are PAgP modes. You can modify the **auto** and **desirable** modes with the **silent** and **non-silent** keywords. By default, the ports are in **auto silent** mode.

Table 6-1 describes the EtherChannel modes that are available in PAgP.

Table 6-1 EtherChannel Modes Available in PAgP

Mode	Description
on	Mode that forces the port to channel without PAgP. With the **on** mode, a usable EtherChannel exists only when a port group in **on** mode is connected to another port group in **on** mode.
off	Mode that prevents the port from channeling.
auto	PAgP mode that places a port into a passive negotiating state in which the port responds to PAgP packets that it receives but does not initiate PAgP packet negotiation. (Default)
desirable	PAgP mode that places a port into an active negotiating state in which the port initiates negotiations with other ports by sending PAgP packets.
silent	Keyword that is used with the **auto** or **desirable** mode when no traffic is expected from the other device to prevent the link from being reported to the Spanning Tree Protocol as down. (Default)
non-silent	Keyword that is used with the **auto** or **desirable** mode when traffic is expected from the other device.

Both the **auto** and **desirable** modes allow ports to negotiate with connected ports to determine if they can form an EtherChannel, based on criteria such as port speed, trunking state, and VLAN numbers.

Ports can form an EtherChannel when they are in different PAgP modes as long as the modes are compatible, as follows:

- A port in **desirable** mode can form an EtherChannel with another port that is in **desirable** or **auto** mode.
- A port in **auto** mode can form an EtherChannel with another port in **desirable** mode.
- A port in **auto** mode cannot form an EtherChannel with another port that is also in **auto** mode, because neither port will initiate negotiation.

When configurable, EtherChannel frame distribution can use MAC addresses, IP addresses, and Layer 4 port numbers. You can specify either the source or the destination address or both the source and destination addresses and Layer 4 port numbers. The mode that you select applies to all EtherChannels that are configured on the switch. Use the option that provides the greatest variety in your configuration. For example, if the traffic on a channel is going to a single MAC address only, using source addresses, IP addresses, or Layer 4 port numbers for frame distribution may provide better frame distribution than selecting MAC addresses.

PAgP Administrative Groups

Configuring an EtherChannel creates an administrative group, which is designated by an integer between 1 and 1024, to which the EtherChannel belongs. When an administrative group is created, you can assign an administrative group number or let the next available administrative group number be assigned automatically. Forming a channel without specifying an administrative group number creates a new automatically numbered administrative group. An administrative group may contain a maximum of eight ports.

Enter the **show channel group** command to verify that all ports in an EtherChannel belong to the same administrative group, and enter the **set port channel** *mod/ports admin_group* command if necessary to assign ports to the same administrative group.

PAgP EtherChannel IDs

Each EtherChannel is automatically assigned a unique EtherChannel ID. Enter the **show channel group** *admin_group* command to display the EtherChannel ID.

Configuring an EtherChannel Using PAgP

These sections describe how to configure an EtherChannel using PAgP:

Note Before you configure the EtherChannel, see the "EtherChannel Configuration Guidelines" section on page 6-3.

Specifying the EtherChannel Protocol

Note The default protocol is PAgP.

Note You can specify only one protocol, PAgP or LACP, per module.

To specify the EtherChannel protocol, perform this task in privileged mode:

Task	Command
Specify the EtherChannel protocol.	**set channelprotocol** [**pagp** | **lacp**] *mod*

This example shows how to specify the PAgP protocol for module 3:

```
Console> (enable) set channelprotocol pagp 3
Channeling protocol set to PAGP for module(s) 3.
Console> (enable)
```

Configuring an EtherChannel

To configure an EtherChannel on a group of Ethernet ports, perform this task in privileged mode:

Task	Command
Configure an EtherChannel on the desired ports.	**set port channel** *mod/ports...* [*admin_group*] **set port channel** *mod/ports...* **mode** {**on** | **off** | **desirable** | **auto**} [**silent** | **non-silent**]

Enter the **show channel group** command to verify that all ports in the EtherChannel belong to the same administrative group, and enter the **set port channel** *mod/ports admin_group* command if necessary to assign ports to the same administrative group.

This example shows how to configure a seven-port EtherChannel in a new administrative group:

```
Console> (enable) set port channel 2/2-8 mode desirable
Ports 2/2-8 left admin_group 1.
Ports 2/2-8 joined admin_group 2.
Console> (enable)
```

Setting the EtherChannel Port Mode

To set a port's EtherChannel mode, perform this task in privileged mode:

Task	Command
Set a port's EtherChannel mode.	**set port channel** *mod/ports...* [*admin_group*] **set port channel** *mod/port* **mode** {**on** | **off** | **desirable** | **auto**} [**silent** | **non-silent**]

This example shows how to set port 2/1 to **auto** mode:

```
Console> (enable) set port channel 2/1 mode auto
Ports 2/1 channel mode set to auto.
Console> (enable)
```

Setting the EtherChannel Port Path Cost

Note You accomplish this task using a global command that configures both LACP and PAgP.

The channel path cost is achieved by adjusting the port costs of each port belonging to the channel. If you do not specify the cost, it is updated based on the current port costs of the channeling ports. You may address one channel or all channels.

To set the EtherChannel port path cost, perform this task in privileged mode:

	Task	Command
Step 1	Use the administrative group number to display the EtherChannel ID.	**show channel group** *admin_group* or **show lacp-channel group** *admin_key*
Step 2	Use the EtherChannel ID to set the EtherChannel port path cost.	**set spantree channelcost** {*channel_id* | **all**} *cost*

Note When you enter the **set spantree channelcost** command, it does not appear in the configuration file. The command causes a "set spantree portcost" entry to be created for each port in the channel. See the "Configuring the PVST+ Port Cost" section in Chapter 7, "Configuring Spanning Tree," for information on using the **set spantree portcost** command.

This example shows how to set the EtherChannel port path cost for channel ID 768:

```
Console> (enable) show channel group 20
Admin Port  Status     Channel   Channel
group                  Mode      id
----- ----- ---------- --------- --------
  20    1/1 notconnect on             768
  20    1/2 connected  on             768

Admin Port  Device-ID                      Port-ID                  Platform
group
----- ----- ------------------------------ ------------------------ ----------
  20    1/1
  20    1/2  066510644(cat26-lnf(NET25))     2/1                     WS-C6009
Console> (enable)

Console> (enable) set spantree channelcost 768 12
Port(s) 1/1,1/2 port path cost are updated to 31.
Channel 768 cost is set to 12.
Warning:channel cost may not be applicable if channel is broken.
Console> (enable)
```

Setting the EtherChannel VLAN Cost

Note You accomplish this task by using a global command that configures both LACP and PAgP.

The EtherChannel VLAN cost feature provides load balancing of VLAN traffic across multiple channels that are configured with trunking.

You enter the **set spantree channelvlancost** command to set the initial spanning-tree costs for all VLANs in the channel. The **set spantree channelvlancost** command provides an alternate cost for some of the VLANs in the channel (assuming that you are trunking across the channel). This command allows you to have up to two different spanning-tree costs assigned per channel; some VLANs in the channel can have the "vlancost" while the remaining VLANs in the channel have the "cost."

The **set spantree channelvlancost** command creates a "set spantree portvlancost" entry to the configuration file for each port in the channel. After you enter the **set spantree channelvlancost** command, you must enter the **set spantree portvlancost** command for at least one port in the channel, specifying the VLAN or VLANs that you want associated with each port. This example shows what occurs when you enter each command:

```
Console> (enable) set spantree channelvlancost 856 10
Port(s) 3/47-48 vlan cost are updated to 16.
Channel 856 vlancost is set to 10.
```

These commands are added to the configuration file:

- **set spantree portvlancost 3/47 cost 16**
- **set spantree portvlancost 3/48 cost 16**

To add the desired VLANs to the above created commands, enter this command:

```
Console> (enable) set spantree portvlancost 3/47 cost 16 1-1005
Port 3/47 VLANs 1025-4094 have path cost 19.
Port 3/47 VLANs 1-1005 have path cost 16.
Port 3/48 VLANs 1-1005 have path cost 16.
```

To set the EtherChannel VLAN cost, perform this task in privileged mode:

	Task	Command
Step 1	Use the administrative group number to display the EtherChannel ID.	**show channel group** *admin_group* or **show lacp-channel group** *admin_key*
Step 2	Use the EtherChannel ID to set the EtherChannel VLAN cost.	**set spantree channelvlancost** *channel_id cost*
Step 3	Configure the port cost for the desired VLANs on each port.	**set spantree portvlancost** {*mod/port*} [**cost** *cost*] [*vlan_list*]

This example shows how to set the EtherChannel VLAN cost for channel ID 856:

```
Console> (enable) show channel group 22
Admin Port  Status     Channel   Channel
group                  Mode      id
----- ----- ---------- --------- --------
  22    1/1 notconnect on             856
  22    1/2 connected  on             856

Admin Port  Device-ID                        Port-ID                   Platform
group
----- ----- ------------------------------ ------------------------- ----------
  22    1/1
  22    1/2  066510644(cat26-lnf(NET25))     2/1                       WS-C6009
Console> (enable)
```

```
Console> (enable) set spantree channelvlancost 856 10
Port(s) 3/47-48 vlan cost are updated to 16.
Channel 856 vlancost is set to 10.
Console> (enable) set spantree portvlancost 3/47 cost 16 1-1005
Port 3/47 VLANs 1025-4094 have path cost 19.
Port 3/47 VLANs 1-1005 have path cost 16.
Port 3/48 VLANs 1-1005 have path cost 16.
Console> (enable)
```

Configuring EtherChannel Load Balancing

The load-balancing policy (frame distribution) can be based on a MAC address (Layer 2), an IP address (Layer 3), or a port number (Layer 4). These policies can be activated, respectively, by the **mac**, **ip**, and **session** keywords. The load balancing can be based solely on the source address (**source** keyword), destination address (**destination** keyword), or both source and destination addresses (**both** keyword).

If a packet does not belong to a selected category, the next lower level category is considered. If the hardware cannot support the frame distribution method that is selected, a "Feature not supported" error message is displayed.

To configure EtherChannel load balancing, perform this task in privileged mode:

Task	Command
Configure EtherChannel load balancing.	**set port channel all distribution** {**ip** | **mac** | **session** | **ip-vlan-session**} [**source** | **destination** | **both**]

Note The **set port channel all distribution session** command option is supported on Supervisor Engine 2, Supervisor Engine 720, and Supervisor Engine 32 only.

Note The **set port channel all distribution ip-vlan-session** command is supported on Supervisor Engine 720 and Supervisor Engine 32 only. Use the command to specify the frame distribution method using the IP address, VLAN, and Layer 4 traffic.

This example shows how to configure EtherChannel to use MAC source addresses:

```
Console> (enable) set port channel all distribution mac source
Channel distribution is set to mac source.
Console> (enable)
```

Displaying EtherChannel Traffic Utilization

To display the traffic utilization on the EtherChannel ports, perform this task:

Task	Command
Display traffic utilization.	**show channel traffic**

This example shows how to display traffic utilization on EtherChannel ports:

```
Console> (enable) show channel traffic
ChanId Port  Rx-Ucst Tx-Ucst Rx-Mcst Tx-Mcst Rx-Bcst Tx-Bcst
------ ----- ------- ------- ------- ------- ------- -------
   808  2/16   0.00%   0.00%  50.00%  75.75%   0.00%   0.00%
   808  2/17   0.00%   0.00%  50.00%  25.25%   0.00%   0.00%
   816  2/31   0.00%   0.00%  25.25%  50.50%   0.00%   0.00%
   816  2/32   0.00%   0.00%  75.75%  50.50%   0.00%   0.00%
Console> (enable)
```

Displaying the Outgoing Ports for a Specified Address or Layer 4 Port Number

To display the outgoing port that is used in an EtherChannel for a specific address or Layer 4 port number, perform this task:

Task	Command
Display the outgoing port for a specified address or Layer 4 port number.	**show channel hash** *channel_id src_ip_addr vlan src_port* [*dest_ip_addr vlan dest_port*] **show channel hash** *channel_id dest_ip_addr vlan dest_port*

This example shows how to display the outgoing port for the specified source and destination IP addresses:

```
Console> (enable) show channel hash 808 172.20.32.10 172.20.32.66
Selected channel port:2/17
Console> (enable)
```

Disabling an EtherChannel

To disable an EtherChannel, perform this task in privileged mode:

Task	Command
Disable an EtherChannel.	**set port channel** *mod/port* **mode off**

This example shows how to disable an EtherChannel:

```
Console> (enable) set port channel 2/2-8 mode off
Ports 2/2-8 channel mode set to off.
Console> (enable)
```

Understanding How the Link Aggregation Control Protocol Works

Note Use the information in these sections if you are configuring EtherChannel using LACP. If you are using PAgP, see the "Understanding How the Port Aggregation Protocol Works" section on page 6-5.

This section contains the following descriptions:

LACP Modes

You may manually turn on channeling by setting the port channel mode to **on**, and you may turn off channeling by setting the port channel mode to **off**.

If you want LACP to handle channeling, use the **active** and **passive** channel modes. To start automatic EtherChannel configuration with LACP, you need to configure at least one end of the link to **active** mode to initiate channeling, because ports in **passive** mode passively respond to initiation and never initiate the sending of LACP packets.

Table 6-2 describes the EtherChannel modes that are available in LACP.

Table 6-2 EtherChannel Modes Available in LACP

Mode	Description
on	Mode that forces the port to channel without LACP. With the **on** mode, a usable EtherChannel exists only when a port group in **on** mode is connected to another port group in **on** mode.
off	Mode that prevents the port from channeling.
passive	LACP mode that places a port into a passive negotiating state in which the port responds to LACP packets that it receives but does not initiate LACP packet negotiation. (Default)
active	LACP mode that places a port into an active negotiating state in which the port initiates negotiations with other ports by sending LACP packets.

LACP Parameters

The parameters that are used in configuring LACP are as follows:

- System priority

  You must assign a system priority that can be specified automatically or through the CLI (see the "Specifying the System Priority" section on page 6-16) to each switch running LACP. The system priority is used with the switch MAC address to form the system ID and is also used during negotiation with other systems.

- Port priority

You must assign a port priority that can be specified automatically or through the CLI (see the "Specifying the Port Priority" section on page 6-16) to each port in the switch. The port priority is used with the port number to form the port identifier. The port priority is used to decide which ports should be put in standby mode when there is a hardware limitation that prevents all compatible ports from aggregating.

- Administrative key

  You must assign an administrative key value that can be specified automatically or through the CLI to each port in the switch (see the "Specifying an Administrative Key Value" section on page 6-16). The ability of a port to aggregate with other ports is defined with the administrative key. A port's ability to aggregate with other ports is determined by these factors:

  - Port physical characteristics, such as data rate, duplex capability, and point-to-point or shared medium
  - Configuration constraints that you establish

When enabled, LACP always tries to configure the maximum number of compatible ports in a channel, up to the maximum that is allowed by the hardware (eight ports). If LACP is not able to aggregate all the ports that are compatible (for example, the remote system might have more restrictive hardware limitations), then all the ports that cannot be actively included in the channel are put in hot standby state and are used only if one of the channeled ports fails.

You can configure different channels with ports that have been assigned the same administrative key. For example, if eight ports are assigned the same administrative key, you may configure four ports in a channel using LACP **active** mode and the remaining four ports in a manually configured channel using the **on** mode. An administrative key is meaningful only in the context of the switch that allocates it; there is no global significance to administrative key values.

Configuring an EtherChannel Using LACP

These sections describe how to configure EtherChannel using LACP:

Note Before you configure the EtherChannel, see the "EtherChannel Configuration Guidelines" section on page 6-3.

Specifying the EtherChannel Protocol

Note The default protocol is PAgP.

Note You can specify only one protocol, PAgP or LACP, per module.

To specify the EtherChannel protocol, perform this task in privileged mode:

Task	Command
Specify the EtherChannel protocol.	**set channelprotocol** [**pagp** | **lacp**] *mod*

This example shows how to specify the LACP protocol for modules 2 and 3:

```
Console> (enable) set channelprotocol lacp 2,3
Mod 2 is set to LACP protocol.
Mod 3 is set to LACP protocol.
Console> (enable)
```

Use the **show channelprotocol** command to display the protocols for all modules.

Specifying the System Priority

Note Although this command is a global option, the command applies only to modules on which LACP is enabled; it is ignored on modules running PAgP.

The system priority value must be a number in the range of 1–65535, where higher numbers represent lower priority. The default priority is 32768.

To specify the system priority, perform this task in privileged mode:

Task	Command
Specify the system priority.	**set lacp-channel system-priority** *value*

This example shows how to specify the system priority as 20000:

```
Console> (enable) set lacp-channel system-priority 20000
LACP system priority is set to 20000
Console> (enable)
```

Use the **show lacp-channel sys-id** command to display the LACP system ID and system priority.

Specifying the Port Priority

The port priority value must be a number in the range of 1–255, where higher numbers represent lower priority. The default priority is 128.

To specify the port priority, perform this task in privileged mode:

Task	Command
Specify the port priority.	**set port lacp-channel** *mod/ports* **port-priority** *value*

This example shows how to specify the port priority as 10 for ports 1/1 to 1/4 and 2/6 to 2/8:

```
Console> (enable) set port lacp-channel 1/1-4,2/6-8 port-priority 10
Port(s) 1/1-4,2/6-8 port-priority set to 10.
Console> (enable)
```

Use the **show lacp-channel group** *admin_key* **info** command to display the port priority.

Specifying an Administrative Key Value

Note When the system or module configuration information that is stored in NVRAM is cleared, the administrative keys are assigned new values automatically. For modules, each group of four consecutive ports, beginning at the 1st, 5th, 9th and so on, are assigned a unique administrative key. Across the module, ports must have unique administrative keys. After NVRAM is cleared, the channel mode of the ports is set to “passive.”

You can specify an administrative key value to a set of ports or the system automatically selects a value if you do not specify the parameter *admin_key*. In both cases, the admin_key value can range from 1–1024.

If you choose a value for the administrative key, and this value has already been used in the system, then all the ports that were originally associated with the previously assigned admin_key value are moved to another automatically assigned value, and the modules and ports that you specified in the command are assigned the admin_key value that you specified.

The maximum number of ports to which an administrative key can be assigned is eight.

The default mode for all ports being assigned the administrative key is passive. However, if the channel was previously assigned a particular mode (see the "Changing the Channel Mode" section on page 6-18), assigning the administrative key will not affect it, and the channel mode that you specified previously is maintained.

To specify the administrative key value, perform this task in privileged mode:

Task	Command
Specify the administrative key value.	**set port lacp-channel** *mod/ports* [*admin_key*]

This example shows how to assign the same administrative key to ports 4/1 to 4/4, with the system picking its value automatically:

```
Console> (enable) set port lacp-channel 4/1-4
Port(s) 4/1-4 are assigned to admin key 96.
Console> (enable)
```

This example shows how to assign the administrative key 96 (you specify the 96) to ports 4/4 to 4/6. In this example, the administrative key was previously assigned to another group of ports by the system (see the previous example):

```
Console> (enable) set port lacp-channel 4/4-6 96
Port(s) 4/1-3 are moved to admin key 97.
Port(s) 4/4-6 are assigned to admin key 96.
Console> (enable)
```

This example shows the system response when more than eight ports are assigned the same administrative key value (the request is denied, and no ports are assigned administrative key 123):

```
Console> (enable) set port lacp-port channel 2/1-2,4/1-8 123
No more than 8 ports can be assigned to an admin key.
Console> (enable)
```

Enter the **show lacp-channel group** command to display administrative key values for ports, and to verify that all ports in an EtherChannel share the same administrative key value. Enter the **set port lacp-channel** *mod/ports admin_key* command if necessary to assign all EtherChannel ports to the same administrative key.

Changing the Channel Mode

You can change the channel mode for a set of ports that were previously assigned the same administrative key (see the "Specifying an Administrative Key Value" section on page 6-16).

To change the channel mode, perform this task in privileged mode:

Task	Command
Change the channel mode.	**set port lacp-channel** *mod/ports* **mode** [**on** | **off** | **active** | **passive**]

This example shows how to change the channel mode for ports 4/1 and 4/6, setting it to **on**. The administrative key for ports 4/1 and 4/6 is unchanged.

```
Console> (enable) set port lacp-channel 4/1,4/6 mode on
Port(s) 4/1,4/6 channel mode set to on.
Console> (enable)
```

Use the **show lacp-channel group** *admin_key* command to display the channel mode for ports.

Specifying the Channel Path Cost

You can specify the channel path cost by using a global command that configures both LACP and PAgP. For more information, see the "Setting the EtherChannel Port Path Cost" section on page 6-9.

Specifying the Channel VLAN Cost

You can specify the channel VLAN cost by using a global command that configures both LACP and PAgP. For more information, see the "Setting the EtherChannel VLAN Cost" section on page 6-9.

Configuring Channel Load Balancing

You can configure channel load balancing by using a global command that configures both LACP and PAgP. For more information, see the "Configuring EtherChannel Load Balancing" section on page 6-11.

Clearing the LACP Statistics

To clear the LACP statistics, perform this task in privileged mode:

Task	Command
Clear LACP statistics.	**clear lacp-channel statistics**

This example shows how to clear LACP statistics:

```
Console> (enable) clear lacp-channel statistics
LACP channel counters are cleared.
Console> (enable)
```

Displaying EtherChannel Traffic Utilization

To display the traffic utilization on the EtherChannel ports, perform this task:

Task	Command
Display traffic utilization on the EtherChannel ports.	**show lacp-channel traffic**

This example shows how to display traffic utilization on the EtherChannel ports:

```
Console> (enable) show lacp-channel traffic
ChanId Port  Rx-Ucst Tx-Ucst Rx-Mcst Tx-Mcst Rx-Bcst Tx-Bcst
------ ----- ------- ------- ------- ------- ------- -------
   808  2/16   0.00%   0.00%  50.00%  75.75%   0.00%   0.00%
   808  2/17   0.00%   0.00%  50.00%  25.25%   0.00%   0.00%
   816  2/31   0.00%   0.00%  25.25%  50.50%   0.00%   0.00%
   816  2/32   0.00%   0.00%  75.75%  50.50%   0.00%   0.00%
Console> (enable)
```

Displaying the Outgoing Ports for a Specified Address or Layer 4 Port Number

To display the outgoing port that is used in an EtherChannel for a specified address or Layer 4 port number, perform this task:

Task	Command
Display the outgoing port for a specified address or Layer 4 port number.	**show lacp-channel hash** *channel_id src_ip_addr* [*dest_ip_addr*] | *dest_ip_address* | *src_mac_addr* [*dest_mac_addr*] | *dest_mac_addr* | *src_port dest_port* | *dest_port*

This example shows how to display the outgoing port for the specified source and destination IP addresses:

```
Console> (enable) show lacp-channel hash 808 172.20.32.10 172.20.32.66
Selected channel port:2/17
Console> (enable)
```

Disabling an EtherChannel

To disable an EtherChannel, perform this task in privileged mode:

Task	Command
Disable an EtherChannel.	**set port lacp-channel** *mod/port* **mode off**

This example shows how to disable an EtherChannel:

```
Console> (enable) set port lacp-channel 2/2-8 mode off
Port(s) 2/2-8 channel mode set to off.
Console> (enable)
```

Displaying the Spanning-Tree Information for EtherChannels

You can display the channel ID and the truncated port list for all ports that are channeling. The ports that are not channeling are identified by their port number.

To display the spanning-tree information for EtherChannels, perform this task:

Task	Command
Display the spanning-tree information for EtherChannels.	**show spantree** *mod/port*

These examples show how to display the spanning-tree information for EtherChannels:

```
Console> show spantree 4/6
Port                     Vlan Port-State    Cost  Priority Portfast   Channel_id
------------------------ ---- ------------- ----- -------- ---------- ----------
 4/6                     1    not-connected     4       32 disabled   0
Console>

Console> show spantree 4/7
Port                     Vlan Port-State    Cost  Priority Portfast   Channel_id
------------------------ ---- ------------- ----- -------- ---------- ----------
4/7-8                    1    blocking          3       32 disabled   770
Console>
```

Clearing and Restoring the EtherChannel Counters

The **show channel traffic** command allows you to display the channel traffic utilization. The channel traffic utilization shows the percentage of traffic that passes through each channel port. The counters are maintained for different types of packets. Before software release 8.3(1), you could not clear the channel hardware counter bases because the bases are MIB objects that do not clear. Enter the **clear counters all** command to reset the channel counter bases. With software release 8.3(1) and later releases, you can clear and restore the channel-based counters on a per-protocol and per-channel basis. To clear or restore the channel-based counters on a per-channel basis, enter the channel ID. To find the channel ID, enter the **show port channel** command for the PAgP channels or the **show port lacp-channel** command for the LACP channels.

Clearing the EtherChannel Counters

To clear the EtherChannel counters, perform these tasks in privileged mode:

Task	Command
Clear all PAgP channel counters.	**clear counter channel all**
Clear a specific PAgP channel counter.	**clear counter channel** *channel_id*
Clear all LACP channel counters.	**clear counter lacp-channel all**
Clear a specific LACP channel counter.	**clear counter lacp-channel** *channel_id*

These examples show the various methods of clearing the EtherChannel counters:

```
Console> (enable) show channel traffic
ChanId Port  Rx-Ucst Tx-Ucst Rx-Mcst Tx-Mcst Rx-Bcst Tx-Bcst
------ ----- ------- ------- ------- ------- ------- -------
   769  1/1    0.00%   0.00%   9.09%  90.90%   0.00%   0.00%
   769  2/1    0.00%   0.00%  90.91%   9.10%   0.00%   0.00%
------ ----- ------- ------- ------- ------- ------- -------
   841  7/17   0.00%   0.00% 100.00% 100.00%   0.00%   0.00%
   841  7/18   0.00%   0.00%   0.00%   0.00%   0.00%   0.00%
Console> (enable) clear counter channel all
This command will reset MAC and port counters reported by the CLI for all ports.
Counters reported by SNMP will not be affected.
Do you want to continue (y/n) [n]? y
MAC and Port counters cleared.
Console> (enable) show channel traffic
ChanId Port  Rx-Ucst Tx-Ucst Rx-Mcst Tx-Mcst Rx-Bcst Tx-Bcst
------ ----- ------- ------- ------- ------- ------- -------
   769  1/1    0.00%   0.00%   0.00% 100.00%   0.00%   0.00%
   769  2/1    0.00%   0.00% 100.00%   0.00%   0.00%   0.00%
------ ----- ------- ------- ------- ------- ------- -------
   841  7/17   0.00%   0.00% 100.00% 100.00%   0.00%   0.00%
   841  7/18   0.00%   0.00%   0.00%   0.00%   0.00%   0.00%
Console> (enable) show channel traffic 769
ChanId Port  Rx-Ucst Tx-Ucst Rx-Mcst Tx-Mcst Rx-Bcst Tx-Bcst
------ ----- ------- ------- ------- ------- ------- -------
   769  1/1    0.00%   0.00%   9.52%  90.47%   0.00%   0.00%
   769  2/1    0.00%   0.00%  90.48%   9.53%   0.00%   0.00%
Console> (enable) clear counter channel 769
This command will reset MAC and port counters reported by the CLI for PAGP channel 769
Counters reported by SNMP will not be affected.
Do you want to continue (y/n) [n]? y
MAC and Port counters cleared.
Console> (enable) show channel traffic 769
ChanId Port  Rx-Ucst Tx-Ucst Rx-Mcst Tx-Mcst Rx-Bcst Tx-Bcst
------ ----- ------- ------- ------- ------- ------- -------
   769  1/1    0.00%   0.00%   0.00% 100.00%   0.00%   0.00%
   769  2/1    0.00%   0.00% 100.00%   0.00%   0.00%   0.00%
Console> (enable)
```

Restoring the EtherChannel Counters

To restore the EtherChannel counters, perform these tasks in privileged mode:

Task	Command
Restore all PAgP channel counters.	**restore counter channel all**
Restore a specific PAgP channel counter.	**restore counter channel** *channel_id*
Restore all LACP channel counters.	**restore counter lacp-channel all**
Restore a specific LACP channel counter.	**restore counter lacp-channel** *channel_id*

This example shows how to restore the counters for channel 769:

```
Console> (enable) restore counter channel 769
This command will restore counter values reported by the CLI
for PAGP channel 769 ports to the hardware counter values.
Do you want to continue (y/n) [n]? y
MAC and Port counters restored.
```

```
Console> (enable) show channel traffic 769
ChanId Port  Rx-Ucst Tx-Ucst Rx-Mcst Tx-Mcst Rx-Bcst Tx-Bcst
------ ----- ------- ------- ------- ------- ------- -------
   769  1/1    0.00%   0.00%   7.69%  92.30%   0.00%   0.00%
   769  2/1    0.00%   0.00%  92.31%   7.70%   0.00%   0.00%
Console> (enable)
```

CHAPTER 7

Configuring Spanning Tree

This chapter describes the IEEE 802.1D bridge Spanning Tree Protocol (STP) and how to use and configure Cisco's proprietary spanning-tree protocols, Per VLAN Spanning Tree + (PVST+) and Multi-Instance Spanning Tree Protocol (MISTP), on the Catalyst 6500 series switches.

Note For information on configuring the spanning-tree PortFast, UplinkFast, and BackboneFast features, see Chapter 9, "Configuring Spanning-Tree PortFast, UplinkFast, BackboneFast, and Loop Guard."

This chapter consists of these sections:

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

Understanding How Spanning Tree Protocols Work

This section describes the specific functions that are common to all spanning-tree protocols. Cisco's proprietary spanning-tree protocols, PVST+ and MISTP, are based on IEEE 802.1D STP. (See the "Understanding How PVST+ and MISTP Modes Work" section on page 7-12 for information about PVST+ and MISTP.) The 802.1D STP is a Layer 2 management protocol that provides path redundancy in a network while preventing undesirable loops. All spanning-tree protocols use an algorithm that calculates the best loop-free path through the network.

STP uses a distributed algorithm that selects one bridge of a redundantly connected network as the root of a spanning tree-connected active topology. STP assigns roles to each port depending on what the port's function is in the active topology. Port roles are as follows:

- Root—A forwarding port that is elected for the spanning-tree topology
- Designated—A forwarding port that is elected for every switched LAN segment
- Alternate—A blocked port providing an alternate path to the root port in the spanning tree
- Backup—A blocked port in a loopback configuration

The switches that have ports with these assigned roles are called the root or designated switches. For more information, see the "Understanding How a Switch Becomes the Root Switch" section on page 7-3.

In Ethernet networks, only one active path may exist between any two stations. Multiple active paths between stations can cause loops in the network. When loops occur, some switches recognize the stations on both sides of the switch. This situation causes the forwarding algorithm to malfunction allowing the duplicate frames to be forwarded.

The spanning-tree algorithms provide path redundancy by defining a tree that spans all of the switches in an extended network and then forces certain redundant data paths into a standby (blocked) state. At regular intervals, the switches in the network send and receive spanning-tree packets that they use to identify the path. If one network segment becomes unreachable, or if the spanning-tree costs change, the spanning-tree algorithm reconfigures the spanning-tree topology and reestablishes the link by activating the standby path.

The spanning-tree operation is transparent to end stations, which do not detect whether they are connected to a single LAN segment or a switched LAN of multiple segments.

These sections describe the STP:

Understanding How a Topology is Created

All switches in an extended LAN participating in a spanning tree gather information about other switches in the network through an exchange of data messages that are known as bridge protocol data units (BPDUs). This exchange of messages results in the following actions:

- A unique root switch is elected for the spanning-tree network topology.
- A designated switch is elected for every switched LAN segment.
- Any loops in the switched network are eliminated by placing redundant switch ports in a backup state; all paths that are not needed to reach the root switch from anywhere in the switched network are placed in STP-blocked mode.

The topology of an active switched network is determined by the following:

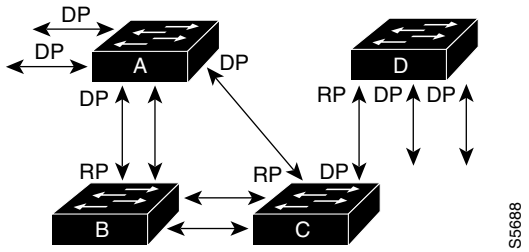
- The unique switch identifier Media Access Control ([MAC] address of the switch) that is associated with each switch
- The path cost to the root that is associated with each switch port
- The port identifier (MAC address of the port) that is associated with each switch port

In a switched network, the root switch is the logical center of the spanning-tree topology. A spanning-tree protocol uses BPDUs to elect the root switch and root port for the switched network, as well as the root port and designated port for each switched segment.

Understanding How a Switch Becomes the Root Switch

If all switches are enabled with default settings, the switch with the lowest MAC address in the network becomes the root switch. In Figure 7-1, Switch A is the root switch because it has the lowest MAC address. However, due to traffic patterns, number of forwarding ports, or line types, Switch A might not be the ideal root switch. A switch can be forced to become the root switch by increasing the priority (that is, lowering the numerical priority number) on the preferred switch. This action causes the spanning tree to recalculate the topology and make the selected switch the root switch.

Figure 7-1 Configuring a Loop-Free Topology



RP = Root Port
DP = Designated Port

You can change the priority of a port to make it the root port. When the spanning-tree topology is based on default parameters, the path between the source and destination stations in a switched network might not be ideal. Connecting higher-speed links to a port that has a higher number than the current root port can cause a root-port change. The goal is to make the fastest link the root port.

For example, assume that a port on Switch B is a fiber-optic link. Also, another port on Switch B (an unshielded twisted-pair [UTP] link) is the root port. Network traffic might be more efficient over the high-speed fiber-optic link. By changing the Port Priority parameter for the fiber-optic port to a higher priority (lower numerical value) than the UTP port, the fiber-optic port becomes the root port. You could also accomplish this scenario by changing the Port Cost parameter for the fiber-optic port to a lower value than that of the UTP port.

Understanding How Bridge Protocol Data Units Work

The BPDUs contain configuration information about the transmitting switch and its ports, including the switch and port MAC addresses, switch priority, port priority, and port cost. Each configuration BPDU contains this information:

- The unique identifier of the switch that the transmitting switch believes to be the root switch
- The cost of the path to the root from the transmitting port
- The identifier of the transmitting port

The switch sends configuration BPDUs to communicate with and compute the spanning-tree topology. A MAC frame conveying a BPDU sends the switch group address to the destination address field. All switches that are connected to the LAN on which the frame is transmitted receive the BPDU. The BPDUs are not directly forwarded by the switch, but the receiving switch uses the information in the frame to calculate a BPDU and, if the topology changes, initiates a BPDU transmission.

A BPDU exchange results in the following:

- One switch is elected as the root switch.
- The shortest distance to the root switch is calculated for each switch.
- A designated switch is selected. This is the switch that is closest to the root switch through which frames will be forwarded to the root.
- A port for each switch is selected. This is the port that provides the best path from the switch to the root switch.
- The ports included in the STP are selected.

Calculating and Assigning Port Costs

By calculating and assigning the port cost of the switch ports, you can ensure that the shortest (lowest cost) distance to the root switch is used to transmit data. You can calculate and assign the lower path cost values (port costs) to the higher bandwidth ports by using either the short method (which is the default) or the long method. The short method uses a 16-bit format that yields values from 1 to 65535. The long method uses a 32-bit format that yields values in the range of 1 to 200,000,000. For more information on setting the default cost mode, see the "Configuring the PVST+ Default Port Cost Mode" section on page 7-29.

Note You should configure all switches in your network to use the same method for calculating the port cost. The short method is used to calculate the port cost unless you specify that the long method be used. You can specify the calculation method using the CLI.

Calculating the Port Cost Using the Short Method

The IEEE 802.1D specification assigns 16-bit (short) default port cost values to each port that is based on bandwidth. You can also manually assign port costs between 1–65535. The 16-bit values are only used for the ports that have not been specifically configured for port cost. Table 7-1 shows the default port cost values that are assigned by the switch for each type of port when you use the short method to calculate the port cost.

Table 7-1 Default Port Cost Values Using the Short Method

Port Speed	Default Cost Value	Default Range
10 Mbps	100	1 to 65535
100 Mbps	19	1 to 65535
1 Gbps	4	1 to 65535

Calculating the Port Cost Using the Long Method

802.1t assigns 32-bit (long) default port cost values to each port using a formula that is based on the bandwidth of the port. You can also manually assign port costs between 1–200,000,000. The formula for obtaining default 32-bit port costs is to divide the bandwidth of the port by 200,000,000. Table 7-2 shows the default port cost values that are assigned by the switch and the recommended cost values and ranges for each type of port when you use the long method to calculate port cost.

Table 7-2 Default Port Cost Values Using the Long Method

Port Speed	Recommended Value	Recommended Range	Available Range
≤100 kbps	200000000	20000000 to 200000000	1 to 200000000
1 Mbps	20000000	2000000 to 200000000	1 to 200000000
10 Mbps	2000000	200000 to 20000000	1 to 200000000
100 Mbps	200000	20000 to 2000000	1 to 200000000
1 Gbps	20000	2000 to 200000	1 to 200000000
10 Gbps	2000	200 to 20000	1 to 200000000

Calculating the Port Cost for Aggregate Links

As individual links are added or removed from an aggregate link (port bundle), the bandwidth of the aggregate link increases or decreases. These changes in bandwidth lead to recalculation of the default port cost for the aggregated port. Changes to the default port cost or changes resulting from links that autonegotiate their bandwidth could lead to recalculation of the spanning-tree topology which may not be desirable, especially if the added or removed link is of little consequence to the bandwidth of the aggregate link (for example, if a 10-Mbps link is removed from a 10-Gbps aggregate link). Because of the limitations that are presented by automatically recalculating the topology, 802.1t states that changes in bandwidth will not result in changes to the cost of the port. The aggregated port will use the same port cost parameters as a standalone port.

Spanning-Tree Port States

Topology changes can take place in a switched network due to a link coming up or a link going down (failing). When a switch port transitions directly from nonparticipation in the topology to the forwarding state, it can create temporary data loops. Ports must wait for new topology information to propagate through the switches in the LAN before they can start forwarding frames. Also, they must allow the frame lifetime to expire for frames that have been forwarded using the old topology.

Note With Cisco IOS Release 12.1.(1)E or later releases on the Multilayer Switch Feature Card (MSFC), the Address Resolution Protocol (ARP) on the STP Topology Change Notification feature ensures that excessive flooding does not occur when the MSFC receives a topology change notification (TCN) from the supervisor engine. The feature causes the MSFC to send ARP requests for all the ARP entries belonging to the VLAN interface where the TCN is received. When the ARP replies come back, the Policy Feature Card (PFC) learns the MAC entries, which were lost as a result of the topology change. Learning the entries immediately following a topology change prevents excessive flooding later. There is no configuration required on the MSFC. This feature works with supervisor engine software release 5.4(2) or later releases.

At any given time, each port on a switch using a spanning-tree protocol is in one of these states:

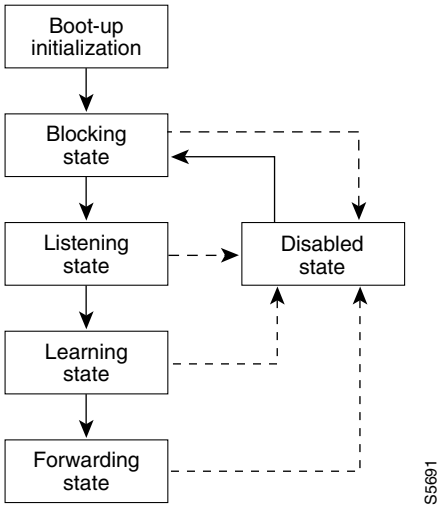
- Blocking
- Listening
- Learning
- Forwarding
- Disabled

A port moves through these states as follows:

- From initialization to blocking
- From blocking to listening or to disabled
- From listening to learning or to disabled
- From learning to forwarding or to disabled
- From forwarding to disabled

Figure 7-2 illustrates how a port moves through the states.

Figure 7-2 STP Port States



You can modify each port state by using management software, for example, VLAN Trunking Protocol (VTP). When you enable spanning tree, every switch in the network goes through the blocking state and the transitory states of listening and learning at power up. If properly configured, each port stabilizes into the forwarding or blocking state.

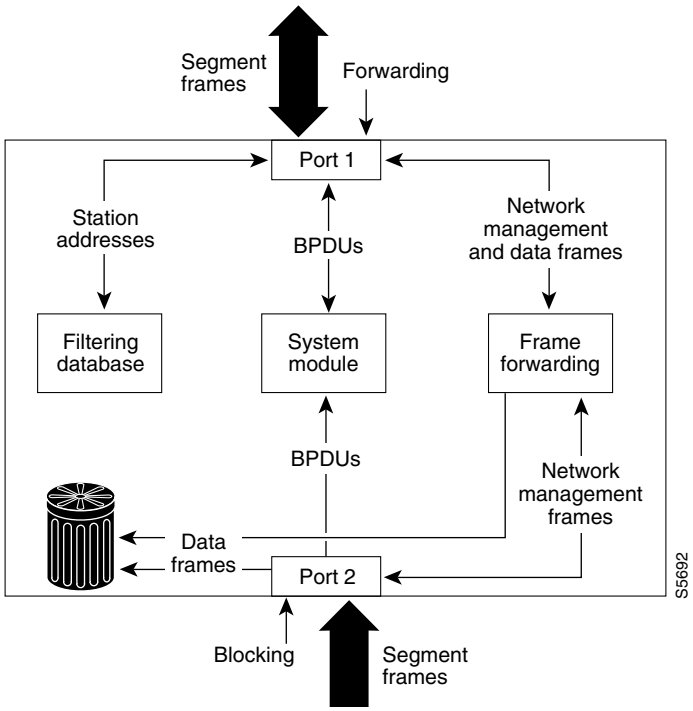
When the spanning-tree algorithm places a port in the forwarding state, the following occurs:

- The port is put into the listening state while it waits for protocol information that suggests it should go to the blocking state.
- The port waits for the expiration of a protocol timer that moves the port to the learning state.
- In the learning state, the port continues to block frame forwarding as it learns station location information for the forwarding database.
- The expiration of a protocol timer moves the port to the forwarding state, where both learning and forwarding are enabled.

Blocking State

A port in the blocking state does not participate in frame forwarding (see Figure 7-3). After initialization, a BPDU is sent to each port in the switch. A switch initially assumes that it is the root until it exchanges BPDUs with other switches. This exchange establishes which switch in the network is really the root. If only one switch resides in the network, no exchange occurs, the forward delay timer expires, and the ports move to the listening state. A switch always enters the blocking state following switch initialization.

Figure 7-3 Port 2 in Blocking State



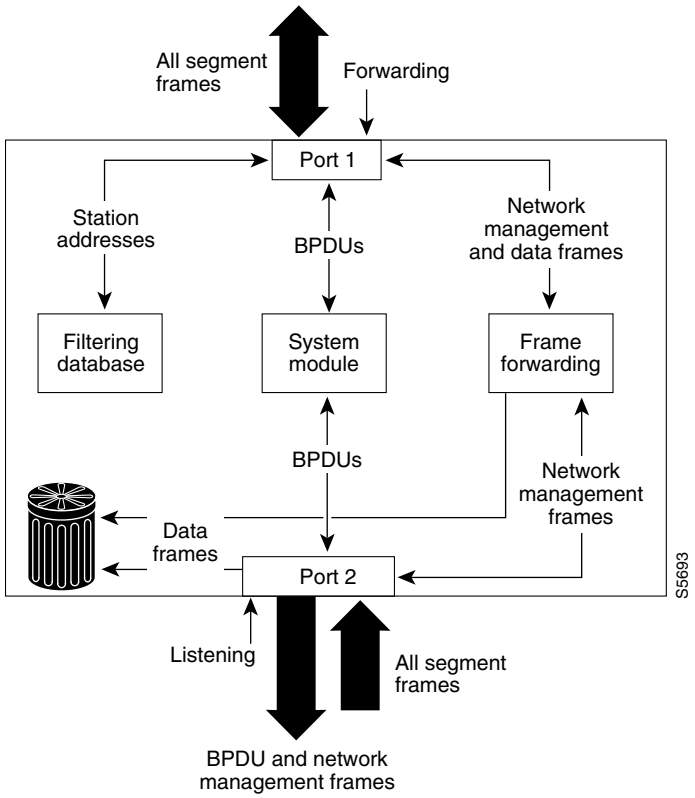
A port in the blocking state performs as follows:

- Discards frames that are received from the attached segment.
- Discards frames that are switched from another port for forwarding.
- Does not incorporate station location into its address database. (There is no learning on a blocking port, so there is no address database update.)
- Receives BPDUs and directs them to the system module.
- Does not transmit BPDUs that are received from the system module.
- Receives and responds to network management messages.

Listening State

The listening state is the first transitional state that a port enters after the blocking state. The port enters this state when the spanning tree determines that the port should participate in frame forwarding. Learning is disabled in the listening state. Figure 7-4 shows a port in the listening state.

Figure 7-4 Port 2 in Listening State



A port in the listening state performs as follows:

- Discards frames that are received from the attached segment.
- Discards frames that are switched from another port for forwarding.
- Does not incorporate station location into its address database. (There is no learning at this point, so there is no address database update.)
- Receives BPDUs and directs them to the system module.
- Processes BPDUs that are received from the system module.
- Receives and responds to network management messages.

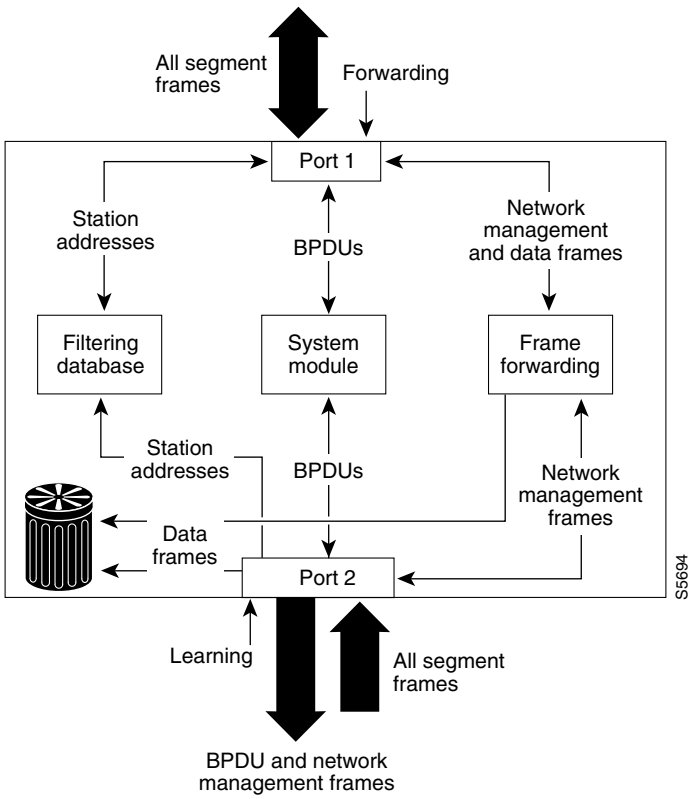
Learning State

A port in the learning state prepares to participate in frame forwarding. The port enters the learning state from the listening state. Figure 7-5 shows a port in the learning state.

A port in the learning state performs as follows:

- Discards frames that are received from the attached segment.
- Discards frames that are switched from another port for forwarding.
- Incorporates station location into its address database.
- Receives BPDUs and directs them to the system module.
- Receives, processes, and transmits BPDUs that are received from the system module.
- Receives and responds to network management messages.

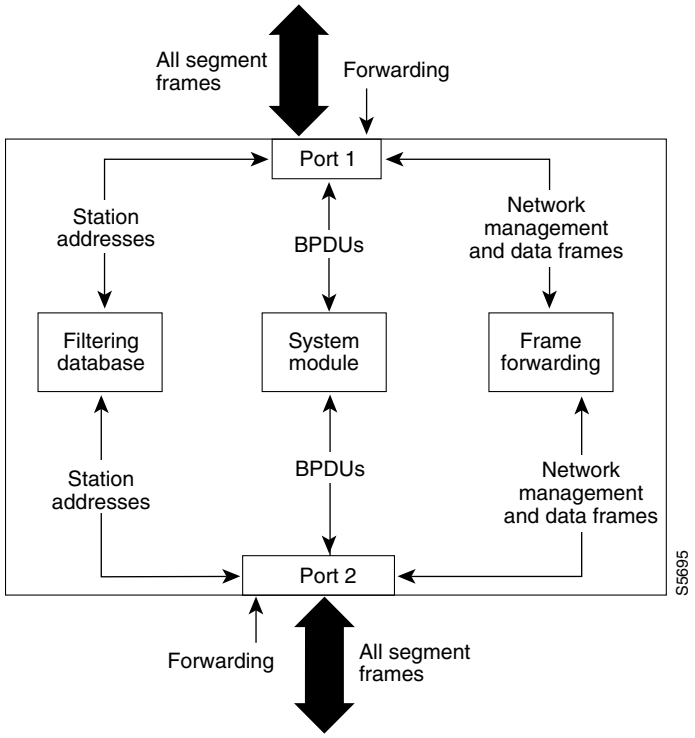
Figure 7-5 Port 2 in Learning State



Forwarding State

A port in the forwarding state forwards frames, as shown in Figure 7-6. The port enters the forwarding state from the learning state.

Figure 7-6 Port 2 in Forwarding State



A port in the forwarding state performs as follows:

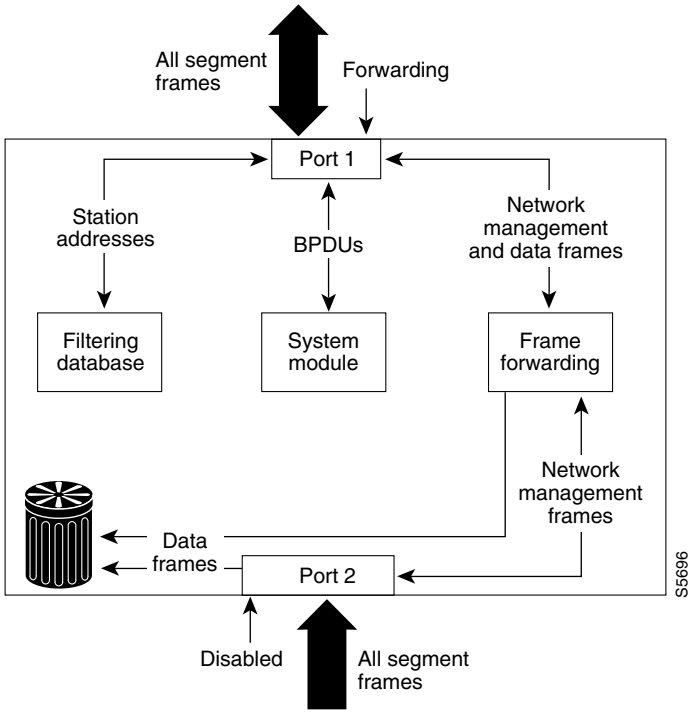
- Forwards frames that are received from the attached segment.
- Forwards frames that are switched from another port for forwarding.
- Incorporates station location information into its address database.
- Receives BPDUs and directs them to the system module.
- Processes BPDUs that are received from the system module.
- Receives and responds to network management messages.

Caution Use spanning-tree PortFast mode only on ports that are directly connected to individual workstations to allow these ports to come up and go directly to the forwarding state, instead of having to go through the entire spanning-tree initialization process. To prevent illegal topologies, enable spanning tree on ports that are connected to switches or other devices that forward messages. For more information about PortFast, see Chapter 9, "Configuring Spanning-Tree PortFast, UplinkFast, BackboneFast, and Loop Guard."

Disabled State

A port in the disabled state does not participate in frame forwarding or STP, as shown in Figure 7-7. A port in the disabled state is virtually nonoperational.

Figure 7-7 Port 2 in Disabled State



A disabled port performs as follows:

- Discards frames that are received from the attached segment.
- Discards frames that are switched from another port for forwarding.
- Does not incorporate station location into its address database. (There is no learning, so there is no address database update.)
- Receives BPDUs but does not direct them to the system module.
- Does not receive BPDUs for transmission from the system module.
- Receives and responds to network management messages.

Understanding How PVST+ and MISTP Modes Work

Catalyst 6500 series switches provide two proprietary spanning-tree modes that are based on the IEEE 802.1D standard and one mode that is a combination of the two modes:

- Per VLAN Spanning Tree (PVST+)
- Rapid-PVST+
- Multi-Instance Spanning Tree Protocol (MISTP)
- MISTP-PVST+ (combination mode)

An overview of each mode is provided in this section. Each mode is described in detail in these sections:

- Configuring PVST+ on the Switch, page 7-26
- Configuring MISTP-PVST+ or MISTP on the Switch, page 7-34

Caution If your network currently uses PVST+ and you plan to use MISTP on any switch, you must first enable MISTP-PVST+ on the switch and configure an MISTP instance to avoid causing loops in the network.

PVST+ Mode

PVST+ runs on each VLAN on the switch, ensuring that each VLAN has a loop-free path through the network.

PVST+ provides Layer 2 load balancing for the VLAN on which it runs; you can create different logical topologies using the VLANs on your network to ensure that all the links are used and no link is oversubscribed.

Each PVST+ instance on a VLAN has a single root switch. This root switch propagates the spanning-tree information that is associated with that VLAN to all other switches in the network. This process ensures that the network topology is maintained because each switch has the same knowledge about the network.

Rapid-PVST+

With software release 8.1(1) and later releases, Rapid-PVST+ is the default spanning-tree protocol that is used on all Ethernet, Fast Ethernet, and Gigabit Ethernet port-based VLANs on Catalyst 6500 series switches. Rapid-PVST+ is similar to PVST+. The only difference is that Rapid-PVST+ uses a rapid STP that is based on IEEE 802.1w instead of 802.1D. Rapid-PVST+ uses the same configuration as PVST+ with minimal additional configuration. See the "Configuring Rapid-PVST+ on the Switch" section on page 7-33 for configuration information. In Rapid-PVST+, dynamic CAM entries are flushed immediately on a per-port basis when any topology change is made. UplinkFast and BackboneFast are enabled, but not active in this mode, as the functionality is built into the rapid STP. Rapid-PVST+ provides for rapid recovery of connectivity following the failure of a bridge, bridge port, or LAN.

A port that is connected to a nonbridging device (for example, a host or a router) is an edge port. A port that connects to a hub is also an edge port if the hub or any LAN that is connected by it does not have a bridge. An edge port can start forwarding as soon as the link is up. You must explicitly configure the ports that are connected to the hosts and routers as edge ports while using Rapid-PVST+.

For complete protocol details, see the "Rapid Spanning Tree Protocol" section on page 7-18.

MISTP Mode

MISTP is an optional spanning-tree protocol that runs on Catalyst 6500 series switches. MISTP allows you to group multiple VLANs under a single instance of spanning tree (an MISTP instance). MISTP combines the Layer 2 load-balancing benefits of PVST+ with the lower CPU load of IEEE 802.1Q.

An MISTP instance is a virtual logical topology that is defined by a set of bridge and port parameters. When you map VLANs to an MISTP instance, this virtual logical topology becomes a physical topology. Each MISTP instance has its own root switch and a different set of forwarding links (different bridge and port parameters).

Each MISTP instance root switch propagates the information that is associated with it to all other switches in the network. This process maintains the network topology because it ensures that each switch has the same information about the network.

MISTP builds MISTP instances by exchanging MISTP BPDUs with peer entities in the network. MISTP uses one BPDU for each MISTP instance, rather than one for each VLAN, as in PVST+. Because there are fewer BPDUs in an MISTP network, MISTP networks converge faster with less overhead. MISTP discards PVST+ BPDUs.

An MISTP instance can have any number of VLANs that are mapped to it, but a VLAN can be mapped only to a single MISTP instance. You can easily move a VLAN (or VLANs) in an MISTP topology to another MISTP instance if it has converged. (However, if ports are added at the same time that the VLAN is moved, convergence time is required.)

MISTP-PVST+ Mode

MISTP-PVST+ is a transition spanning-tree mode that allows you to use the MISTP functionality on Catalyst 6500 series switches while continuing to communicate with the Catalyst 5000 and 6500 series switches in your network that use PVST+. A switch using PVST+ mode that is connected to a switch using MISTP mode cannot see the BPDUs of the other switch, which is a condition that can cause loops in the network. MISTP-PVST+ allows interoperability between PVST+ and pure MISTP because it sees the BPDUs of both modes. To convert your network to MISTP, use MISTP-PVST+ to transition the network from PVST+ to MISTP.

Because MISTP-PVST+ conforms to the limits of PVST+, you cannot configure more VLAN ports on your MISTP-PVST+ switches than on your PVST+ switches.

Understanding How Bridge Identifiers Work

These sections explain how MAC addresses are used in PVST+ and MISTP as unique bridge identifiers:

- MAC Address Allocation, page 7-14
- MAC Address Reduction, page 7-15

MAC Address Allocation

Catalyst 6500 series switches have a pool of 1024 MAC addresses that can be used as bridge identifiers for VLANs running under PVST+ or for MISTP instances. You can use the **show module** command to view the MAC address range.

MAC addresses are allocated sequentially, with the first MAC address in the range assigned to VLAN 1, the second MAC address in the range assigned to VLAN 2, and so on. The last MAC address in the range is assigned to the supervisor engine in-band (sc0) management interface.

For example, if the MAC address range is 00-e0-1e-9b-2e-00 to 00-e0-1e-9b-31-ff, the VLAN 1 bridge ID is 00-e0-1e-9b-2e-00, the VLAN 2 bridge ID is 00-e0-1e-9b-2e-01, the VLAN 3 bridge ID is 00-e0-1e-9b-2e-02, and so forth. The in-band (sc0) interface MAC address is 00-e0-1e-9b-31-ff.

MAC Address Reduction

For Catalyst 6500 series switches that support 4096 VLANs, MAC address reduction allows up to 4096 VLANs running under PVST+ or 16 MISTP instances to have unique identifiers without increasing the number of MAC addresses that are required on the switch. MAC address reduction reduces the number of MAC addresses that are required by the STP from one per VLAN or MISTP instance to one per switch. However, because VLANs running under PVST+ and MISTP instances running under MISTP-PVST+ or MISTP are considered logical bridges, each bridge must have its own unique identifier in the network.

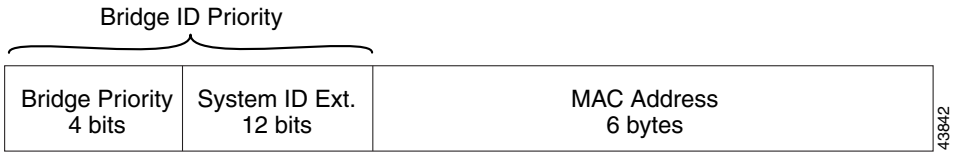
When you enable MAC address reduction, the bridge identifier that is stored in the spanning-tree BPDU contains an additional field called the *system ID extension*. Combined with the bridge priority, the system ID extension functions as the unique identifier for a VLAN or an MISTP instance. The system ID extension is always the number of the VLAN or the MISTP instance; for example, the system ID extension for VLAN 100 is 100, and the system ID extension for MISTP instance 2 is 2.

Figure 7-8 shows the bridge identifier when you do not enable MAC address reduction. The bridge identifier consists of the bridge priority and the MAC address.

Figure 7-8 Bridge Identifier without MAC Address Reduction

Figure 7-9 shows the bridge identifier when you enable MAC address reduction. The bridge identifier consists of the bridge priority, the system ID extension, and the MAC address. The bridge priority and the system ID extension combined are known as the *bridge ID priority.* The bridge ID priority is the unique identifier for the VLAN or the MISTP instance.

Figure 7-9 Bridge Identifier with MAC Address Reduction Enabled



When you enter the **show spantree** command, you can see the bridge ID priority for a VLAN in PVST+ or for an MISTP instance in MISTP or MISTP-PVST+ mode.

This example shows the bridge ID priority for VLAN 1 when you enable MAC address reduction in PVST+ mode. The unique identifier for this VLAN is 32769.

```
Console> (enable) show spantree 1
VLAN 1
Spanning tree mode          PVST+
Spanning tree type          ieee
  .
  .
  .
Bridge ID MAC ADDR          00-d0-00-4c-18-00
Bridge ID Priority          32769  (bridge priority: 32768, sys ID ext: 1)
Bridge Max Age 20 sec   Hello Time 2  sec   Forward Delay 15 sec
```

If you have a Catalyst switch in your network with MAC address reduction enabled, you should also enable MAC address reduction on all other Layer-2 connected switches to avoid undesirable root election and spanning-tree topology issues.

When MAC address reduction is enabled, the root bridge priority becomes a multiple of 4096 plus the VLAN ID. With MAC address reduction enabled, a switch bridge ID (used by the spanning-tree algorithm to determine the identity of the root bridge, the lowest being preferred) can only be specified as a multiple of 4096. Only the following values are possible: 0, 4096, 8192, 12288, 16384, 20480, 24576, 28672, 32768, 36864, 40960, 45056, 49152, 53248, 57344, and 61440.

If another bridge in the same spanning-tree domain does not run the MAC address reduction feature, it could claim and win root bridge ownership because of the finer granularity in the selection of its bridge ID.

Note MAC address reduction is enabled by default on Cisco switches that have 64 MAC addresses (to find the number of MAC addresses supported on a switch, refer to the *Catalyst 6500 Series Switch Release Notes for Software Release 8.x* publication).

Understanding How Multiple Spanning Tree Works

The Multiple Spanning Tree (MST) feature is the IEEE 802.1s and is an amendment to 802.1Q. MST extends the 802.1w Rapid Spanning Tree (RST) algorithm to multiple spanning trees. This extension provides for both rapid convergence and load balancing in a VLAN environment. In software release 8.3(1), the MST protocol is compliant with IEEE 802.1s and is backward compatible with 802.1D STP, 802.1w, the Rapid Spanning Tree Protocol (RSTP), and the Cisco PVST+ architecture that was implemented in previous software releases. The MST protocol in software release 8.3(1) will interoperate with MST in earlier software releases.

MST allows you to build multiple spanning trees over VLAN trunks. You can group and associate VLANs to spanning-tree instances. Each instance can have a topology that is independent of other spanning-tree instances, and each instance can have a different port instance cost and port instance priority. This architecture provides multiple forwarding paths for data traffic and enables load balancing. Network fault tolerance is improved because a failure in one instance (forwarding path) does not affect other instances (forwarding paths).

In large networks, having different VLAN spanning-tree instance assignments that are located in different parts of the network makes it easier to administrate and optimally utilize redundant paths. However, a spanning-tree instance can exist only on bridges that have compatible VLAN instance assignments. MST requires that you configure a set of bridges with the same MST configuration information, which allows them to participate in a given set of spanning-tree instances. Interconnected bridges that have the same MST configuration are referred to as an MST region.

MST uses the modified RSTP version called the Multiple Spanning Tree Protocol (MSTP). The MST feature has these characteristics:

- MST runs a variant of spanning tree that is called Internal Spanning Tree (IST). IST augments the Common Spanning Tree (CST) information with internal information about the MST region. The MST region appears as a single bridge to adjacent Single Spanning Tree (SST) and MST regions.

- A bridge running MST provides interoperability with single spanning-tree bridges as follows:
  - MST bridges run a variant of STP (IST) that augments the Common Spanning Tree (CST) information with internal information about the MST region.
  - IST connects all the MST bridges in the region and appears as a subtree in the CST that encompasses the whole bridged domain. The MST region appears as a virtual bridge to adjacent SST bridges and MST regions.
  - The collection of ISTs in each MST region, the CST that interconnects the MST regions, and the SST bridges define Common and Internal Spanning Tree (CIST). CIST is the same as an IST inside an MST region and the same as a CST outside an MST region. The STP, RSTP, and MSTP together elect a single bridge as the root of CIST.
- MST establishes and maintains additional spanning trees within each MST region. These spanning trees are referred to as MST instances (MSTIs). The IST is numbered 0, and the MSTIs are numbered 1,2,3,... and so on. Any given MSTI is local to the MST region that is independent of MSTIs in another region, even if the MST regions are interconnected. MST instances combine with the IST at the boundary of MST regions to become the CST as follows:
  - Spanning-tree information for an MSTI is contained in an MSTP record (M-record).

    M-records are always encapsulated within MST BPDUs (MST BPDUs). The original spanning trees that are computed by MSTP are called M-trees. M-trees are active only within the MST region. M-trees merge with the IST at the boundary of the MST region and form the CST.
- MST provides interoperability with PVST+ by generating PVST+ BPDUs for the non-CST VLANs.
- MST supports some of the PVST+ extensions in MSTP as follows:
  - UplinkFast and BackboneFast are not available in MST mode; they are part of RSTP.
  - PortFast is supported.
  - BPDU filtering and BPDU guard are supported in MST mode.
  - Loop guard and root guard are supported in MST. MST preserves the VLAN 1 disabled functionality except that BPDUs are still transmitted in VLAN 1.
  - MST switches behave as if MAC reduction is enabled.
  - For private VLANs (PVLANs), secondary VLANs are mapped to the same instance as the primary.

Follow these guidelines when using MST:

- Do not disable spanning tree on any VLAN in any of the PVST bridges.
- Ensure that all PVST spanning-tree root bridges have lower (numerically higher) priority than the CST root bridge.
- Do not use PVST bridges as the root of CST.
- Ensure that trunks carry all of the VLANs that are mapped to an instance or do not carry any VLANs at all.
- Do not connect switches with access links because access links may partition a VLAN.
- You should perform any MST configuration involving a large number of either existing or new logical VLAN ports during the maintenance window because the complete MST database gets reinitialized for any incremental changes (such as adding new VLANs to instances or moving VLANs across instances).

These sections describe MST:

Rapid Spanning Tree Protocol

RSTP significantly reduces the time to reconfigure the active topology of the network when changes to the physical topology or its configuration parameters occur. RSTP selects one switch as the root of a spanning-tree-connected active topology and assigns port roles to individual ports of the switch, depending on whether that port is part of the active topology.

RSTP provides rapid connectivity following the failure of a switch, switch port, or a LAN. A new root port and the designated port on the other side of the bridge transition to forwarding through an explicit handshake between them. RSTP allows switch port configuration so that the ports can transition to forwarding directly when the switch reinitializes.

RSTP, specified in 802.1w, supersedes STP, which is specified in 802.1D, while remaining compatible with STP. RSTP provides the structure on which the MST operates. You configure RSTP when you configure the MST feature. For more information, see the "Configuring Multiple Spanning Tree on the Switch" section on page 7-51.

RSTP provides backward compatibility with 802.1D bridges as follows:

- RSTP selectively sends 802.1D-configured BPDUs and Topology Change Notification (TCN) BPDUs on a per-port basis.
- When a port initializes, the Migration Delay timer starts and RSTP BPDUs are transmitted. While the Migration Delay timer is active, the bridge processes all BPDUs that are received on that port. RSTP BPDUs are not visible on the port; only version 3 BPDUs are visible.
- If the bridge receives an 802.1D BPDU after a port's Migration Delay timer expires, the bridge assumes that it is connected to an 802.1D bridge and starts using only 802.1D BPDUs.
- When RSTP uses 802.1D BPDUs on a port and receives an RSTP BPDU after the migration delay expires, RSTP restarts the Migration Delay timer and begins using RSTP BPDUs on that port.

RSTP Port Roles

RSTP uses the following definitions for port roles:

- Root—A forwarding port that is elected for the spanning-tree topology.
- Designated—A forwarding port that is elected for every switched LAN segment.
- Alternate—An alternate path to the root bridge to that provided by the current root port.

- Backup—A backup for the path that is provided by a designated port toward the leaves of the spanning tree. Backup ports can exist only where two ports are connected together in a loopback by a point-to-point link or bridge with two or more connections to a shared LAN segment.
- Disabled—A port that has no role within the operation of spanning tree.

Port roles are assigned as follows:

- A root port or designated port role includes the port in the active topology.
- An alternate port or backup port role excludes the port from the active topology.

RSTP Port States

The port state controls the forwarding and learning processes and provides the values of discarding, learning, and forwarding. Table 7-3 provides a comparison between STP port states and RSTP port states.

Table 7-3 Comparison Between STP and RSTP Port States

Operational Status	STP Port State	RSTP Port State	Port Included in Active Topology
Enabled	Blocking[1]	Discarding[2]	No
Enabled	Listening	Discarding	No
Enabled	Learning	Learning	Yes
Enabled	Forwarding	Forwarding	Yes
Disabled	Disabled	Discarding	No

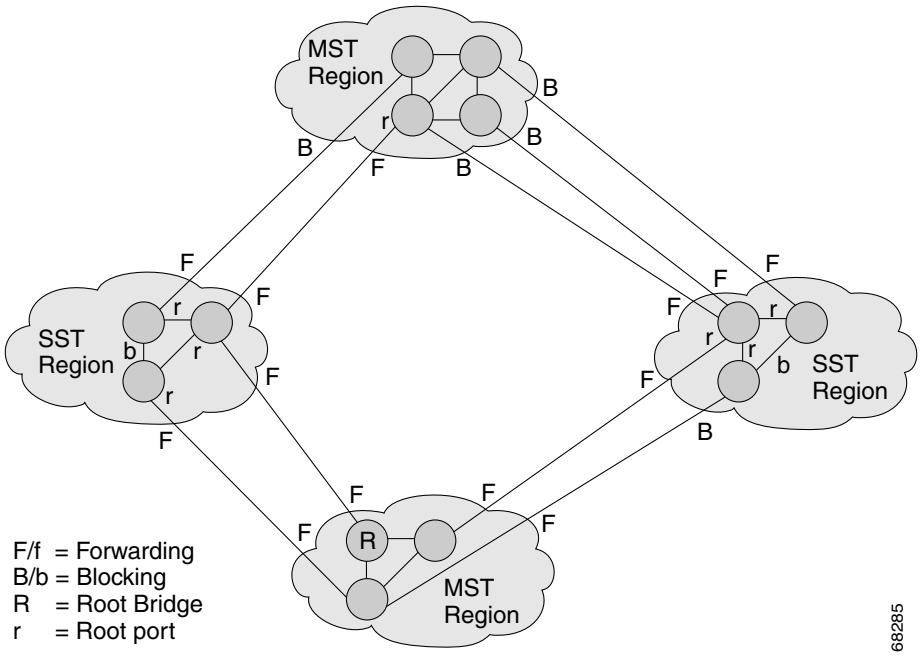
1. IEEE 802.1D port state designation.
2. IEEE 802.1w port state designation. Discarding is the same as blocking in MST.

In a stable topology, RSTP ensures that every root port and designated port transition to forwarding while all alternate ports and backup ports are always in the discarding state.

MST-to-SST Interoperability

A virtual bridged LAN may contain interconnected regions of SST and MST bridges. See Figure 7-10.

Figure 7-10 Network with Interconnected SST and MST Regions



To the spanning-tree protocol running in the SST region, an MST region appears as a single SST or pseudobridge. Pseudobridges operate as follows:

- The same values for root identifiers and root path costs are sent in all BPDUs of all the pseudobridge ports. Pseudobridges differ from a single SST bridge as follows:
  - The pseudobridge BPDUs have different bridge identifiers. This difference does not affect STP operation in the neighboring SST regions because the root identifier and root cost are the same.
  - BPDUs that are sent from the pseudobridge ports may have significantly different message ages. Because the message age increases by 1 second for each hop, the difference in the message age is in the order of seconds.
- Data traffic from one port of a pseudobridge (a port at the edge of a region) to another port follows a path that is entirely contained within the pseudobridge or MST region.
- Data traffic belonging to different VLANs may follow different paths within the MST regions that are established by MST.
- Loop prevention is achieved by either of the following:
  - Blocking the appropriate pseudobridge ports by allowing one forwarding port on the boundary and blocking all other ports.
  - Setting the CST partitions to block the ports of the SST regions.
- A pseudobridge differs from a single SST bridge because the BPDUs that are sent from the pseudobridge's ports have different bridge identifiers. The root identifier and root cost are the same for both bridges.

Common Spanning Tree

CST (802.1Q) is a single spanning tree for all the VLANs. In a Catalyst 6500 series switch running PVST+, the VLAN 1 spanning tree corresponds to CST. In a Catalyst 6500 series switch running MST, IST (instance 0) corresponds to CST.

MST Instances

MST supports up to 64 instances; each spanning-tree instance is identified by an instance ID that ranges from 0–63. Instance 0 is mandatory and is always present. Instances 1–63 are optional.

With software release 8.3(1) and later releases, the instance ID can range from 0–4094. Instances 1–4094 are optional.

MST Configuration

MST configuration has three parts as follows:

- Name—A 32-character string (null padded and null terminated) identifying the MST region.
- Revision number—An unsigned 16-bit number that identifies the revision of the current MST configuration.

Note You must set the revision number when required as part of the MST configuration. The revision number is not incremented automatically each time that the MST configuration is committed.

- MST configuration table—An array of 4096 elements representing all the possible extended-range VLANs. The value of element number X represents the instance to which VLAN X is mapped. VLAN 0 and VLAN 4095 are unused and are always mapped to the instance 0.

You must configure each byte manually. You can use SNMP or the CLI to perform the configuration.

MST BPDUs contain the MST configuration ID and the checksum. An MST bridge accepts an MST BPDU only if the MST BPDU configuration ID and the checksum match its own MST region configuration ID and checksum. If one value is different, the MST BPDU is treated as an SST BPDU.

When you modify an MST configuration through either a console or Telnet connection, the session exits without committing those changes and the edit buffer locks. Further configuration is impossible until you discard the existing edit buffer and acquire a new edit buffer by entering the **set spantree mst config rollback force** command.

With software release 8.3(1) and later releases, if you configure the MST configuration on a switch that is the VTP mode primary server for MST, all the other switches receive the MST configuration. For detailed information on VTP version 3 MST propagation, see the "Understanding How VTP Version 3 Works" section on page 10-12.

MST Region

Interconnected bridges that have the same MST configuration are referred to as an MST region. There is no limit on the number of MST regions in the network.

To form an MST region, bridges can be either of the following:

- An MST bridge that is the only member of the MST region.
- An MST bridge that is interconnected by a LAN. A LAN's designated bridge has the same MST configuration as an MST bridge. All the bridges on the LAN can process MST BPDUs.

If you connect two MST regions with different MST configurations, the MST regions do the following:

- Load balance across redundant paths in the network. If two MST regions are redundantly connected, all traffic flows on a single connection with the MST regions in a network.
- Provide an RSTP handshake to enable rapid connectivity between regions. However, the handshaking is not as fast as between two bridges. To prevent loops, all the bridges inside the region must agree upon the connections to other regions. This situation introduces a certain delay. We do not recommend partitioning the network into a large number of regions.

Switches running software release 8.3(1) and later releases form a different region than that of neighboring switches running earlier releases.

Boundary Ports

A boundary port is a port that connects to a LAN, the designated bridge of which, is either an SST bridge or a bridge with a different MST configuration. A designated port knows that it is on the boundary if it detects an STP bridge or receives an agreement message from an RST or MST bridge with a different configuration.

At the boundary, the role of MST ports does not matter; the MST-port state is forced to be the same as the IST port state. If the boundary flag is set for the port, the MSTP port role selection mechanism assigns a port role to the boundary and the same state as that of the IST port. The IST port at the boundary can take up any port role except a backup port role.

CIST Regional Root

The CIST regional root of an MST region is the bridge with the lowest bridge identifier and the least path cost to the CST root. If an MST bridge is the root bridge for CST, then it is the CIST regional root of that MST region. If the CST root is outside the MST region, then one of the MST bridges at the boundary is selected as the CIST regional root. Other bridges on the boundary that belong to the same region eventually block the boundary ports that lead to the root.

If two or more bridges at the boundary of a region have an identical path to the root, you can set a slightly lower bridge priority (higher port priority number) to make a specific bridge the CIST regional root.

The root path cost and message age inside a region stay constant, but the IST path cost is incremented and the IST remaining hops are decremented at each hop. Enter the **show spantree mst** command to display the information about the CIST regional root, path cost, and remaining hops for the bridge.

Edge Ports

A port that is connected to a nonbridging device (for example, a host or a router) is an edge port. A port that connects to a hub is also an edge port if the hub or any LAN that is connected by it does not have a bridge. An edge port can start forwarding as soon as the link is up.

MST requires that you configure all ports for each host or router. To establish rapid connectivity after a failure, you need to block the nonedge designated ports of an intermediate bridge. If the port connects to another bridge that can send back an agreement, then the port starts forwarding immediately. Otherwise, the port needs twice the forward delay time to start forwarding again. You must explicitly configure the ports that are connected to the hosts and routers as edge ports while using MST.

Note To configure a port as an edge port, you enable PortFast on that port. When you enter the **show spantree portfast** *mod/port* command, if the designation for a port is displayed as edge, that port is also a PortFast port. For more information, see Chapter 9, "Configuring Spanning-Tree PortFast, UplinkFast, BackboneFast, and Loop Guard."

To prevent a misconfiguration, the PortFast operation is turned off if the port receives a BPDU. You can display the configured and operational status of PortFast by using the **show spantree mst** *mod/port* command.

Link Type

Rapid connectivity is established only on point-to-point links. You must configure ports explicitly to a host or router. However, cabling in most networks meets this requirement, and you can avoid explicit configuration by treating all full-duplex links as point-to-point links by entering the **set spantree mst link-type** command.

Message Age and Hop Count

IST and MST instances do not use the Message Age and Maximum Age timer settings in the BPDU. IST and MST use a separate hop count mechanism that is very similar to the IP TTL mechanism. You can configure each MST bridge with a maximum hop count. The root bridge of the instance sends a BPDU (or M-record) with the remaining hop count that is equal to the maximum hop count. When a bridge receives a BPDU (or M-record), it decrements the received remaining hop count by one. The bridge discards the BPDU (M-record) and ages out the information that is held for the port if the count reaches zero after decrementing. The nonroot bridges propagate the decremented count as the remaining hop count in the BPDUs (M-records) that they generate.

The Message Age and Maximum Age timer settings in the RST portion of the BPDU remain the same throughout the region, and the same values are propagated by the region's designated ports at the boundary.

MST-to-PVST+ Interoperability

These guidelines apply in a topology where you configure MST switches (all in the same region) to interact with PVST+ switches that have VLANs 1–100 set up to span throughout the network:

- Configure the root for all VLANs inside the MST region. The ports that belong to the MST switch at the boundary simulate PVST+ and send PVST+ BPDUs for all the VLANs. This example shows the ports simulating PVST:

```
Console> (enable) show spantree mst 3
Spanning tree mode          MST
Instance                    3
VLANs Mapped:               31-40

Designated Root             00-10-7b-bb-2f-00
Designated Root Priority    8195  (root priority:8192, sys ID ext:3)
Designated Root Cost        0          Remaining Hops 20
Designated Root Port        1/0

Bridge ID MAC ADDR          00-10-7b-bb-2f-00
Bridge ID Priority          8195  (bridge priority:8192, sys ID ext:3)

Port                    State         Role Cost     Prio Type
----------------------- ------------- ---- -------- ------------------------
 6/1                    forwarding    BDRY    10000   30 P2P,
Boundary(PVST)
 6/2                    blocking      BDRY    20000   32 P2P,
Boundary(PVST)
```

  If you enable loop guard on the PVST+ switches, the ports might change to a loop-inconsistent state when the MST switches change their configuration. To correct the loop-inconsistent state, you must disable and reenable loop guard on that PVST+ switch.

- Do not locate the root for some or all of the VLANs inside the PVST+ side of the MST switch because when the MST switch at the boundary receives PVST+ BPDUs for all or some of the VLANs on its designated ports, root guard sets the port to the blocking state. Do not designate switches with a slower CPU running PVST+ as a switch running MST.

When you connect a PVST+ switch to two different MST regions, the topology change from the PVST+ switch does not pass beyond the first MST region. In this case, the topology changes are only propagated in the instance to which the VLAN is mapped. The topology change stays local to the first MST region and the CAM entries in the other region are not flushed To make the topology change visible throughout other MST regions, you can map that VLAN to IST or connect the PVST+ switch to the two regions through access links.

Understanding How BPDU Skewing Works

BPDU skewing is the difference in time between when the BPDUs are expected to be received by the switch and when the BPDUs are actually received by that switch. Skewing occurs as follows:

- Spanning-tree timers lapse.
- Expected BPDUs are not received by the switch.
- Spanning tree detects topology changes.

The skew causes BPDUs to reflood the network to keep the spanning-tree topology database current.

The root switch advertises its presence by sending out BPDUs for the configured hello time interval. The nonroot switches receive and process one BPDU during each configured time period. A VLAN may not receive the BPDU as scheduled. If the BPDU is not received on a VLAN at the configured time interval, the BPDU is skewed.

Spanning tree uses the hello time (see the "Configuring the Hello Time" section on page 7-50) to detect when a connection to the root switch exists through a port and when that connection is lost. This feature applies to both PVST+ and MISTP. In MISTP, the skew detection is on a per-instance basis.

BPDU skewing detects BPDUs that are not processed in a regular time frame on the nonroot switches in the network. If BPDU skewing occurs, a syslog message is displayed. The syslog applies to both PVST+ and MISTP.

The number of syslog messages that are generated may impact the network convergence and the CPU utilization of the switch. New syslog messages are not generated as individual messages for every VLAN because the higher the number of syslog messages that are reported, the slower the switching process will be. To reduce the impact on the switch, the syslog messages are as follows:

- Generated 50 percent of the maximum age time (see the "Configuring the Maximum Aging Time" section on page 7-51)
- Rate limited at one for every 60 seconds

Understanding How Layer 2 PDU Rate Limiting Works

You can use rate limiters to prevent receiving an unwanted number of protocol data units (PDUs) or more than a certain number of PDUs from a neighboring switch. The Layer 2 PDU rate limiters are supported in the hardware on the Catalyst 6500 series switches. They rate limit traffic on the Local Target Logic (LTL) index.

You can configure up to four rate limiters. You can configure rate limters to limit the following PDU types globally on the switch:

- Spanning-tree BPDUs—IEEE and SSTP, CDP, UDLD, VTP, and PAgP
- Layer 2 protocol tunnel-encapsulated PDUs
- 802.1X port security

These restrictions apply if you want to enable rate limiting:

- Hardware-based rate limiters are supported on Catalyst 6500 series switches that are configured with a PFC3A or later PFC.
- The Catalyst 6500 series switch cannot be in truncated mode. If you attempt to enable rate limiting and you are in truncated mode, an error message is displayed.
- If the rate limiter is enabled and some events cause the system to go from nontruncated mode to truncated mode, rate limiting is disabled and an informational message is displayed.

Configuring PVST+ on the Switch

These sections describe how to configure PVST+ on Ethernet VLANs:

Default PVST+ Configuration

Table 7-4 shows the default PVST+ configuration.

Table 7-4 PVST+ Default Configuration

Feature	Default Value
VLAN 1	All ports assigned to VLAN 1
Enable state	PVST+ enabled for all VLANs
MAC address reduction	Disabled
Bridge priority	32768
Bridge ID priority	32769 (bridge priority plus system ID extension of VLAN 1)
Port priority	32
Port cost	• 10-Gigabit Ethernet: 2 • Gigabit Ethernet: 4 • Fast Ethernet: 19[1] • FDDI/CDDI: 10 • Ethernet: 100[2]
Default spantree port cost mode	Short (802.1D)
Port VLAN priority	Same as port priority but configurable on a per-VLAN basis in PVST+
Port VLAN cost	Same as port cost but configurable on a per-VLAN basis in PVST+
Maximum aging time	20 seconds
Hello time	2 seconds
Forward delay time	15 seconds

1. If 10/100 Mbps ports autonegotiate or are hard set to 100 Mbps, the port cost is 19.
2. If 10/100 Mbps ports autonegotiate or are hard set to 10 Mbps, the port cost is 100.

Setting the PVST+ Bridge ID Priority

The bridge ID priority is the priority of a VLAN when the switch is in PVST+ mode.

When the switch is in PVST+ mode without MAC address reduction enabled, you can enter a bridge priority value between 0–65535. The bridge priority value that you enter also becomes the VLAN bridge ID priority for that VLAN.

When the switch is in PVST+ mode with MAC address reduction enabled, you can enter one of 16 bridge priority values: 0, 4096, 8192, 12288, 16384, 20480, 24576, 28672, 32768, 36864, 40960, 45056, 49152, 53248, 57344, or 61440.

The bridge priority is combined with the system ID extension (that is, the ID of the VLAN) to create the bridge ID priority for the VLAN.

To set the spanning-tree bridge priority for a VLAN, perform this task in privileged mode:

	Task	Command
Step 1	Set the PVST+ bridge ID priority for a VLAN.	**set spantree priority** *bridge_ID_priority* [*vlan*]
Step 2	Verify the bridge ID priority.	**show spantree** [*vlan*] [**active**]

This example shows how to set the PVST+ bridge ID priority when MAC address reduction is not enabled (default):

```
Console> (enable) set spantree priority 30000 1
Spantree 1 bridge priority set to 30000.
Console> (enable) show spantree 1
VLAN 1
Spanning tree mode          PVST+
Spanning tree type          ieee
Spanning tree enabled

Designated Root             00-60-70-4c-70-00
Designated Root Priority    16384
Designated Root Cost        19
Designated Root Port        2/3
Root Max Age   14 sec   Hello Time 2 sec   Forward Delay 10 sec

Bridge ID MAC ADDR          00-d0-00-4c-18-00
Bridge ID Priority          30000
Bridge Max Age 20 sec   Hello Time 2 sec   Forward Delay 15 sec

Port                     Vlan Port-State    Cost      Prio Portfast Channel_id
------------------------ ---- ------------- --------- ---- -------- ----------
 1/1                     1    not-connected         4   32 disabled 0
 1/2                     1    not-connected         4   32 disabled 0
 2/1                     1    not-connected       100   32 disabled 0
 2/2                     1    not-connected       100   32 disabled 0
```

This example shows how to set the PVST+ bridge ID priority when MAC reduction is enabled:

```
Console> (enable) set spantree priority 32768 1
Spantree 1 bridge ID priority set to 32769
(bridge priority: 32768 + sys ID extension: 1)
Console> (enable) show spantree 1/1 1
VLAN 1
Spanning tree mode          PVST+
Spanning tree type          ieee
Spanning tree enabled
```

```
Designated Root             00-60-70-4c-70-00
Designated Root Priority    16384
Designated Root Cost        19
Designated Root Port        2/3
Root Max Age   14 sec   Hello Time 2 sec   Forward Delay 10 sec

Bridge ID MAC ADDR          00-d0-00-4c-18-00
Bridge ID Priority          32769 (bridge priority: 32768, sys ID ext: 1)
Bridge Max Age 20 sec   Hello Time 2 sec   Forward Delay 15 sec

Port                     Vlan Port-State    Cost      Prio Portfast Channel_id
------------------------ ---- ------------- --------- ---- -------- ----------
 1/1                     1    not-connected         4   32 disabled 0
 1/2                     1    not-connected         4   32 disabled 0
 2/1                     1    not-connected       100   32 disabled 0
 2/2                     1    not-connected       100   32 disabled 0
```

Configuring the PVST+ Port Cost

You can configure the port cost of switch ports. The ports with lower port costs are more likely to be chosen to forward frames. Assign lower numbers to ports that are attached to faster media (such as full duplex) and higher numbers to ports that are attached to slower media. The possible cost is from 1–65535 when using the short method for calculating port cost and from 1–200000000 when using the long method. The default cost differs for different media. For information about calculating the port cost, see the "Calculating and Assigning Port Costs" section on page 7-4.

To configure the PVST+ port cost for a port, perform this task in privileged mode:

	Task	Command
Step 1	Configure the PVST+ port cost for a switch port.	**set spantree portcost** {*mod/port*} *cost*
Step 2	Verify the port cost setting.	**show spantree** *mod/port*

Note When you enter the **set spantree channelcost** command, it does not appear in the configuration file. The command causes a "set spantree portcost" entry to be created for each port in the channel. See the "Setting the EtherChannel Port Path Cost" section in Chapter 6, "Configuring EtherChannel," for information on using the **set spantree channelcost** command.

This example shows how to configure the PVST+ port cost on a port and verify the configuration:

```
Console> (enable) set spantree portcost 2/3 12
Spantree port 2/3 path cost set to 12.
Console> (enable) show spantree 2/3
VLAN 1
   .
   .
   .
Port                     Vlan Port-State    Cost      Prio Portfast Channel_id
------------------------ ---- ------------- --------- ---- -------- ----------
 1/1                     1    not-connected         4   32 disabled 0
 1/2                     1    not-connected         4   32 disabled 0
 2/1                     1    not-connected       100   32 disabled 0
 2/2                     1    not-connected       100   32 disabled 0
 2/3                     1    forwarding           12   32 disabled 0
 2/4                     1    not-connected       100   32 disabled
```

Configuring the PVST+ Port Priority

You can configure the port priority of switch ports in PVST+ mode. The port with the lowest priority value forwards frames for all VLANs. The possible port priority value is a multiple of 16 from 0–240. The default is 32. If all ports have the same priority value, the port with the lowest port number forwards frames.

To configure the PVST+ port priority for a port, perform this task in privileged mode:

	Task	Command
Step 1	Configure the PVST+ port priority for a switch port.	**set spantree portpri** *mod/port priority*
Step 2	Verify the port priority setting.	**show spantree** *mod/port*

This example shows how to configure the PVST+ port priority for a port:

```
Console> (enable) set spantree portpri 2/3 48
Bridge port  2/3 port priority set to 48.
Console> (enable) show spantree 2/3
VLAN 1
   .
   .
   .
Port                     Vlan Port-State    Cost      Prio Portfast Channel_id
------------------------ ---- ------------- --------- ---- -------- ----------
 1/1                     1    not-connected         4   32 disabled 0
 1/2                     1    not-connected         4   32 disabled 0
 2/1                     1    not-connected       100   32 disabled 0
 2/2                     1    not-connected       100   32 disabled 0
 2/3                     1    forwarding           19   48 disabled 0
 2/4                     1    not-connected       100   32 disabled 0
```

This example shows that values that are not multiples of 16 (between the values of 0–63) are converted to the nearest multiple of 16:

```
Console> (enable) set spantree portpri 2/3 2
Vlan port priority must be one of these numbers:0, 16, 32, 48, 64, 80,
96, 112, 128, 144,
160, 176, 192, 208, 224, 240
converting 2 to 0 nearest multiple of 16
Bridge port  2/3 port priority set to 0.
```

Configuring the PVST+ Default Port Cost Mode

If any switch in your network is using a port speed of 10 Gb or over and the network is using PVST+ spanning-tree mode, all switches in the network must have the same path cost defaults. You can enter the **set spantree defaultcostmode** command to force all VLANs that are associated with all the ports to have the same port cost default set.

Two default port cost modes are available—short and long.

- The short mode has these parameters:
  - Portcost
  - Portvlancost (trunk ports only)
  - When UplinkFast is enabled, the actual cost is incremented by 3000

- The long mode has these parameters:
  - Portcost
  - Portvlancost (trunk ports only)
  - When UplinkFast is enabled, the actual cost is incremented by 10,000,000
  - EtherChannel computes the cost of a bundle using the formula, AVERAGE_COST/NUM_PORT

The default port cost mode is set to short in PVST+ mode. For port speeds of 10 Gb and greater, the default port cost mode must be set to long.

To configure the PVST+ default port cost mode, perform this task in privileged mode:

Task	Command
Configure the PVST+ default port cost mode.	**set spantree defaultcostmode** {**short** | **long**}

This example shows how to configure the PVST+ default port cost mode:

```
Console> (enable) set spantree defaultcostmode long
Portcost and portvlancost set to use long format default values.
Console> (enable)
```

Configuring the PVST+ Port Cost for a VLAN

You can configure the port cost for a port on a per-VLAN basis. Ports with a lower port cost in the VLAN are more likely to be chosen to forward frames. You should assign lower numbers to ports that are attached to faster media (such as full duplex) and higher numbers to ports that are attached to slower media. The possible cost is from 1–65535 when using the short method for calculating port cost and from 1–200000000 when using the long method. The default cost differs for different media. For information about calculating port cost, see the "Calculating and Assigning Port Costs" section on page 7-4.

To configure the PVST+ port VLAN cost for a port, perform this task in privileged mode:

Task	Command
Configure the PVST+ port cost for a VLAN on a port.	**set spantree portvlancost** {*mod/port*} [**cost** *cost*] [*vlan_list*]

Note When you use the **set spantree channelcost** command, it does not appear in the configuration file. The command causes a "set spantree portcost" entry to be created for each port in the channel. See the "Setting the EtherChannel Port Path Cost" section in Chapter 6, "Configuring EtherChannel," for information on using the **set spantree channelcost** command.

This example shows how to configure the PVST+ port VLAN cost on port 2/3 for VLANs 1–5:

```
Console> (enable) set spantree portvlancost 2/3 cost 20000 1-5
Port 2/3 VLANs 6-11,13-1005,1025-4094 have path cost 12.
Port 2/3 VLANs 1-5,12 have path cost 20000.
This parameter applies to trunking ports only.
Console> (enable)
```

Configuring the PVST+ Port Priority for a VLAN

When the switch is in PVST+ mode, you can set the port priority for a trunking port in a VLAN. The port with the lowest priority value for a specific VLAN forwards frames for that VLAN. The possible port priority value is a multiple of 16 from 0–240. The default is 16. If all ports have the same priority value for a particular VLAN, the port with the lowest port number forwards frames for that VLAN.

The port VLAN priority value must be lower than the port priority value.

To configure the port VLAN priority for a port, perform this task in privileged mode:

	Task	Command
Step 1	Configure the PVST+ port priority for a VLAN on a port.	**set spantree portvlanpri** *mod/port priority* [*vlans*]
Step 2	Verify the port VLAN priority.	**show config all**

This example shows how to configure the port priority for VLAN 6 on port 2/3:

```
Console> (enable) set spantree portvlanpri 2/3 16 6
Port 2/3 vlans 6 using portpri 16.
Port 2/3 vlans 1-5,7-800,802-1004,1006-4094 using portpri 32.
Port 2/3 vlans 801,1005 using portpri 4.
This parameter applies to trunking ports only.
```

```
Console> (enable) show config all
   .
   .
   .
set spantree portcost    2/12,2/15 19
set spantree portcost    2/1-2,2/4-11,2/13-14,2/16-48 100
set spantree portcost    2/3 12
set spantree portpri     2/1-48 32
set spantree portvlanpri 2/1 0
set spantree portvlanpri 2/2 0
   .
   .
   .
set spantree portvlanpri 2/48 0
set spantree portvlancost 2/1 cost 99
set spantree portvlancost 2/2 cost 99
set spantree portvlancost 2/3 cost 20000 1-5,12
```

Disabling the PVST+ Mode on a VLAN

When the switch is in PVST+ mode, you can disable spanning tree on individual VLANs or all VLANs. When you disable spanning tree on a VLAN, the switch does not participate in spanning tree and any BPDUs that are received in that VLAN are flooded on all ports.

Caution We do not recommend disabling spanning tree, even in a topology that is free of physical loops. Spanning tree serves as a safeguard against misconfigurations and cabling errors. Do not disable spanning tree in a VLAN without ensuring that there are no physical loops present in the VLAN.

Caution Do not disable spanning tree on a VLAN unless all switches or routers in the VLAN have spanning tree disabled. You cannot disable spanning tree on some switches or routers in a VLAN and leave spanning tree enabled on other switches or routers in the VLAN. If spanning tree remains enabled on the switches and routers, they will have incomplete information about the physical topology of the network. This situation may cause unexpected results.

To disable PVST+, perform this task in privileged mode:

Task	Command
Disable PVST+ mode on a VLAN.	**set spantree disable** *vlans* [**all**]

This example shows how to disable PVST+ on a VLAN:

```
Console> (enable) set spantree disable 4
Spantree 4 disabled.
Console> (enable)
```

Configuring Rapid-PVST+ on the Switch

Rapid-PVST+ is the default spanning tree protocol that is used on all Ethernet, Fast Ethernet, and Gigabit Ethernet port-based VLANs on Catalyst 6500 series switches. To configure Rapid-PVST+, you need to also configure PVST+ on your switch. You can configure PVST+ either before or after you enable Rapid-PVST+.

To configure Rapid-PVST+, perform this task in privileged mode:

	Task	Command
Step 1	Enable Rapid-PVST+.	**set spantree mode rapid-pvst+**
Step 2	Set the link-type to point-to-point mode for the port.	**set spantree link-type** *mod/port* **point-to-point**
Step 3	Detect any legacy bridges on the port.	**clear spantree detected-protocols** *mod/port*
Step 4	Verify the Rapid-PVST+ configuration.	**show spantree** *vlan*

This example shows how to configure Rapid-PVST+:

```
Console> (enable) set spantree mode rapid-pvst+
Spantree mode set to RAPID-PVST+.
Console> (enable) set spantree link-type 3/1 point-to-point
Link type set to point-to-point on port 3/1.
Console> (enable) clear spantree detected-protocols 3/1
Spanning tree protocol detection forced on port 3/1
Console> (enable)
```

This example shows how to verify the Rapid-PVST+ configuration for VLAN 1. Notice that the first line in the output displays the spanning-tree mode.

```
Console> show spantree 1
Spanning tree mode          RAPID-PVST+
Spanning tree type          ieee
Spanning tree enabled.
.
.
.
Port          State       Role     Cost    Prio   Type
------------  ----------- -------  -----   ----   -----------------
6/1          forwarding  ROOT     20000    16    Shared, PEER(STP)

Console> (enable)
```

This example shows how to verify the link type, edge port, and guard type for port 3/6:

```
Console> show spantree 3/6
Port 3/6
Edge Port:      No, (Configured) Default
Port Guard:     Default
Link Type:      P2P(Configured) Auto

Port     VLAN     State       Role    Cost     Prio  Type
------   -----    ----------  ------  -------- ----  -----
3/6       1       listening   DESG    20000    32    P2P
3/6       2       listening   DESG    20000    32    P2P
3/6       3       listening   DESG    20000    32    P2P
3/6       4       listening   DESG    20000    32    P2P
3/6       5       listening   DESG    20000    32    P2P
3/6       6       listening   DESG    20000    32    P2P
```

```
3/6        7       listening   DESG    20000    32    P2P
3/6        8       listening   DESG    20000    32    P2P
3/6        9       listening   DESG    20000    32    P2P
3/6        10      listening   DESG    20000    32    P2P
3/6        11      listening   DESG    20000    32    P2P
3/6        12      listening   DESG    20000    32    P2P
3/6        13      listening   DESG    20000    32    P2P
3/6        14      listening   DESG    20000    32    P2P
3/6        15      listening   DESG    20000    32    P2P
3/6        16      listening   DESG    20000    32    P2P
3/6        17      listening   DESG    20000    32    P2P
3/6        18      listening   DESG    20000    32    P2P
3/6        19      listening   DESG    20000    32    P2P

Console> (enable)
```

Configuring MISTP-PVST+ or MISTP on the Switch

The default spanning-tree mode on the Catalyst 6500 series switches is Rapid-PVST+ mode. If you want to use MISTP mode in your network, we recommend that you carefully follow the procedures that are described in the following sections in order to avoid losing connectivity in your network.

When you change the spanning-tree mode, the current mode stops, the information collected at runtime is used to build the port database for the new mode, and the new spanning-tree mode restarts the computation of the active topology. Information about the port states is lost; however, all of the configuration parameters are preserved for the previous mode. If you return to the previous mode, the configuration is still there.

Note We recommend that if you use MISTP mode, you should configure *all* of your Catalyst 6500 series switches to run MISTP.

To use MISTP mode, you first enable an MISTP instance and then map at least one VLAN to the instance. You must have at least one forwarding port in the VLAN in order for the MISTP instance to be active.

Note Map VLANs to MISTP instances on Catalyst 6500 series switches that are either in VTP server mode or transparent mode only. You cannot map VLANs to MISTP instances on switches that are in VTP client mode. To avoid VTP configuration errors that could cause problems with your MISTP configuration, see Chapter 10, “Configuring VTP” for detailed information on using VTP versions 1, 2, and 3.

If you are changing a switch from PVST+ mode to MISTP mode and you have other switches in the network that are using PVST+, you must first enable MISTP-PVST+ mode on each switch on which you intend to use MISTP so that PVST+ BPDUs can flow through the switches while you configure them.

When all switches in the network are configured in MISTP-PVST+, you can then enable MISTP on all of the switches.

These sections describe how to use MISTP-PVST+ or MISTP:

- Mapping VLANs to an MISTP Instance, page 7-41
- Disabling MISTP-PVST+ or MISTP, page 7-44

Default MISTP and MISTP-PVST+ Configuration

Table 7-5 shows the default MISTP and MISTP-PVST+ configuration.

Table 7-5 MISTP and MISTP-PVST+ Default Configuration

Feature	Default Value
Enable state	Disabled until a VLAN is mapped to an MISTP instance
MAC address reduction	Disabled
Bridge priority	32768
Bridge ID priority	32769 (bridge priority plus the system ID extension of MISTP instance 1)
Port priority	32 (global)
Port cost	• 10-Gigabit Ethernet: 2 • Gigabit Ethernet: 4 • Fast Ethernet: 19[1] • FDDI/CDDI: 10 • Ethernet: 100[2]
Default port cost mode	Short (802.1D)
Port VLAN priority	Same as port priority but configurable on a per-VLAN basis in PVST+
Port VLAN cost	Same as port cost but configurable on a per-VLAN basis in PVST+
Maximum aging time	20 seconds
Hello time	2 seconds
Forward delay time	15 seconds

1. If 10/100-Mbps ports autonegotiate or are hard set to 100 Mbps, the port cost is 19.
2. If 10/100-Mbps ports autonegotiate or are hard set to 10 Mbps, the port cost is 100.

Setting the MISTP-PVST+ Mode or the MISTP Mode

If you enable MISTP in a PVST+ network, you must be careful to avoid bringing down the network. This section explains how to enable MISTP or MISTP-PVST+ on your network.

Caution If you have more than 6000 VLAN ports that are configured on your switch, changing from MISTP to either PVST+ or MISTP-PVST+ mode could bring down your network. Reduce the number of configured VLAN ports on your switch to no more than 6000 to avoid losing connectivity.

Caution If you are working from a Telnet connection to your switch, the first time that you enable MISTP-PVST+ or MISTP mode, you must do so from the switch console; do not use a Telnet connection through the data port or you will lose your connection to the switch. After you map a VLAN to an MISTP instance, you can Telnet to the switch.

To change from PVST+ to MISTP-PVST+ or MISTP, perform this task in privileged mode:

Task	Command
Set a spanning-tree mode.	**set spantree mode** {**mistp** | **pvst+** | **mistp-pvst+**}

This example shows how to set a switch to MISTP-PVST+ mode:

```
Console> (enable) set spantree mode mistp-pvst+
PVST+ database cleaned up.
Spantree mode set to MISTP-PVST+.
Warning!! There are no VLANs mapped to any MISTP instance.
Console> (enable)
```

You can display VLAN-to-MISTP instance mapping information that is propagated from the root switch at runtime. This display is available only in the MISTP or MISTP-PVST+ mode. In the PVST+ mode, use the optional keyword **config** to display the list of mappings that is configured on the local switch.

Note MAC addresses are not displayed when you specify the **config** keyword.

To display spanning-tree mapping, perform this task in privileged mode:

	Task	Command
Step 1	Set the spanning-tree mode to MISTP.	**set spantree mode mistp**
Step 2	Show the spanning tree mapping.	**show spantree mapping** [**config**]

This example shows how to display the spanning-tree VLAN instance mapping in MISTP mode:

```
MISTP/MISTP-PVST+
Console> (enable) set spantree mode mistp
PVST+ database cleaned up.
Spantree mode set to MISTP.
Console> (enable) show spantree mapping
Inst Root Mac          Vlans
---- ----------------- --------------------------
1    00-50-3e-78-70-00 1
2    00-50-3e-78-70-00 -
3    00-50-3e-78-70-00 -
4    00-50-3e-78-70-00 -
5    00-50-3e-78-70-00 -
6    00-50-3e-78-70-00 -
7    00-50-3e-78-70-00 -
8    00-50-3e-78-70-00 -
9    00-50-3e-78-70-00 -
10   00-50-3e-78-70-00 -
11   00-50-3e-78-70-00 -
12   00-50-3e-78-70-00 -
13   00-50-3e-78-70-00 -
```

```
14   00-50-3e-78-70-00 -
15   00-50-3e-78-70-00 -
16   00-50-3e-78-70-00 -
```

Configuring an MISTP Instance

These sections describe how to configure MISTP instances:

Configuring the MISTP Bridge ID Priority

You can set the bridge ID priority for an MISTP instance when the switch is in MISTP or MISTP-PVST+ mode.

The bridge priority value is combined with the system ID extension (the ID of the MISTP instance) to create the bridge ID priority. You can set 16 possible bridge priority values: 0, 4096, 8192, 12288, 16384, 20480, 24576, 28672, 32768, 36864, 40960, 45056, 49152, 53248, 57344, and 61440.

To configure the bridge ID priority for an MISTP instance, perform this task in privileged mode:

	Task	Command
Step 1	Configure the bridge ID priority for an MISTP instance.	**set spantree priority** *bridge_ID_priority* [**mistp-instance** *instance*]
Step 2	Verify the bridge ID priority.	**show spantree mistp-instance** *instance* [*mod/port*] **active**

This example shows how to configure the bridge ID priority for an MISTP instance:

```
Console> (enable) set spantree priority 32768 mistp-instance 1
Spantree 1 bridge ID priority set to 32769
(bridge priority: 32768 + sys ID extension: 1)
Console> (enable) show spantree mistp-instance 1
Instance 1
Spanning tree mode          MISTP
Spanning tree type          ieee
Spanning tree instance enabled

Designated Root             00-05-31-40-64-00
Designated Root Priority    32769  (root priority:32768, sys ID ext:1)
Designated Root Cost        20000
Designated Root Port        1/1
VLANs mapped:               1,74
Root Max Age   20 sec   Hello Time 2  sec   Forward Delay 15 sec

Bridge ID MAC ADDR          00-d0-02-27-9c-00
Bridge ID Priority          32769  (bridge priority:32768, sys ID ext:1)
VLANs mapped:               1,74
Bridge Max Age 20 sec   Hello Time 2  sec   Forward Delay 15 sec
```

```
Port                     Inst Port-State    Cost      Prio Portfast Channel_id
------------------------ ---- ------------ --------- ---- -------- ----------
 1/1                     1    forwarding        20000   32 disabled 0
 3/1                     1    forwarding       200000   32 disabled 0
 3/25                    1    forwarding       200000   32 disabled 0
 3/26                    1    forwarding       200000   32 disabled 0
 3/27                    1    forwarding       200000   32 disabled 0
 3/28                    1    forwarding       200000   32 disabled 0
 3/29                    1    forwarding       200000   32 disabled 0
 3/30                    1    forwarding       200000   32 disabled 0
7/1-4                    1    blocking           5000   32 disabled 833
 7/5                     1    forwarding        20000   32 disabled 0
 7/6                     1    forwarding        20000   32 disabled 0
 8/37                    1    blocking         200000   32 disabled 0
 8/38                    1    blocking         200000   32 disabled 0
15/1                     1    forwarding        20000   32 enabled  0
16/1                     1    forwarding        20000   32 enabled  0
Console> (enable)
```

Configuring the MISTP Port Cost

You can configure the port cost of switch ports. The ports with lower port costs are more likely to be chosen to forward frames. Assign lower numbers to ports that are attached to faster media (such as full duplex) and higher numbers to ports that are attached to slower media. The possible range of cost is from 1–65535 when using the short method for calculating port cost and from 1–200000000 when using the long method. The default cost differs for different media. For information about calculating path cost, see the "Calculating and Assigning Port Costs" section on page 7-4.

To configure the port cost for a port, perform this task in privileged mode:

	Task	Command
Step 1	Configure the MISTP port cost for a switch port.	**set spantree portcost** *mod/port cost*
Step 2	Verify the port cost setting.	**show spantree mistp-instance** *instance* [*mod/port*] **active**

This example shows how to configure the port cost on an MISTP instance and verify the configuration:

```
Console> (enable) set spantree portcost 1/1 20000
Spantree port 1/1 path cost set to 20000.
Console> (enable) show spantree mistp-instance 1 active
Instance 1
Spanning tree mode          MISTP
Spanning tree type          ieee
Spanning tree instance enabled

Designated Root             00-05-31-40-64-00
Designated Root Priority    32769  (root priority:32768, sys ID ext:1)
Designated Root Cost        20000
Designated Root Port        1/1
VLANs mapped:               1,74
Root Max Age   20 sec   Hello Time 2  sec   Forward Delay 15 sec

Bridge ID MAC ADDR          00-d0-02-27-9c-00
Bridge ID Priority          32769  (bridge priority:32768, sys ID ext:1)
VLANs mapped:               1,74
Bridge Max Age 20 sec   Hello Time 2  sec   Forward Delay 15 sec
```

```
Port                     Inst Port-State    Cost      Prio Portfast Channel_id
------------------------ ---- ------------ --------- ---- -------- ----------
 1/1                     1    forwarding        20000   32 disabled 0
 3/1                     1    forwarding       200000   32 disabled 0
 3/25                    1    forwarding       200000   32 disabled 0
 3/26                    1    forwarding       200000   32 disabled 0
 3/27                    1    forwarding       200000   32 disabled 0
 3/28                    1    forwarding       200000   32 disabled 0
 3/29                    1    forwarding       200000   32 disabled 0
 3/30                    1    forwarding       200000   32 disabled 0
7/1-4                    1    blocking           5000   32 disabled 833
 7/5                     1    forwarding        20000   32 disabled 0
 7/6                     1    forwarding        20000   32 disabled 0
 8/37                    1    blocking         200000   32 disabled 0
 8/38                    1    blocking         200000   32 disabled 0
15/1                     1    forwarding        20000   32 enabled  0
16/1                     1    forwarding        20000   32 enabled  0
Console> (enable)
```

Configuring the MISTP Port Priority

You can configure the port priority of ports. The port with the lowest priority value forwards frames for all VLANs. The possible port priority value is a multiple of 16 from 0–240. The default is 32. If all ports have the same priority value, the port with the lowest port number forwards frames.

To configure the port priority for a port, perform this task in privileged mode:

	Task	Command
Step 1	Configure the MISTP port priority for a port.	**set spantree portpri** *mod/port*
Step 2	Verify the port priority setting.	**show spantree mistp-instance** *instance* [*mod/port*] **active**

This example shows how to configure the port priority and verify the configuration:

```
Console> (enable) set spantree portpri 1/1 32
Bridge port 1/1 port priority set to 32.
Console> (enable) show spantree mistp-instance 1
Instance 1
Spanning tree mode          MISTP
Spanning tree type          ieee
Spanning tree instance enabled

Designated Root             00-05-31-40-64-00
Designated Root Priority    32769  (root priority:32768, sys ID ext:1)
Designated Root Cost        20000
Designated Root Port        1/1
VLANs mapped:               1,74
Root Max Age   20 sec   Hello Time 2  sec   Forward Delay 15 sec

Bridge ID MAC ADDR          00-d0-02-27-9c-00
Bridge ID Priority          32769  (bridge priority:32768, sys ID ext:1)
VLANs mapped:               1,74
Bridge Max Age 20 sec   Hello Time 2  sec   Forward Delay 15 sec
```

```
Port                     Inst Port-State    Cost      Prio Portfast Channel_id
------------------------ ---- ------------ --------- ---- -------- ----------
 1/1                     1    forwarding        20000   32 disabled 0
 3/1                     1    forwarding       200000   32 disabled 0
 3/25                    1    forwarding       200000   32 disabled 0
 3/26                    1    forwarding       200000   32 disabled 0
 3/27                    1    forwarding       200000   32 disabled 0
 3/28                    1    forwarding       200000   32 disabled 0
 3/29                    1    forwarding       200000   32 disabled 0
 3/30                    1    forwarding       200000   32 disabled 0
7/1-4                    1    blocking           5000   32 disabled 833
 7/5                     1    forwarding        20000   32 disabled 0
 7/6                     1    forwarding        20000   32 disabled 0
 8/37                    1    blocking         200000   32 disabled 0
 8/38                    1    blocking         200000   32 disabled 0
15/1                     1    forwarding        20000   32 enabled  0
16/1                     1    forwarding        20000   32 enabled  0
Console> (enable)
```

Configuring the MISTP Port Instance Cost

You can configure the port instance cost for an instance of MISTP or MISTP-PVST+. Ports with a lower instance cost are more likely to be chosen to forward frames. You should assign lower numbers to ports that are attached to faster media (such as full duplex) and higher numbers to ports that are attached to slower media. The default cost differs for different media. The possible value for port instance cost is 1–268435456.

To configure the port instance cost for a port, perform this task in privileged mode:

Task	Command
Configure the MISTP port instance cost on a port.	**set spantree portinstancecost** {*mod/port*} [**cost** *cost*] [*instances*]

This example shows how to configure the MISTP port instance cost on a port:

```
Console> (enable) set spantree portinstancecost 1/1 cost 110110 2
Port 1/1 instances 1,3-16 have path cost 20000.
Port 1/1 instances 2 have path cost 110110.
This parameter applies to trunking ports only.
Console> (enable)
```

Configuring the MISTP Port Instance Priority

You can set the port priority for an instance of MISTP. The port with the lowest priority value for a specific MISTP instance forwards frames for that instance. The possible port instance range is 0–63. The possible port priority value is a multiple of 16 from 0–240. If all ports have the same priority value for an MISTP instance, the port with the lowest port number forwards frames for that instance.

To configure the port instance priority on an MISTP instance, perform this task in privileged mode:

Task	Command
Configure the port instance priority on an MISTP instance.	**set spantree portinstancepri** {*mod/port*} *priority* [*instances*]

This example shows how to configure the port instance priority on an MISTP instance and verify the configuration:

```
Console> (enable) set spantree portinstancepri 1/1 16 2
Port 1/1 MISTP Instances 2 using portpri 16.
Port 1/1 mistp-instance 1,3-16 using portpri 32.
Console> (enable)
```

Enabling an MISTP Instance

You can enable up to 16 MISTP instances. Each MISTP instance defines a unique spanning-tree topology. MISTP instance 1, the default instance, is enabled by default; however, you must map a VLAN to it in order for it to be active. You can enable a single MISTP instance, a range of instances, or all instances at once using the **all** keyword.

Note The software does not display the status of an MISTP instance until it has a VLAN with an active port that is mapped to it.

To enable an MISTP instance, perform this task in privileged mode:

	Task	Command
Step 1	Enable an MISTP instance.	**set spantree enable mistp-instance** *instance* [**all**]
Step 2	Verify that the instance is enabled.	**show spantree mistp-instance** [*instance*] [**active**] *mod/port*

Note Enter the **active** keyword to display active ports only.

This example shows how to enable an MISTP instance:

```
Console> (enable) set spantree enable mistp-instance 2
Spantree 2 enabled.

Console> (enable) show spantree mistp-instance 2
Instance 2
Spanning tree mode          MISTP
Spanning tree type          ieee
Spanning tree instance enabled
```

Mapping VLANs to an MISTP Instance

When you are using MISTP-PVST+ or MISTP on a switch, you must map at least one VLAN to an MISTP instance in order for MISTP-PVST+ or MISTP to be active. These sections describe how to configure MISTP instances:

- Determining MISTP Instances—VLAN Mapping Conflicts, page 7-42
- Unmapping VLANs from an MISTP Instance, page 7-44

Note See Chapter 11, “Configuring VLANs” for details on using and configuring VLANs.

Follow these guidelines when mapping VLANs to an MISTP instance:

- You can map only Ethernet VLANs to MISTP instances.
- At least one VLAN in the instance must have an active port in order for MISTP-PVST+ or MISTP to be active.
- You can map as many Ethernet VLANs as you wish to an MISTP instance.
- You cannot map a VLAN to more than one MISTP instance.

Note To use VLANs 1025–4094, you must enable MAC address reduction. See the "Creating Extended-Range VLANs" section on page 11-7 in Chapter 11, "Configuring VLANs" for details on using extended-range VLANs.

To map a VLAN to an MISTP instance, perform this task in privileged mode:

	Task	Command
Step 1	Map a VLAN to an MISTP instance.	**set vlan** *vlan* **mistp-instance** *instance*
Step 2	Verify the VLAN is mapped.	**show spantree mistp-instance** [*instance*] [**active**] *mod/port*

This example shows how to map a VLAN to MISTP instance 1 and verify the mapping:

```
Console> (enable) set vlan 6 mistp-instance 1
Vlan 6 configuration successful
Console> (enable) show spantree mist-instance 1
Instance 1
Spanning tree mode          MISTP-PVST+
Spanning tree type          ieee
Spanning tree instance enabled

Designated Root             00-d0-00-4c-18-00
Designated Root Priority    49153 (root priority: 49152, sys ID ext: 1)
Designated Root Cost        0
Designated Root Port        none
VLANs mapped:               6
Root Max Age   20 sec   Hello Time 2 sec   Forward Delay 15 sec

Bridge ID MAC ADDR          00-d0-00-4c-18-00
Bridge ID Priority          49153 (bridge priority: 49152, sys ID ext: 1)
VLANs mapped:               6
Bridge Max Age 20 sec   Hello Time 2 sec   Forward Delay 15 sec
Port                      Inst Port-State    Cost      Prio Portfast Channel_id
------------------------ ---- ------------- --------- ---- -------- ----------
 2/12                     1    forwarding     22222222   40 disabled 0
```

Determining MISTP Instances—VLAN Mapping Conflicts

A VLAN can only be mapped to one MISTP instance. If you attempt to map a VLAN to more than one instance, all of its ports are set to blocking mode. You can use the **show spantree conflicts** command to determine to which MISTP instances you have attempted to map the VLAN.

This command prints a list of the MISTP instances that are associated with the VLAN, the MAC addresses of the root switches that are sending the BPDUs containing the VLAN mapping information, and the timers that are associated with the mapping of a VLAN to an MISTP instance. When only one

entry is printed or when all the entries are associated to the same instance, the VLAN is mapped to that instance. If two or more entries in the list are associated with different MISTP instances, the VLAN is in conflict.

To clear up the conflict, you must manually remove the incorrect mapping(s) from the root switch. The remaining entry on the list becomes the official mapping.

To determine VLAN mapping conflicts, perform this task in privileged mode:

Task	Command
Determine VLAN mapping conflicts.	**show spantree conflicts** *vlan*

This example shows that there is an attempt to map VLAN 2 to MISTP instance 1 and to MISTP instance 3 on two different switches as seen from a third switch in the topology:

```
Console> (enable) show spantree conflicts 2
Inst MAC               Delay     Time left
---- ----------------- --------- ---------
1   00-30-a3-4a-0c-00 inactive        20
3   00-30-f1-e5-00-01 inactive        10
```

The Delay timer shows the time in seconds remaining before the VLAN joins the instance. The field displays *inactive* if the VLAN is already mapped to an instance (the timer has expired), or if the VLAN is in conflict between instances.

The Time Left timer shows the time in seconds left before the entry expires and is removed from the table. The timer is restarted every time an incoming BPDU confirms the mapping. Entries pertaining to the root switch show *inactive* on the root switch itself.

The following examples are with VTP version 3 enabled. The root switch is also the primary server for the nonroot switch. The root switch is not the primary server for the switch in conflict, because that switch has been partitioned.

This example is from the root switch:

```
Console> (enable) show spantree conflicts 1
No conflicts for vlan 1.
Inst MAC               Delay     Time left
---- ----------------- --------- ---------
  1  00-05-31-40-64-00  inactive  inactive
Console> (enable)
```

This example is from the nonroot switch:

```
Console> (enable) show spantree conflicts 3
No conflicts for vlan 3.
Inst MAC               Delay     Time left
---- ----------------- --------- ---------
  3  00-05-31-40-64-00  inactive        19
Console> (enable)
```

This example is from the switch in conflict (note that the switch is inactive):

```
Console> (enable) show spantree conflicts 6
Inst MAC               Delay     Time left
---- ----------------- --------- ---------
  6  00-05-31-40-64-00  inactive        18
  5  00-09-7b-62-b0-80  inactive  inactive

Console> (enable)
```

Unmapping VLANs from an MISTP Instance

The **none** keyword is used to unmap the specified VLANs from the MISTP instances to which they are currently mapped. When you unmap a VLAN from an MISTP instance, the resulting state of all the ports of the VLAN (if the VLAN exists) is *blocking*.

To unmap a VLAN or all VLANs from an MISTP instance, perform this task in privileged mode:

Task	Command
Unmap a VLAN from an MISTP instance.	**set vlan** *vlan* **mistp-instance none**

This example shows how to unmap a VLAN from an MISTP instance:

```
Console> (enable) set vlan 6 mistp-instance none
Vlan 6 configuration successful
```

Disabling MISTP-PVST+ or MISTP

When the switch is in MISTP mode, you disable spanning tree on an instance, not for the whole switch.

When you disable spanning tree on an MISTP instance, the instance still exists on the switch, all of the VLANs mapped to it have all of their ports forwarding, and the instance BPDUs are flooded.

To disable an MISTP instance, perform this task in privileged mode:

Task	Command
Disable an MISTP instance.	**set spantree disable mistp-instance** *instance* [**all**]

This example shows how to disable an MISTP instance:

```
Console> (enable) set spantree disable mistp-instance 2
MI-STP instance 2 disabled.
```

Configuring a Root Switch

These sections describe how to configure a root switch:

Configuring a Primary Root Switch

You can set a root switch on a VLAN when the switch is in PVST+ mode or on an MISTP instance when the switch is in MISTP mode. You enter the **set spantree root** command to reduce the bridge priority (the value that is associated with the switch) from the default (32768) to a lower value, which allows the switch to become the root switch.

When you specify a switch as the primary root, the default bridge priority is modified so that it becomes the root for the specified VLANs. The switch checks the bridge priority of the current root switches for each VLAN. The bridge priority for the specified VLANs is set to 8192 if this value will cause the switch to become the root for the specified VLANs. If any root switch for the specified VLANs has a bridge priority lower than 8192, the switch sets the bridge priority for the specified VLANs to 1 less than the lowest bridge priority. Because different VLANs could potentially have different root switches, the bridge VLAN-priority chosen makes this switch the root for all the VLANs that are specified. If reducing the bridge priority as low as 1 still does not make the switch the root switch, the system displays a message.

Caution Enter the **set spantree root** command on backbone switches or distribution switches only; do not enter this command on access switches.

To configure a switch as the primary root switch, perform this task in privileged mode:

Task	Command
Configure a switch as the primary root switch.	**set spantree root** [*vlans*] [**dia** *network_diameter*] [**hello** *hello_time*]

This example shows how to configure the primary root switch for VLANs 1–10:

```
Console> (enable) set spantree root 1-10 dia 4
VLANs 1-10 bridge priority set to 8192
VLANs 1-10 bridge max aging time set to 14 seconds.
VLANs 1-10 bridge hello time set to 2 seconds.
VLANs 1-10 bridge forward delay set to 9 seconds.
Switch is now the root switch for active VLANs 1-6.
Console> (enable)
```

To configure a switch as the primary root switch for an instance, perform this task in privileged mode:

Task	Command
Configure a switch as the primary root switch for an instance.	**set spantree root mistp-instance** *instance* [**dia** *network_diameter*] [**hello** *hello_time*]

This example shows how to configure the primary root switch for an instance:

```
Console> (enable) set spantree root mistp-instance 2-4 dia 4
Instances 2-4 bridge priority set to 8192
VLInstances 2-4 bridge max aging time set to 14 seconds.
Instances 2-4 bridge hello time set to 2 seconds.
Instances 2-4 bridge forward delay set to 9 seconds.
Switch is now the root switch for active Instances 1-6.
Console> (enable)
```

Configuring a Secondary Root Switch

You can set a secondary root switch on a VLAN when the switch is in PVST+ mode or on an MISTP instance when the switch is in MISTP mode.

The **set spantree root secondary** command reduces the bridge priority to 16,384, making it the probable candidate to become the root switch if the primary root switch fails. You can run this command on more than one switch to create multiple backup switches in case the primary root switch fails.

To configure a switch as the secondary root switch, perform this task in privileged mode:

Task	Command
Configure a switch as the secondary root switch.	**set spantree root** [**secondary**] *vlans* [**dia** *network_diameter*] [**hello** *hello_time*]

This example shows how to configure the secondary root switch for VLANs 22 and 24:

```
Console> (enable) set spantree root secondary 22,24 dia 5 hello 1
VLANs 22,24 bridge priority set to 16384.
VLANs 22,24 bridge max aging time set to 10 seconds.
VLANs 22,24 bridge hello time set to 1 second.
VLANs 22,24 bridge forward delay set to 7 seconds.
Console> (enable)
```

To configure a switch as the secondary root switch for an instance, perform this task in privileged mode:

Task	Command
Configure a switch as the secondary root switch for an instance.	**set spantree root** [**secondary**] **mistp-instance** *instance* [**dia** *network_diameter*] [**hello** *hello_time*]

This example shows how to configure the secondary root switch for an instance:

```
Console> (enable) set spantree root secondary mistp-instance 2-4 dia 4
Instances 2-4 bridge priority set to 8192
VLInstances 2-4 bridge max aging time set to 14 seconds.
Instances 2-4 bridge hello time set to 2 seconds.
Instances 2-4 bridge forward delay set to 9 seconds.
Switch is now the root switch for active Instances 1-6.
Console> (enable)
```

Configuring a Root Switch to Improve Convergence

By lowering the values for the Hello Time, Forward Delay Timer, and Maximum Age Timer parameters on the root switch, you can reduce the convergence time. For information on configuring these timers, see the "Configuring Spanning-Tree Timers on the Switch" section on page 7-49.

Note Reducing the timer parameter values is possible only if your network has LAN links of 10 Mbps or faster. In a network with links of 10 Mbps or faster, the network diameter can reach the maximum value of 7. With WAN connections, you cannot reduce the parameters.

When a link failure occurs in a bridged network, the network reconfiguration is not immediate. Reconfiguring the default parameters (specified by IEEE 802.1D) for the Hello Time, Forward Delay Timer, and Maximum Age Timer requires a 50-second delay. This reconfiguration time depends on the network diameter, which is the maximum number of bridges between any two end stations.

To speed up convergence, use the nondefault parameter values that are permitted by 802.1D. See Table 7-6 for the nondefault parameters for a reconvergence of 14 seconds.

Table 7-6 Nondefault Parameters

Parameter	Time
Network Diameter (dia)	2
Hello Time	2 seconds
Forward Delay Timer	4 seconds
Maximum Age Timer	6 seconds

Note You can set the switch ports in PortFast mode for improved convergence. PortFast mode affects only the transition from disable (link down) to enable (link up) by moving the port immediately to the forwarding state. If a port in the PortFast mode begins blocking, it then goes through listening and learning before reaching the forwarding state. For information about PortFast, see the "Understanding How PortFast Works" section on page 9-2 in Chapter 9, "Configuring Spanning-Tree PortFast, UplinkFast, BackboneFast, and Loop Guard."

To configure the spanning tree parameters to improve convergence, perform this task in privileged mode:

	Task	Command
Step 1	Configure the hello time for a VLAN or an MISTP instance.	**set spantree hello** *interval* [*vlan*] **mistp-instance** [*instances*]
Step 2	Verify the configuration.	**show spantree** [*vlan* | **mistp-instance** *instances*]
Step 3	Configure the forward delay time for a VLAN or an MISTP instance.	**set spantree fwddelay** *delay* [*vlan*] **mistp-instance** [*instances*]
Step 4	Verify the configuration.	**show spantree** [*mod/port*] **mistp-instance** [*instances*] [**active**]
Step 5	Configure the maximum aging time for a VLAN or an MISTP instance.	**set spantree maxage** *agingtime* [*vlans*] **mistp-instance** *instances*
Step 6	Verify the configuration.	**show spantree** [*mod/port*] **mistp-instance** [*instances*] [**active**]

This example shows how to configure the spanning-tree hello time, Forward Delay Timer, and Maximum Age Timer to 2, 4, and 4 seconds:

```
Console> (enable) set spantree hello 2 100
Spantree 100 hello time set to 7 seconds.
Console> (enable)
Console> (enable) set spantree fwddelay 4 100
Spantree 100 forward delay set to 21 seconds.
Console> (enable)
Console> (enable) set spantree maxage 6 100
```

```
Spantree 100 max aging time set to 36 seconds.
Console> (enable)
Console> (enable) set spantree root 1-10 dia 4
VLANs 1-10 bridge priority set to 8192
VLANs 1-10 bridge max aging time set to 14 seconds.
VLANs 1-10 bridge hello time set to 2 seconds.
VLANs 1-10 bridge forward delay set to 9 seconds.
Switch is now the root switch for active VLANs 1-6.
Console> (enable)
```

Using Root Guard—Preventing Switches from Becoming Root

You may want to prevent switches from becoming the root switch. Root guard forces a port to become a designated port so that no switch on the other end of the link can become a root switch.

When you enable root guard on a per-port basis, it is automatically applied to all of the active VLANs to which that port belongs. When you disable root guard, it is disabled for the specified port(s). If a port goes into the root-inconsistent state, it automatically goes into the listening state.

To prevent switches from becoming root, perform this task in privileged mode:

	Task	Command
Step 1	Enable root guard on a port.	**set spantree guard** {**root** | **none**} *mod/port*
Step 2	Verify that root guard is enabled.	**show spantree guard** {*mod/port* | *vlan*} {**mistp-instance** *instance* | *mod/port*}

This example shows how to enable root guard:

```
Console> (enable) set spantree guard root 5/1
Rootguard on port 5/1 is enabled.
Warning!! Enabling rootguard may result in a topolopy change.
Console> (enable)
```

Displaying Spanning-Tree BPDU Statistics

Enter the **show spantree statistics bpdu** command to display the total number of spanning-tree BPDUs (transmitted, received, processed, and dropped). The command also provides the rate of the BPDUs in seconds. The BPDU counters are cleared when you enter the **clear spantree statistics bpdu** command or when the system is booted.

To display the spanning-tree BPDU statistics, perform this task in normal mode (clear the statistics from privileged mode):

	Task	Command
Step 1	Display spanning-tree BPDU statistics.	**show spantree statistics bpdu**
Step 2	Clear the BPDU statistics.	**clear spantree statistics bpdu**

This example shows how to display spanning-tree BPDU statistics:

```
Console> show spantree statistics bpdu
              Transmitted       Received      Processed        Dropped
            -------------- -------------- -------------- --------------

Total            52943073       52016589       52016422            167

Rate(/sec)            989            971            971              0
```

This example shows how to clear spanning-tree BPDU statistics:

```
Console> (enable) clear spantree statistics bpdu
Spanning tree BPDU statistics cleared on the switch.
Console> (enable)
```

Configuring Spanning-Tree Timers on the Switch

The spanning-tree timers affect the spanning-tree performance. You can configure the spanning-tree timers for a VLAN in PVST+ or an MISTP instance in MISTP mode. If you do not specify a VLAN when the switch is in PVST+ mode, VLAN 1 is assumed, or if you do not specify an MISTP instance when the switch is in MISTP mode, MISTP instance 1 is assumed.

These sections describe how to configure the spanning-tree timers:

- Configuring the Hello Time, page 7-50
- Configuring the Forward Delay Time, page 7-50
- Configuring the Maximum Aging Time, page 7-51

Caution Be careful when using these commands. For most situations, we recommend that you use the **set spantree root** and **set spantree root secondary** commands to modify the spanning tree performance parameters.

Table 7-7 describes the switch variables that affect spanning tree performance.

Table 7-7 Spanning-Tree Timers

Variable	Description	Default
Hello Time	Determines how often the switch broadcasts its hello message to other switches.	2 seconds
Maximum Age Timer	Measures the age of the received protocol information that is recorded for a port and ensures that this information is discarded when its age limit exceeds the value of the maximum age parameter that is recorded by the switch. The timeout value is the maximum age parameter of the switches.	20 seconds
Forward Delay Timer	Monitors the time that is spent by a port in the learning and listening states. The timeout value is the forward delay parameter of the switches.	15 seconds

Configuring the Hello Time

Enter the **set spantree hello** command to change the hello time for a VLAN, an MISTP instance, or on a per-port basis. The possible range of *interval* is 1–10 seconds.

To configure the spanning-tree bridge hello time for a VLAN or an MISTP instance, perform this task in privileged mode:

	Task	Command
Step 1	Configure the hello time for a VLAN or an MISTP instance.	**set spantree hello** *interval* {[*vlan*] | **mistp-instance** [*instances*] | **mst** [*mod/port*]}
Step 2	Verify the configuration.	**show spantree** [*vlan* | **mistp-instance** *instances*]

This example shows how to configure the spanning-tree hello time for VLAN 100 to 7 seconds:

```
Console> (enable) set spantree hello 7 100
Spantree 100 hello time set to 7 seconds.
Console> (enable)
```

This example shows how to configure the spanning-tree hello time for an instance to 3 seconds:

```
Console> (enable) set spantree hello 3 mistp-instance 1
Spantree 1 hello time set to 3 seconds.
Console> (enable)
```

This example shows how to configure the spanning-tree hello time for port 4/5 to 4 seconds:

```
Console> (enable) set spantree hello 4 mst 4/1
MST hello time set to 4 on port 4/1.
Console> (enable)
```

Configuring the Forward Delay Time

Enter the **set spantree fwddelay** command to configure the spanning-tree forward delay time for a VLAN. The possible range of *delay* is 4–30 seconds.

To configure the spanning-tree forward delay time for a VLAN, perform this task in privileged mode:

	Task	Command
Step 1	Configure the forward delay time for a VLAN or an MISTP instance.	**set spantree fwddelay** *delay* [*vlan*] **mistp-instance** [*instances*]
Step 2	Verify the configuration.	**show spantree** [*mod/port*] **mistp-instance** [*instances*] [**active**]

This example shows how to configure the spanning-tree forward delay time for VLAN 100 to 21 seconds:

```
Console> (enable) set spantree fwddelay 21 100
Spantree 100 forward delay set to 21 seconds.
Console> (enable)
```

This example shows how to set the bridge forward delay for an instance to 16 seconds:

```
Console> (enable) set spantree fwddelay 16 mistp-instance 1
Instance 1 forward delay set to 16 seconds.
Console> (enable)
```

Configuring the Maximum Aging Time

Enter the **set spantree maxage** command to change the spanning-tree maximum aging time for a VLAN or an instance. The possible range of *agingtime* is 6–40 seconds.

To configure the spanning-tree maximum aging time for a VLAN or an instance, perform this task in privileged mode:

	Task	Command
Step 1	Configure the maximum aging time for a VLAN or an MISTP instance.	**set spantree maxage** *agingtime* [*vlans*] **mistp-instance** *instances*
Step 2	Verify the configuration.	**show spantree** [*mod/port*] **mistp-instance** [*instances*] [**active**]

This example shows how to configure the spanning-tree maximum aging time for VLAN 100 to 36 seconds:

```
Console> (enable) set spantree maxage 36 100
Spantree 100 max aging time set to 36 seconds.
Console> (enable)
```

This example shows how to set the maximum aging time for an instance to 25 seconds:

```
Console> (enable) set spantree maxage 25 mistp-instance 1
Instance 1 max aging time set to 25 seconds.
Console> (enable)
```

Configuring Multiple Spanning Tree on the Switch

These sections describe how to configure MST:

- Enabling Multiple Spanning Tree, page 7-51
- Mapping and Unmapping VLANs to an MST Instance, page 7-58

Enabling Multiple Spanning Tree

To enable and configure MST on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Begin in PVST+ mode.	**set spantree mode pvst+** [**mistp** | **pvst+** | **mistp-pvst+** | **mst**]
Step 2	Display the STP ports.	**show spantree active**
Step 3	Configure the MST region.	**set spantree mst config** {[**name** *name*] | [**revision** *number*] [**commit** | **rollback** | **force**]}

	Task	Command
Step 4	Verify your configuration.	**show spantree mst config**
Step 5	Map VLANs to the MST instance.	**set spantree mst** *instance* **vlan** *vlan*
Step 6	Commit the new region mapping.	**set spantree mst config commit**
Step 7	Enable MST.	**set spantree mode mst** [**mistp** | **pvst+** | **mistp-pvst+** | **mst**]
Step 8	Verify your MST configuration.	**show spantree mst config**
Step 9	Verify your MST instance configuration.	**show spantree mst** *instance*
Step 10	Verify your MST module and port configuration.	**show spantree mst** *mod/port*

These examples show how to enable MST:

```
Console> (enable)
Console> (enable) set spantree mode pvst+
Spantree mode set to PVST+.
Console> (enable) show spantree active
VLAN 1
Spanning tree mode          PVST+
Spanning tree type          ieee
Spanning tree enabled

Designated Root             00-60-70-4c-70-00
Designated Root Priority    16384
Designated Root Cost        19
Designated Root Port        3/48
Root Max Age   14 sec   Hello Time 2  sec   Forward Delay 10 sec

Bridge ID MAC ADDR          00-d0-00-4c-18-00
Bridge ID Priority          32768
Bridge Max Age 20 sec   Hello Time 2  sec   Forward Delay 15 sec

Port                     Vlan Port-State    Cost      Prio Portfast Channel_id
------------------------ ---- ------------- --------- ---- -------- ----------
 3/48                    1    forwarding           19   32 disabled 0
 7/2                     1    forwarding            4   32 enabled  0
Console> (enable) set spantree mst config name cisco revision 1
Edit Buffer modified.
Use 'set spantree mst config commit' to apply the changes
Console> (enable) show spantree mst config
Current (NVRAM) MST Region Configuration:              1 instance
Configuration Name:                                    Revision: 0
Instance VLANs
-------- ---------------------------------------------------------------
   0     1-4094
=============================================================================
NEW MST Region Configuration (Not committed yet)       1 instance
Configuration Name: cisco                              Revision: 1
Instance VLANs
-------- ---------------------------------------------------------------
   0     1-4094
=============================================================================
Edit buffer is locked by: Console (pid 143)
Console> (enable) set spantree mst 1 vlan 2-10
Edit Buffer modified.
Use 'set spantree mst config commit' to apply the changes
Console> (enable) set spantree mst 2 21-30
Usage:set spantree mst <instance> vlan <vlan>
Console> (enable) set spantree mst 2 vlan 21-30
```

```
Edit Buffer modified.
Use 'set spantree mst config commit' to apply the changes
Console> (enable) set spantree mst 3 vlan 31-40
Edit Buffer modified.
Use 'set spantree mst config commit' to apply the changes
Console> (enable) set spantree mst 4 vlan 41-50
Edit Buffer modified.
Use 'set spantree mst config commit' to apply the changes
Console> (enable) show spantree mst config
Current (NVRAM) MST Region Configuration:              1 instance
Configuration Name:                                    Revision: 0
Instance VLANs
-------- ------------------------------------------------------------
   0     1-4094
================================================================================
NEW MST Region Configuration (Not committed yet)       5 instances
Configuration Name: cisco                              Revision: 1
Instance VLANs
-------- ------------------------------------------------------------
   0     1,11-20,51-4094
   1     2-10
   2     21-30
   3     31-40
   4     41-50
================================================================================
Edit buffer is locked by: Console (pid 143)
Console> (enable) set spantree mst config commit
Console> (enable) show spantree mst config
Current (NVRAM) MST Region Configuration:              5 instances
Configuration Name: cisco                              Revision: 1
Instance VLANs
-------- ------------------------------------------------------------
   0     1,11-20,51-4094
   1     2-10
   2     21-30
   3     31-40
   4     41-50
================================================================================
Console> (enable) set spantree mode mst
PVST+ database cleaned up.
Spantree mode set to MST.
Console> (enable) show spantree mst 0
Spanning tree mode          MST
Instance                    0
VLANs Mapped:               1,11-20,51-4094

Designated Root             00-60-70-4c-70-00
Designated Root Priority    16384  (root priority: 16384, sys ID ext: 0)
Designated Root Cost        200000
Designated Root Port        3/48
Root Max Age   14 sec   Forward Delay 10 sec

CIST Regional Root          00-d0-00-4c-18-00
CIST Regional Root Priority 32768
CIST Internal Root Cost     0          Remaining Hops 20

Bridge ID MAC ADDR          00-d0-00-4c-18-00
Bridge ID Priority          32768  (bridge priority: 32768, sys ID ext: 0)
Bridge Max Age 20 sec   Hello Time 2  sec   Forward Delay 15 sec  Max Hops 20

Port                     State         Role Cost      Prio Type
------------------------ ------------- ---- --------- ---- --------------------
 3/48                    forwarding    ROOT    200000   32 Shared, Boundary(STP)
 7/2                     forwarding    DESG     20000   32 P2P, Edge
```

```
Console> (enable) show spantree mst 1
Spanning tree mode          MST
Instance                    1
VLANs Mapped:               2-10

Designated Root             00-00-00-00-00-00
Designated Root Priority    0  (root priority: 0, sys ID ext: 0)
Designated Root Cost        0          Remaining Hops 0
Designated Root Port        1/0

Bridge ID MAC ADDR          00-d0-00-4c-18-00
Bridge ID Priority          32769  (bridge priority: 32768, sys ID ext: 1)

Port                      State         Role Cost      Prio Type
------------------------ ------------ ---- --------- ---- --------------------
Console> (enable) show spantree mst 7/2
Edge Port:          Yes, (Configured) Enable
Link Type:          P2P, (Configured) Auto
Port Guard:    Default
Boundary:            No
Hello:                2, (Local bridge hello: 2)

Inst State         Role Cost      Prio VLANs
---- ------------- ---- --------- ---- ----------------------------------
   0 forwarding    DESG     20000   32 1
Console> (enable) show spantree mst config
Current (NVRAM) MST Region Configuration:              5 instances
Configuration Name: cisco                              Revision: 1
Instance VLANs
-------- ------------------------------------------------------------
   0     1,11-20,51-4094
   1     2-10
   2     21-30
   3     31-40
   4     41-50
==============================================================================
Console> (enable)
```

Configuring the MST Bridge ID Priority

You can set the bridge ID priority for an MST instance when the switch is in MST mode.

The bridge priority value is combined with the system ID extension (the ID of the MST instance) to create the bridge ID priority. You can set 16 possible bridge priority values: 0, 4096, 8192, 12288, 16384, 20480, 24576, 28672, 32768, 36864, 40960, 45056, 49152, 53248, 57344, and 61440.

To configure the bridge ID priority for an MST instance, perform this task in privileged mode:

	Task	Command
Step 1	Configure the bridge ID priority for an MST instance.	**set spantree priority** *bridge_priority* **mst** [*instance*]
Step 2	Verify the bridge ID priority.	**show spantree mst** [*instance* | *mod/port*]

The example shows how to configure the bridge ID priority for an MST instance:

```
Console> (enable) set spantree priority 8192 mst 3
set spantree priority 8192 mst 3
MST instance 3 bridge ID priority set to 8195
(bridge priority: 8192 + sys ID extension: 3)
```

```
Console> (enable) show spantree mst 3
Spanning tree mode          MST
Instance                    3
VLANs Mapped:               31-40

Designated Root             00-00-00-00-00-00
Designated Root Priority    0  (root priority: 0, sys ID ext: 0)
Designated Root Cost        0          Remaining Hops 0
Designated Root Port        1/0

Bridge ID MAC ADDR          00-d0-00-4c-18-00
Bridge ID Priority          8195  (bridge priority: 8192, sys ID ext: 3)

Port                      State         Role Cost      Prio Type
------------------------ ------------- ---- --------- ---- --------------------
6/1                       forwarding    MSTR     2000   32   P2P, Boundary (PVST)
6/2                       blocking      MSTR     2000   32   P2P, Boundary (PVST)
```

Configuring the MST Port Cost

You can configure the port cost of the switch ports. The ports with the lower port costs are more likely to be chosen to forward frames. Assign the lower numbers to the ports that are attached to faster media (such as full duplex) and higher numbers to the ports that are attached to slower media. The possible range of cost is from 1–65535 when using the short method for calculating port cost and from 1–200000000 when using the long method. The default cost differs for different media. For information about calculating the path cost, see the "Calculating and Assigning Port Costs" section on page 7-4.

To configure the port cost for a port, perform this task in privileged mode:

	Task	Command
Step 1	Configure the MST port cost for a switch port.	**set spantree portcost** *mod/port cost* [**mst**]
Step 2	Verify the port cost setting.	**show spantree mst** [*instance* | *mod/port*]

This example shows how to configure the port cost on an MST instance and verify the configuration:

```
Console> (enable) set spantree portcost 6/1 10000 mst
Spantree port 6/1 path cost set to 10000.
Console> (enable)
Console> (enable) show spantree mst 6/1
Edge Port:       No,  (Configured) Default
Link Type:      P2P,  (Configured) Auto
Port Guard:   Default
Boundary:     Yes (PVST)

Inst State         Role Cost      Prio VLANs
---- ------------- ---- --------- ----------------------------------------
   0 forwarding    ROOT     10000   32 1
   1 forwarding    MSTR     10000   32 2-20
   2 forwarding    MSTR     10000   32 21-30
   3 forwarding    MSTR     10000   32 31-40
   4 forwarding    MSTR     10000   32 41-50
Console> (enable)
```

Configuring the MST Port Priority

You can configure the port priority of ports. The port with the lowest priority value forwards the frames for all VLANs. The possible port priority value is a multiple of 16 from 0–240. The default is 32. If all the ports have the same priority value, the port with the lowest port number forwards the frames.

To configure the port priority for a port, perform this task in privileged mode:

	Task	Command
Step 1	Configure the MST port priority for a port.	**set spantree portpri** *mod/port priority* [**mst**]
Step 2	Verify the port priority setting.	**show spantree mst** [*instance* | *mod/port*]

This example shows how to configure the port priority and verify the configuration:

```
Console> (enable) set spantree portpri 6/1 30 mst
Bridge port  6/1 port priority set to 30.
Console> (enable)
Console> (enable) show spantree mst 6/1
Edge Port:       No,  (Configured) Default
Link Type:      P2P,  (Configured) Auto
Port Guard:   Default
Boundary:     Yes (PVST)

Inst State         Role Cost      Prio VLANs
---- ------------- ---- --------- ---------------------------------------
   0 forwarding    ROOT     10000   30 1
   1 forwarding    MSTR     10000   30 2-20
   2 forwarding    MSTR     10000   30 21-30
   3 forwarding    MSTR     10000   30 31-40
   4 forwarding    MSTR     10000   30 41-50
Console> (enable)
```

Configuring the MST Port Instance Cost

You can configure the port instance cost for an instance of MST. The ports with a lower instance cost are more likely to be chosen to forward frames. You should assign lower numbers to the ports that are attached to faster media (such as full duplex) and higher numbers to the ports that are attached to slower media. The default cost differs for different media. The possible value for the port instance cost is 1–268435456.

You can assign a different port instance cost for the instances within a trunk port.

To configure the port instance cost for a port, perform this task in privileged mode:

	Task	Command
Step 1	Configure the MST port instance cost on a port.	**set spantree portinstancecost** *mod/port* [**cost** *cost*] **mst** [*instances*]
Step 2	Verify the path cost for the MST instances on a port.	**show spantree portinstancecost** *mod/port* **mst**

This example shows how to configure the MST port instance cost on a port:

```
Console> (enable) set spantree portinstancecost 4/1 cost 5000 mst 4
Command successful. Modified port 4/1 configuration:
        Cost       Instances
```

```
        ---------- ------------------------------------------------------------
        5000       4
Default 200000     0-3,5-4094
Console> (enable) set spantree portinstancecost 4/1 cost 6000 mst 4000
Command successful. Modified port 4/1 configuration:
        Cost       Instances
        ---------- ------------------------------------------------------------
        5000       4
        6000       4000
Default 200000     0-3,5-3999,4001-4094

Console> (enable) show spantree portinstancecost 4/1
This command is not valid when STP is in MST mode.
Console> (enable) show spantree portinstancecost 4/1 mst
Port 4/1 cost configuration:
        Cost       Instances
        ---------- ------------------------------------------------------------
        5000       4
        6000       4000
Default 200000     0-3,5-3999,4001-4094
Console> (enable)
```

Configuring the MST Port Instance Priority

You can set the port priority for an instance of MST. The port with the lowest priority value for a specific MST instance forwards the frames for that instance. The possible port instance range is 0–240. If all ports have the same priority value for an MST instance, the port with the lowest port priority number forwards the frames for that instance.

You can assign a different port instance priority for instances within a trunk port.

To configure the port instance priority on an MST instance, perform this task in privileged mode:

	Task	Command
Step 1	Configure the port instance priority on an MST instance.	**set spantree portinstancepri** *mod/port priority* **mst** [*instance*]
Step 2	Verify the port instance priority setting.	**show spantree mst** [*instance* | *mod/port*]

This example shows how to configure the port instance priority on an MST instance and verify the configuration:

```
Console> (enable) set spantree portinstancepri 4/1 16 mst 2
Command successful. Modified port 4/1 configuration:
        Priority   Instances
        ---------- ------------------------------------------------------------
        16         2
Default 32         0-1,3-4094
Console> (enable) set spantree portinstancepri 4/1 48 mst 200
Command successful. Modified port 4/1 configuration:
        Priority   Instances
        ---------- ------------------------------------------------------------
        16         2
        48         200
Default 32         0-1,3-199,201-4094
Console> (enable) show spantree mst 4/1
Edge Port:         No, (Configured) Default
Link Type:        P2P, (Configured) Auto
Port Guard:   Default
Boundary:          No
```

```
Hello:               4, (Local port hello:4)

Inst State         Role Cost      Prio VLANs
---- ------------- ---- --------- ---- ----------------------------------
   0 forwarding    DESG    200000   32 None
   2 forwarding    DESG    200000   16 1
 200 forwarding    DESG    200000   48 2
Console> (enable)
```

Mapping and Unmapping VLANs to an MST Instance

By default, all VLANs are mapped to IST (instance 0). For an MST instance (MSTI) 1–15 to be active, you must map at least one VLAN to that MSTI. IST will always be active whether VLANs are mapped to IST or not. MST has separate regions, which prevents VLAN mapping conflicts. Follow these guidelines for mapping and unmapping VLANs to an MST instance:

Note See Chapter 11, "Configuring VLANs" for details on using and configuring VLANs.

- You can map only Ethernet VLANs to MST instances.
- At least one VLAN in the instance must have an active port in order for MST to be active.
- You can map as many Ethernet VLANs as you wish to an MST instance.
- You cannot map a VLAN to more than one MST instance.
- The Hello Time, Maximum Age timer, and Forward Delay timer set for mode and all spanning trees are used globally by MST.

Note To use VLANs 1025–4094, you must enable MAC address reduction. See the "Creating Extended-Range VLANs" section on page 11-7 in Chapter 11, "Configuring VLANs" for details on using extended-range VLANs.

To map a VLAN to an MST instance, perform this task in privileged mode:

	Task	Command
Step 1	Map a VLAN to an MST instance.	**set spantree mst** *instance* **vlan** *vlan*
Step 2	Make the new region mapping effective.	**set spantree mst config commit**
Step 3	Verify that the VLAN is mapped.	**show spantree mst** [*instance*] [**active**] *mod/port*

This example shows how to map a VLAN to MST instance 1 and verify the mapping:

```
Console> (enable) show spantree mst config
Current (NVRAM) MST Region Configuration:             3 instances
Configuration Name:arthur                             Revision:23703
Instance VLANs
-------- ---------------------------------------------------------------
   0     1,31-4094
   2     2-20
   3     21-30
================================================================================
Console> (enable) set spantree mst 1400 vlan 900-999
Edit Buffer modified.
Use 'set spantree mst config commit' to apply the changes
```

```
Console> (enable) show spantree mst config
Current (NVRAM) MST Region Configuration:             3 instances
Configuration Name:arthur                             Revision:23703
Instance VLANs
-------- ---------------------------------------------------------------
   0     1,31-4094
   2     2-20
   3     21-30
================================================================================
NEW MST Region Configuration (Not committed yet)       4 instances
Configuration Name:arthur                             Revision:23703
Instance VLANs
-------- ---------------------------------------------------------------
   0     1,31-899,1000-4094
   2     2-20
   3     21-30
1400     900-999
================================================================================
Edit buffer is locked by:Console (pid 143)
Console> (enable) clear spantree mst 1400 vlan 900-998
Edit Buffer modified.
Use 'set spantree mst config commit' to apply the changes

Console> (enable) set spantree mst config commit
Console> (enable) show spantree mst config
Current (NVRAM) MST Region Configuration:             4 instances
Configuration Name:arthur                             Revision:23703
Instance VLANs
-------- ---------------------------------------------------------------
   0     1,31-998,1000-4094
   2     2-20
   3     21-30
1400     999
================================================================================
Console> (enable)
```

Configuring BPDU Skewing on the Switch

Commands that support the spanning-tree BPDU skewing allow you to perform these functions:

- Enable or disable BPDU skewing. The default is disabled.
- Modify the **show spantree summary** output to show if the skew detection is enabled and for which VLANs or PVST+ or MISTP instances the skew was detected.
- Provide a display of the VLAN, PVST+, or MISTP instance and the port that is affected by the skew, including this information:
  - The last skew duration (in absolute time)
  - The worst skew duration (in absolute time)
  - The date and time of the worst duration

To change how spanning tree performs BPDU skewing statistics gathering, enter the **set spantree bpdu-skewing** command. The **bpdu-skewing** command is disabled by default.

To configure the BPDU skewing statistics gathering for a VLAN, perform this task in privileged mode:

	Task	Command
Step 1	Configure BPDU skewing.	**set spantree bpdu-skewing** [**enable** | **disable**]
Step 2	Verify the configuration.	**show spantree bpdu-skewing** *vlan* [*mod/port*] **show spantree bpdu-skewing mistp-instance** [*instance*] [*mod/port*]

This example shows how to configure BPDU skewing and display the skewing statistics:

```
Console> (enable) set spantree bpdu-skewing
Usage:set spantree bpdu-skewing <enable|disable>
Console> (enable) set spantree bpdu-skewing enable
Spantree bpdu-skewing enabled on this switch.
Console> (enable)

Console> (enable) show spantree bpdu-skewing 1
Bpdu skewing statistics for vlan 1
Port    Last Skew ms   Worst Skew ms     Worst Skew Time
------  -------------  -------------  ------------------------
8/2              5869         108370  Tue Nov 21 2000, 06:25:59
8/4              4050         113198  Tue Nov 21 2000, 06:26:04
8/6            113363         113363  Tue Nov 21 2000, 06:26:05
8/8              4111         113441  Tue Nov 21 2000, 06:26:05
8/10           113522         113522  Tue Nov 21 2000, 06:26:05
8/12             4111         113600  Tue Nov 21 2000, 06:26:05
8/14           113678         113678  Tue Nov 21 2000, 06:26:05
8/16             4111         113755  Tue Nov 21 2000, 06:26:05
8/18           113833         113833  Tue Nov 21 2000, 06:26:05
8/20             4111         113913  Tue Nov 21 2000, 06:26:05
8/22           113917         113917  Tue Nov 21 2000, 06:26:05
8/24             4110         113922  Tue Nov 21 2000, 06:26:05
8/26           113926         113926  Tue Nov 21 2000, 06:26:05
8/28             4111         113931  Tue Nov 21 2000, 06:26:05
Console> (enable)
```

This example shows how to configure BPDU skewing for VLAN 1 on module 8, port 2 and display the skewing statistics:

```
Console> (enable) show spantree bpdu-skewing 1 8/4
Bpdu skewing statistics for vlan 1
Port    Last Skew ms   Worst Skew ms     Worst Skew Time
------  -------------  -------------  ------------------------
8/4              5869         108370  Tue Nov 21 2000, 06:25:59
```

You will receive a similar output when MISTP is running.

The **show spantree summary** command displays if BPDU skew detection is enabled and also lists the VLANs or instances that are affected in the skew. This example shows the output when using the **show spantree summary** command:

```
Console> (enable) show spantree summary
Root switch for vlans: 1
BPDU skewing detection enabled for the bridge
BPDU skewed for vlans: 1
Portfast bpdu-guard disabled for bridge.
Portfast bpdu-filter disabled for bridge.
Uplinkfast disabled for bridge.
Backbonefast disabled for bridge.

Summary of connected spanning tree ports by vlan
```

```
VLAN  Blocking Listening Learning Forwarding STP Active
----- -------- --------- -------- ---------- ----------
    1        6         4        2          0         12

      Blocking Listening Learning Forwarding STP Active
----- -------- --------- -------- ---------- ----------
Total       6         4        2          0         12
Console> (enable)
```

Configuring Layer 2 PDU Rate Limiting on the Switch

Note This feature is only supported with PFC3A or later PFC.

Note This feature does not work in truncated mode.

You can use the Layer 2 PDU rate limiters to limit the number of packets to a normal rate and to avoid abnormal incoming rates.

The commands that support Layer 2 PDU rate limiting allow you to perform these functions:

- Enable, disable, or set rate limiting for the spanning-tree BPDUs—IEEE and PVST/Shared Spanning Tree Protocol (SSTP), Cisco Discovery Protocol (CDP), Dynamic Trunking Protocol (DTP), UniDirectional Link Detection (UDLD), VLAN Trunking Protocol (VTP), Link Aggregation Control Protocol (LACP), and Port Aggregation Protocol (PAgP)—globally on the switch.
- Enable, disable, or set rate limiting for the Layer 2 protocol tunnel-encapsulated PDUs globally on the switch.
- Enable, disable, or set the 802.1X port security rate limiters globally on the switch.

All three types of rate limiters work independently of each other.

To enable or disable Layer 2 PDU rate limiting, enter the **set rate-limit** {**l2pdu** | **l2port-security** | **l2protocol-tunnel**} {**enable** | **disable**} command. Layer 2 PDU rate limiting is disabled by default.

To configure Layer 2 PDU rate limiting, perform this task in privileged mode:

	Task	Command
Step 1	Enable Layer 2 PDU rate limiting.	**set rate-limit** {**l2pdu** | **l2port-security** | **l2protocol-tunnel**} **enable**
Step 2	Set the rate limiter value.	**set rate-limit** {**l2pdu** | **l2port-security** | **l2protocol-tunnel**} **rate** *rate*
Step 3	Verify the configuration.	**show rate-limit** **show rate-limit config**

Use the **l2pdu** keyword for rate limiting the Layer 2 protocol packets including the following:

- Spanning-tree IEEE—destination MAC address 01-80-c2-00-00-00
- PVST/SSTP—destination MAC address 01-00-0C-CC-CC-CD

- CDP/DTP/UDLD/LACP/PAgP/VTP—destination MAC address 01-00-0C-CC-CC-CC

Note Rate limiting Layer 2 protocols works as follows: 1) Frames are classified as Layer 2 control frames by the destination MAC address (listed above). 2) The software allocates an LTL index for these frames. 3) The LTL index is submitted to the forwarding engine for (aggregate) rate limiting of all the associated frames.

Use the **l2port-security** keyword for rate limiting the Layer 2 802.1X port security packets.

Use the **l2protocol-tunnel** keyword for rate limiting the Layer 2 protocol tunnel-encapsulated packets with the MAC address (01-00-0C-CD-CD-D0).

This example shows how to enable Layer 2 rate limiting, set the rate limiter value, and verify the configuration:

```
Console>(enable) set rate-limit l2pdu enable
Layer 2 rate limiter for PDUs enabled on the switch.
Console>(enable)

Console>(enable) set rate-limit l2pdu rate 1000
Layer 2 rate limiter for PDU rate set to 1000.
Console>(enable)

Console>(enable) set rate-limit l2protocol-tunnel disable
Layer 2 rate limiter for l2protocol-tunnel disabled on the switch.
Console>(enable)

Console>(enable) show rate-limit
Configured Rate Limiter Settings:
Rate Limiter Type    Status  Rate (pps)     Burst
-------------------- ------  -------------- -----
VACL LOG             On      2500           1
ARP INSPECTION       On      500            1
L2 PDU               On      1000           1
L2 PROTOCOL TUNNEL   On      1000           1
L2 PORT SECURITY     On      1000           1
MCAST NON RPF        Off     *              *
MCAST DFLT ADJ       Off     *              *
MCAST DIRECT CON     Off     *              *
ACL INGRESS BRIDGE   Off     *              *
ACL EGRESS BRIDGE    Off     *              *
L3 SEC FEATURES      Off     *              *
FIB RECEIVE          Off     *              *
FIB GLEAN            Off     *              *
MCAST PARTIAL SC     Off     *              *
RPF FAIL             Off     *              *
TTL FAIL             Off     *              *
NO ROUTE             Off     *              *
ICMP UNREACHABLE     Off     *              *
ICMP REDRECT         Off     *              *
MTU FAIL             Off     *              *
Console>(enable)
```

This example shows how to display the Layer 2 rate-limiter administrative and operation status information:

```
Console> show rate-limit config
Rate Limiter Type    Admin Status Oper Status
-------------------- ------------ -----------
l2pdu                On           On
l2protocol-tunnel    On           On
l2port-security      On           On
Console>
```

CHAPTER 8

Configuring IEEE 802.1Q Tunneling and Layer 2 Protocol Tunneling

This chapter describes how to configure IEEE 802.1Q tunneling and Layer 2 protocol tunneling on the Catalyst 6500 series switches.

This chapter consists of these sections:

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

Understanding How 802.1Q Tunneling Works

802.1Q tunneling enables the service providers to use a single VLAN to support the customers who have multiple VLANs, while preserving the customer VLAN IDs and keeping traffic in the different customer VLANs segregated.

A port that is configured to support 802.1Q tunneling is called a tunnel port. When you configure tunneling, you assign a tunnel port to a VLAN that you dedicate to tunneling. To keep the customer traffic segregated, each customer requires a separate VLAN, but that one VLAN supports all of the customer's VLANs.

With 802.1Q tunneling, tagged traffic comes from an 802.1Q trunk port on a customer device and enters the switch through a tunnel port. The link between the 802.1Q trunk port on a customer device and the tunnel port is called an asymmetrical link because one end is configured as an 802.1Q trunk port and the other end is configured as a tunnel port.

When a tunnel port receives the tagged customer traffic from an 802.1Q trunk port, it does not strip the received 802.1Q tag from the frame header; instead, the tunnel port leaves the 802.1Q tag intact, adds a 2-byte EtherType field (0x8100) and a 2-byte length field, and puts the received customer traffic into the VLAN to which the tunnel port is assigned. This EtherType 0x8100 traffic, with the received 802.1Q tag intact, is called tunnel traffic.

A VLAN that carries tunnel traffic is an 802.1Q tunnel. The tunnel ports in the VLAN are the tunnel's ingress and egress points.

The tunnel ports do not have to be on the same network device. The tunnel can cross the other network links and the other network devices before reaching the egress tunnel port. A tunnel can have as many tunnel ports as required to support the customer devices that need to communicate through the tunnel.

An egress tunnel port strips the 2-byte EtherType field (0x8100) and the 2-byte length field and transmits the traffic with the 802.1Q tag still intact to an 802.1Q trunk port on a customer device. The 802.1Q trunk port on the customer device strips the 802.1Q tag and puts the traffic into the appropriate customer VLAN.

Not all switches support the standard 2-byte EtherType field (0x8100). If your switch does not support the 2-byte EtherType field, you can connect the switch to a Gigabit Interface Converter (GBIC) or 10-Gigabit port and separate untagged IP traffic from the IP management traffic with a specified EtherType. The untagged IP traffic is automatically assigned to the native VLAN, and the traffic with the specified EtherType is switched to a specified VLAN.

802.1Q Tunneling Configuration Guidelines

This section provides the guidelines for configuring 802.1Q tunneling in your network:

- Use asymmetrical links to put traffic into a tunnel or to remove traffic from a tunnel.
- Configure tunnel ports only to form an asymmetrical link.
- Dedicate one VLAN for each tunnel.
- Assign tunnel ports only to VLANs that are used for tunneling.
- Trunks require no special configuration to carry tunnel VLANs.
- We recommend that you use ISL trunks to carry tunnel traffic between devices that do not have tunnel ports. Because of the 802.1Q native VLAN feature, using 802.1Q trunks requires that you be very careful when you configure tunneling. A mistake might direct tunnel traffic to a nontunnel port.
- Because tunnel traffic retains the 802.1Q tag within the switch, the Layer 2 frame header length imposes the following restrictions:
  - The Layer 3 packet within the Layer 2 frame cannot be identified.
  - Layer 3 and higher parameters are not identifiable in tunnel traffic (for example, Layer 3 destination and source addresses).
  - Tunnel traffic cannot be routed.
  - The switch can filter tunnel traffic using only Layer 2 parameters (VLANs and source and destination MAC addresses).
  - The switch can provide only MAC-layer quality of service (QoS) for tunnel traffic.
  - QoS cannot detect the received class of service (CoS) value in the 802.1Q 2-byte Tag Control Information field.

- Asymmetrical links do not support the Dynamic Trunking Protocol (DTP), because only one port on the link is a trunk. Configure the 802.1Q trunk port on an asymmetrical link with the **nonegotiate dot1q** trunking keywords.
- Ensure that the native VLAN of the 802.1Q trunk port in an asymmetrical link carries no traffic. Because traffic in the native VLAN is untagged, it cannot be tunneled correctly. Alternatively, you can enter the global **set dot1q-all-tagged enable** command to ensure that egress traffic in the native VLAN is tagged with 802.1Q tags.

Note See Chapter 5, "Configuring Ethernet VLAN Trunks," for information on using the global **set dot1q-all-tagged enable** command.

- On an asymmetrical link, the Cisco Discovery Protocol (CDP) reports a native VLAN mismatch if the VLAN of the tunnel port does not match the native VLAN of the 802.1Q trunk. The 802.1Q tunnel feature does not require that the VLANs match. Ignore the messages if your configuration requires nonmatching VLANs.
- You can tunnel jumbo frames if the jumbo frame length combined with the 802.1Q tag does not exceed the maximum frame size.

Note To set the correct maximum transmission unit (MTU) size, you must enable jumbo frames on all ports that carry 802.1 tunnel traffic.

- You cannot configure 802.1Q tunneling on ports that are configured to support the following:
  - Private VLANs
  - Voice over IP (Cisco IP Phone 7960)
- The following Layer 2 protocols work between devices that are connected by an asymmetrical link:
  - CDP
  - UniDirectional Link Detection (UDLD)
  - Port Aggregation Protocol (PAgP)
- VLAN Trunking Protocol (VTP) does not work between the following devices:
  - Devices that are connected by an asymmetrical link
  - Devices communicating through a tunnel

Note To configure an EtherChannel as an asymmetrical link, all ports in the EtherChannel must have the same tunneling configuration. Because the Layer 3 packet within the Layer 2 frame cannot be identified, configure the EtherChannel to use MAC-address-based frame distribution.

- Generic Attribute Registration Protocol (GARP) VLAN Registration Protocol (GVRP) works between devices that communicate through a tunnel but does not work between devices that are connected by an asymmetrical link.
- An interconnected network cannot have redundant paths to two different edge switches in an ISP. An interconnected network can have redundant paths to the same edge switch in an ISP, but the customer network must use Per VLAN Spanning Tree + (PVST+); it cannot be configured for Multi-Instance Spanning Tree Protocol (MISTP) or Multiple Spanning Tree (MST). The ISP infrastructure must use either PVST+, MISTP-PVST+, or MST-PVST+.

Configuring 802.1Q Tunneling on the Switch

These sections describe how to configure 802.1Q tunneling:

Note See Chapter 5, "Configuring Ethernet VLAN Trunks," for information on using the global **set dot1q-all-tagged enable** command.

Configuring 802.1Q Tunnel Ports

Caution When you configure tunneling in any VLAN, make sure that you configure only the appropriate tunnel ports and that you use one VLAN for each tunnel. Incorrect assignment of tunnel ports to VLANs can cause traffic forwarding problems.

To configure 802.1Q tunneling on a port, perform this task in privileged mode:

	Task	Command
Step 1	Configure tunneling on a port.	**set port dot1qtunnel** {**all** | *mod/port* **access** | **disable**}
Step 2	Verify the configuration.	**show port dot1qtunnel** [*mod*[*/port*]]

This example shows how to configure tunneling on port 4/1 and verify the configuration:

```
Console> (enable) set port dot1qtunnel 4/1 access
Dot1q tunnel feature set to access mode on port 4/1.
Port 4/1 trunk mode set to off.
Console> (enable) show port dot1qtunnel 4/1
Port   Dot1q tunnel mode
-----  -----------------
 4/1   access
```

Clearing 802.1Q Tunnel Ports

To clear 802.1Q tunneling support from a port, perform this task in privileged mode:

	Task	Command
Step 1	Clear tunneling from a port.	**set port dot1qtunnel** {*mod/port*} **disable**
Step 2	Verify the configuration.	**show port dot1qtunnel** [*mod*[*/port*]]

This example shows how to clear tunneling on port 4/1 and verify the configuration:

```
Console> (enable) set port dot1qtunnel 4/1 disable
Dot1q tunnel feature disabled on port 4/1.
Console> (enable) show port dot1qtunnel 4/1
Port   Dot1q tunnel mode
-----  -----------------
 4/1   disabled
```

Disabling Global Support for 802.1Q Tunneling

The **set port dot1qtunnel all disable** command is the only command that is required to clear 802.1Q tunneling from the port. You do not need to enter the **set dot1q-all-tagged disable** command to clear 802.1Q tunneling.

To disable global support for 802.1Q tunneling on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Disable global tunneling support on the switch.	**set port dot1qtunnel all disable**
Step 2	Verify the configuration.	**show port dot1qtunnel**

This example shows how to disable tunneling support on the switch and verify the configuration:

```
Console> (enable) set port dot1qtunnel all disable
Dot1q tunnel feature disabled on all applicable ports.
Console> (enable) show port dot1qtunnel
Port   Dot1q tunnel mode
-----  -----------------
 2/1   disabled
 2/2   disabled
 3/1   disabled
 3/2   disabled
 3/3   disabled
 3/4   disabled
 3/5   disabled
 3/6   disabled
 3/7   disabled
 3/8   disabled
 3/9   disabled
 3/10  disabled
 3/11  disabled
 3/12  disabled
 3/13  disabled
 3/14  disabled
 3/15  disabled
 3/16  disabled
<output truncated>
```

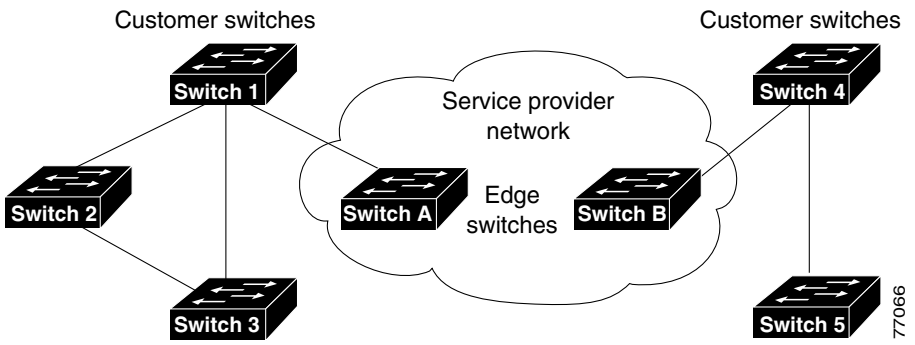
Understanding How Layer 2 Protocol Tunneling Works

Layer 2 protocol tunneling allows the protocol data units (PDUs) (CDP, STP, and VTP) to be tunneled through a network. Some terminology that is used in this section is defined as follows:

- Edge switch—The switch that is connected to the customer switch and placed on the boundary of the service provider network (see Figure 8-1).
- Layer 2 protocol tunnel port—A port on the edge switch on which a specific tunneled protocol can be encapsulated or deencapsulated. The tunnel port is configured through CLI commands.
- Tunneled PDU—A CDP, STP, or VTP PDU.

In the current implementation of 802.1Q tunneling, spanning-tree BPDUs are flooded only on the special 802.1Q tunnel ports that belong to the same edge switch. This implementation prevents loops between the edge switch and the customer switch at each site. The BPDUs are not flooded on the ports that are connected to other service provider switches inside the service provider network. This handling of the BPDUs creates different spanning-tree domains (different spanning-tree roots) for the customer network. For example, STP for a VLAN on switch 1 (see Figure 8-1) builds a spanning-tree topology on Switches 1, 2, and 3 without considering the convergence parameters that are based on Switches 4 and 5. To provide a single spanning-tree domain for the customer, a generic scheme to tunnel BPDUs was created for control protocol PDUs (CDP, STP, and VTP). This process is referred to as Layer 2 protocol tunneling.

Figure 8-1 Layer 2 Protocol Tunneling Network Configuration



Layer 2 protocol tunneling provides a scalable approach to PDU tunneling by software encapsulating the PDUs in the ingress edge switches and then multicasting them in hardware. All switches inside the service provider network treat these encapsulated frames as data packets and forward them to the other end. The egress edge switch listens for these special encapsulated frames and deencapsulates them; they are then forwarded out of the tunnel.

The encapsulation rewrites the destination Media Access Control (MAC) address in the PDU. An ingress edge switch rewrites the destination MAC address of the PDUs that are received on a tunneled port with the Cisco proprietary multicast address (01-00-0c-cd-cd-d0). The PDU is then flooded to the native VLAN of the tunneled port. If you enable Layer 2 protocol tunneling on a port, the PDUs of an enabled protocol are not sent out. If you disable Layer 2 protocol tunneling on a port, the disabled protocols behave the same way that they behaved before Layer 2 protocol tunneling was disabled on the port.

Layer 2 Protocol Tunneling Configuration Guidelines

This section provides the guidelines for configuring protocol tunneling in your network:

- The protocol tunneling functions independently from 802.1Q tunneling.
- For performance reasons, we do not recommend that you configure Layer 2 protocol tunneling in systems with a Supervisor Engine 1.
- You can enable Layer 2 protocol tunneling on access ports, trunk ports, or 802.1Q tunneling ports.
- Layer 2 protocol tunneling is not supported with private VLANs.
- Layer 2 protocol tunneling is not supported with dynamic VLANs.
- If you are running MST and connecting to an ISP network using EtherChannels, you must set the link type to **shared** on all the channeling ports by using the **set spantree link-type** *mod/port* **shared** command. This command prevents the EtherChannels from going into the errdisable state because of channel misconfiguration.
- With a PFC3A, you can enter the **set rate-limit l2protocol-tunnel** commands to enable, disable, or set rate limiting for the Layer 2 protocol tunnel-encapsulated PDUs globally on the switch. For detailed information on configuring rate limiting, see the "Configuring Layer 2 PDU Rate Limiting on the Switch" section on page 7-61.

Configuring Layer 2 Protocol Tunneling on the Switch

These sections describe the protocol tunneling configuration:

Specifying a Layer 2 Protocol

To specify a Layer 2 protocol on a port or range of ports, perform this task in privileged mode:

	Task	Command
Step 1	Specify a Layer 2 protocol on a port.	**set port l2protocol-tunnel** *mod/port* {**cdp** | **eoam** | **stp** | **vtp**} {**enable** | **disable**}
Step 2	Verify the configuration.	**show l2protocol-tunnel statistics** [*mod*[*/port*]]

This example shows how to specify a Layer 2 protocol on a port and verify the configuration:

Note You can specify more than one protocol type at a time. In the CLI, separate protocol types with a space.

```
Console> (enable) set port l2protocol-tunnel 3/15 cdp enable
Layer 2 protocol tunneling enabled for CDP on port 3/15.
Port 3/15 trunk mode set to off.

Console> (enable) set port l2protocol-tunnel 3/15 cdp disable
Layer 2 protocol tunneling disabled for CDP on port 3/15.

Console> (enable) set port l2protocol-tunnel 3/15 cdp stp vtp enable
Layer 2 protocol tunneling enabled for CDP STP VTP on port 3/15.
Port 3/15 trunk mode set to off.

Console> (enable) show l2protocol-tunnel statistics 3/15
Tunneling CoS is set to 5.

Port                     CDP Frames Encap     CDP Frames De-encap
------------------------ -------------------- --------------------
 3/15                                  97465                94434

Port                     STP Frames Encap     STP Frames De-encap
------------------------ -------------------- --------------------
 3/15                                  67465                34434

Port                     VTP Frames Encap     VTP Frames De-encap
------------------------ -------------------- --------------------
 3/15                                   1212                 1213
Console> (enable)
```

Configuring Layer 2 Protocol Tunneling on Trunk Ports

Layer 2 protocol tunneling on trunks allows third-party vendors' equipment to interoperate with the Catalyst 6500 series switch in service-provider networks. Layer 2 protocol tunneling makes control protocol PDUs such as STP, CDP, and VTP, transparent to the service provider cloud when passing traffic through trunk ports. Because of some interoperability issues with other vendors' switches, you cannot achieve true transparent LAN service (TLS) or enable 802.1Q tunneling when using third-party switches. In earlier releases, Layer 2 protocol tunneling was available on access ports only.

Note The service providers should not allow customers to connect directly to trunks on which Layer 2 protocol tunneling is enabled.

Follow the guidelines in the "Layer 2 Protocol Tunneling Configuration Guidelines" section on page 8-7. Additionally, you cannot configure 802.1Q tunneling on the trunk ports; however, 802.1Q tunneling can be tunneled through the trunk ports.

Note If you have a mixed network environment that is using both 802.1Q tunneling and Layer 2 protocol tunneling, you must double-tag packets in order to interoperate with third-party equipment.

To enable or disable Layer 2 protocol tunneling on a trunk port or a range of trunk ports, perform this task in privileged mode:

Task	Command
Enable or disable Layer 2 protocol tunneling on a trunk.	**set l2protocol-tunnel trunk** {**enable** | **disable**}

Note Do not configure (enable or disable) Layer 2 protocol tunneling on trunks when active Layer 2 protocol tunnels are already configured. If you plan to configure Layer 2 protocol tunneling on trunks, make sure that you do so before performing any other Layer 2 protocol tunneling tasks.

This example shows how to enable Layer 2 protocol tunneling on a trunk:

```
Console> (enable) set l2protocol-tunnel trunk enable
Layer 2 Protocol Tunnel on trunks is allowed.
Console> (enable)
```

This example shows how to disable Layer 2 protocol tunneling on a trunk:

```
Console> (enable) set l2protocol-tunnel trunk disable
Warning!! Clear any layer 2 protocol tunnel configuration on trunks
before using this command.
Layer 2 Protocol Tunnel on trunks is not allowed.
Console> (enable)
```

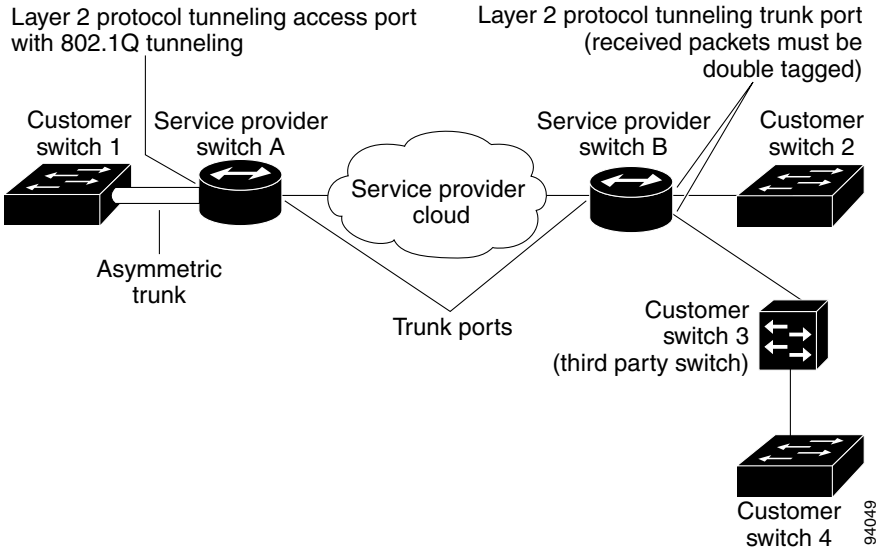
Layer 2 Protocol Tunneling on Trunks Example

The example in Figure 8-2 shows a service provider network that includes Layer 2 protocol tunneling ports (nontrunk) with 802.1Q tunneling configured and two trunk ports with Layer 2 protocol tunneling configured.

Service provider A sends double-tagged encapsulated packets through the service provider cloud with the expectation that the packets will be received in the same double-tagged format on the other end. If customer switch 2 and customer switch 3 send single-tagged packets to service provider B, there is no way to identify the VLAN at egress at service provider A. However, if all switches are sending double-tagged packets, service provider A can correctly tunnel the packets at egress. To achieve correct results, all packets that are received on Layer 2 protocol tunneling trunk ports must be double tagged.

Another example is when a customer wants to tunnel CDP and VTP packets. The CDP/VTP packets are received by a Catalyst 6500 series switch from a third-party switch that is tunneled from other Cisco switches. If the service provider wants to support multiple customers, the service provider must tunnel CDP and VTP packets on a VLAN other than VLAN 1 because Catalyst 6500 series switches use VLAN 1 for transmitting CDP and VTP packets. Because the third-party switches should not directly connect to Layer 2 protocol tunneling trunk ports, one of the third-party switches needs to do VLAN translation or VLAN tagging to ensure that packets are tunneled on the correct VLAN.

Figure 8-2 Layer 2 Protocol Tunneling on Trunks Network Example



Specifying Drop and Shutdown Thresholds on Layer 2 Protocol Tunneling Ports

The shutdown threshold provides a type of rate limiting that prevents the edge switch from being overwhelmed by attached customer switches. We recommend that you configure a shutdown threshold value whenever you use Layer 2 protocol tunneling ports with 802.1Q tunneling.

We recommend that the maximum value for the shutdown threshold is 1000. This value reflects the number of PDUs that an edge switch can handle per second (without dropping any) while performing egress and ingress tunneling. For an edge switch, the shutdown threshold value determines the number of Layer 2 protocol tunneling ports that can be connected to customer switches and the number of customer VLANs per Layer 2 protocol tunneling port. In determining the recommended maximum value of 1000, egress tunneling from the service provider network was also taken into consideration.

To determine the number of Layer 2 protocol tunneling ports (links) and the number of customer VLANs per Layer 2 protocol tunneling port (VLANs per link) that an edge switch can handle, multiply the number of Layer 2 protocol tunneling ports by the number of VLANs. The result should be less than or equal to 1000. Some acceptable configurations are as follows:

- 1 Layer 2 protocol tunneling port x 1000 VLANs
- 2 Layer 2 protocol tunneling ports x 500 VLANs
- 5 Layer 2 protocol tunneling ports x 200 VLANs
- 10 Layer 2 protocol tunneling ports x 100 VLANs
- 20 Layer 2 protocol tunneling ports x 50 VLANs
- 100 Layer 2 protocol tunneling ports x 10 VLANs

Note After reaching the shutdown threshold factor, the port or range of ports goes into the errdisable state and is restored after the errdisable timeout interval. The shutdown threshold factor should exceed the drop threshold factor. After reaching the drop threshold factor, the port or range of ports starts dropping PDUs.

The default for the drop threshold and the shutdown threshold is zero (0). A zero indicates that no limit is set.

Note With software release 8.4(1) and later releases, you can specify the drop and shutdown thresholds for individual protocols on a per-port basis. If you configure thresholds only and do not specify a protocol, the packets are rate limited cumulatively irrespective of protocols. If you specify a threshold for a protocol on a port, the packets are rate limited on a cumulative basis and then per-protocol thresholds are applied to the packets. The range for the per-port protocols drop threshold and shutdown threshold is from 0–65535.

To specify the drop and shutdown thresholds on a port, perform this task in privileged mode:

	Task	Command
Step 1	Specify the drop and shutdown thresholds on a port.	**set port l2protocol-tunnel** *mod/port* {**drop-threshold** *drop-threshold*} {**shutdown-threshold** *shutdown-threshold*} [**cdp** | **eoam** | **stp** | **vtp**]
Step 2	Verify the configuration.	**show port l2protocol-tunnel** [*mod*[*/port*]]

This example shows how to specify a drop threshold of 500 and a shutdown threshold of 1000 on a port:

```
Console> (enable) set port l2protocol-tunnel 3/15 drop-threshold 500 shutdown-threshold
1000
Drop Threshold=500, Shutdown Threshold=1000 set on port 3/15.
Console> (enable)
```

This example shows how to specify a drop threshold of 100 and a shutdown threshold of 400 for the CDP packets on a port:

```
Console> (enable) set port l2protocol-tunnel 3/1 drop-threshold 200 shutdown-threshold 400
cdp
Drop Threshold=200, Shutdown Threshold=400 set on port 3/1.
Console> (enable)

Console> (enable) show port l2protocol-tunnel 3/15
Port                      Tunnel Protocol(s) Drop Threshold Shutdown Threshold
------------------------- ------------------ -------------- ------------------
 3/15                     None                          500               1000

Port                 CDP       CDP       STP       STP       VTP       VTP
                     Drop      Shutdown  Drop      Shutdown  Drop      Shutdown
                     Threshold Threshold Threshold Threshold Threshold Threshold
-------------------- --------- --------- --------- --------- --------- ---------
 3/15                        0         0         0         0         0         0
Console> (enable)
```

```
Console> (enable) show port l2protocol-tunnel 3/1
Port                      Tunnel Protocol(s) Drop Threshold Shutdown Threshold
------------------------- ------------------ -------------- ------------------
 3/1                      None                            0                  0

Port                 CDP       CDP       STP       STP       VTP       VTP
                     Drop      Shutdown  Drop      Shutdown  Drop      Shutdown
                     Threshold Threshold Threshold Threshold Threshold Threshold
-------------------- --------- --------- --------- --------- --------- ---------
 3/1                       200       400         0         0         0         0
Console> (enable)
```

Specifying CoS on Layer 2 Protocol Tunneling Ports

You can specify a class of service (CoS) value globally on all the ingress Layer 2 protocol tunneling ports. Because the CoS value applies to all the ingress tunneling ports, all the encapsulated PDUs that are sent out by the switch have the same CoS value. The valid values are from 0–7, and the default CoS is 5.

To specify a CoS value globally on all the ingress Layer 2 protocol tunneling ports, perform this task in privileged mode:

	Task	Command
Step 1	Globally specify a CoS value.	**set l2protocol-tunnel cos** *cos-value*
Step 2	Verify the configuration.	**show l2protocol-tunnel statistics** [*mod*[*/port*]]

This example shows how to set the CoS value to 6:

```
Console> (enable) set l2protocol-tunnel cos 6
New CoS value is 6.
Console> (enable)

Console> (enable) show l2protocol-tunnel statistics 4/1
Tunneling CoS is set to 6.
Port  CDP Frames Encap     CDP Frames De-encap
----- -------------------- --------------------
 4/1                 97465                94434
Console> (enable)

Console> (enable) clear l2protocol-tunnel cos
Default Cos set to 5.
Console> (enable)
```

Clearing Layer 2 Protocol Tunneling Statistics

To clear the Layer 2 protocol tunneling statistics on a port or on all the tunneling ports, perform this task in privileged mode:

Task	Command
Clear Layer 2 tunnel port statistics.	**clear l2protocol-tunnel statistics** [*mod/port*]

This example shows how to clear the Layer 2 tunnel port statistics on port 7/1:

```
Console> (enable) clear l2protocol-tunnel statistics 7/1
Layer 2 Protocol Tunnel statistics cleared on ports: 7/1.
Console> (enable)
```

CHAPTER 9

Configuring Spanning-Tree PortFast, UplinkFast, BackboneFast, and Loop Guard

This chapter describes how to configure the spanning-tree PortFast, UplinkFast, BackboneFast, and loop guard features on the Catalyst 6500 series switches.

Note For information on configuring the Spanning Tree Protocol (STP), see Chapter 7, "Configuring Spanning Tree."

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Understanding How PortFast Works

Spanning-tree PortFast causes a switch or trunk port to enter the spanning-tree forwarding state immediately, bypassing the listening and learning states.

You can use PortFast on switch or trunk ports that are connected to a single workstation, switch, or server to allow those devices to connect to the network immediately, instead of waiting for the port to transition from the listening and learning states to the forwarding state.

Caution You can use PortFast to connect a single end station or a switch port to a switch port. If you enable PortFast on a port that is connected to another Layer 2 device, such as a switch, you might create network loops.

When the switch powers up, or when a device is connected to a port, the port enters the spanning-tree listening state. When the Forward Delay timer expires, the port enters the learning state. When the Forward Delay timer expires a second time, the port transitions to the forwarding or blocking state.

When you enable PortFast on a switch or trunk port, the port transitions immediately to the spanning-tree forwarding state.

Understanding How PortFast BPDU Guard Works

BPDU guard prevents spanning-tree loops by moving a port into the errdisable state when a BPDU is received on that port. When you enable BPDU guard on the switch, spanning tree shuts down the interfaces that receive BPDUs instead of putting the interfaces into the spanning-tree blocking state. When you enable BPDU guard globally and set the port configuration as the default for BPDU guard (see the "Configuring PortFast BPDU Guard on the Switch" section on page 9-11), then the PortFast configuration enables or disables BPDU guard.

If the port configuration is not set to default, then PortFast will not affect BPDU guard. Table 9-1 lists all the possible BPDU guard port configurations. BPDU guard can prevent invalid configurations, because you must manually put the interface back in service.

Table 9-1 BPDU Guard Port Configurations

Per-Port Configuration	Global Configuration	PortFast Operational Value	Operational BPDU Guard
Default	Enable	Enable	Enable
Default	Enable	Disable	Disable
Default	Disable	X	Disable
Disable	X	X	Disable
Enable	X	X	Enable

Understanding How PortFast BPDU Filtering Works

BPDU filtering allows you to avoid transmitting BPDUs on a port that is connected to an end system. When you enable BPDU filtering on the switch, spanning tree places that port in the forwarding state immediately, instead of going through the listening, learning, and forwarding states. When you enable BPDU filtering globally and set the port configuration as the default for BPDU filtering (see the "Configuring PortFast BPDU Filtering on the Switch" section on page 9-13), then PortFast enables or disables the BPDU filter.

If the port configuration is not set to default, then the PortFast configuration will not affect BPDU filtering. Table 9-2 lists all the possible BPDU filter combinations. The BPDU filter allows access ports to move directly to the forwarding state as soon as the end hosts are connected.

Table 9-2 BPDU Filter Port Configurations

Per-Port Configuration	Global Configuration	PortFast Operational Value	Operational BPDU Filter
Default	Enable	Enable	Enable[1]
Default	Enable	Disable	Disable
Default	Disable	X	Disable
Disable	X	X	Disable
Enable	X	X	Enable

1. The port transmits at least 10 BPDUs. If this port receives any BPDUs, then the operational PortFast value is set to disable and the operational BPDU filter is then disabled.

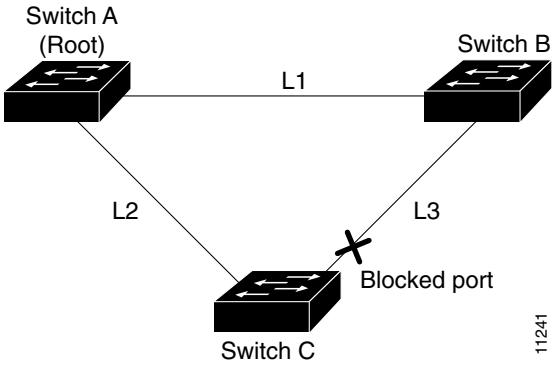
Understanding How UplinkFast Works

UplinkFast provides fast convergence after a spanning-tree topology change and achieves load balancing between redundant links using uplink groups. An uplink group is a set of ports (per VLAN), only one of which is forwarding at any given time. Specifically, an uplink group consists of the root port (which is forwarding) and a set of blocked ports. The blocked ports do not include self-looping ports. The uplink group provides an alternate path in case the currently forwarding link fails.

Note UplinkFast is most useful in wiring-closet switches. This feature may not be useful for other types of applications.

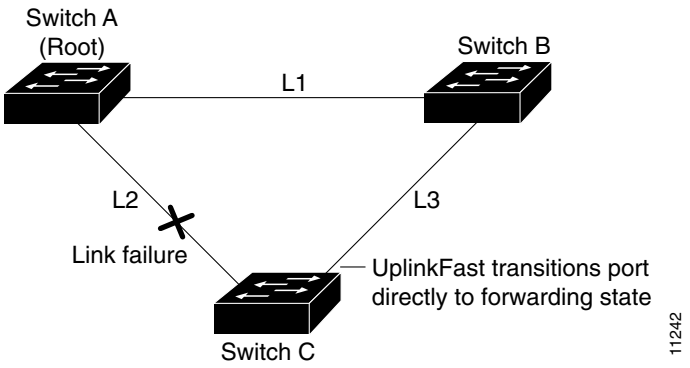
Figure 9-1 shows an example topology with no link failures. Switch A, the root switch, is connected directly to Switch B over link L1 and to Switch C over link L2. The port on Switch C that is connected directly to Switch B is in the blocking state.

Figure 9-1 UplinkFast Example Before Direct Link Failure



If Switch C detects a link failure on the currently active link L2 (a direct link failure), UplinkFast unblocks the blocked port on Switch C and transitions it to the forwarding state without going through the listening and learning states, as shown in Figure 9-2. This switchover takes approximately 1 to 5 seconds.

Figure 9-2 UplinkFast Example After Direct Link Failure



Understanding How BackboneFast Works

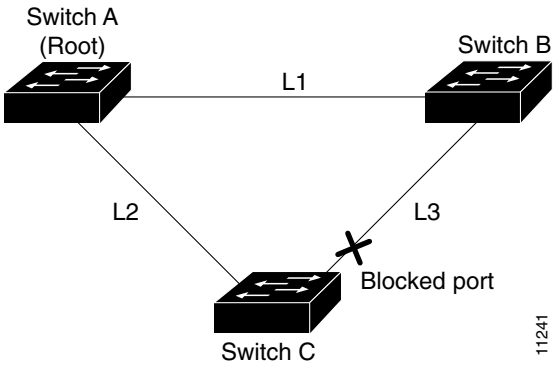
BackboneFast is initiated when a designated port on a network device receives inferior BPDUs from its designated bridge. Inferior BPDUs can occur when the designated device loses the root and advertises a root with a higher bridge ID or the device path/cost to the root is higher than the network device. When a switch receives an inferior BPDU, it indicates that a link to which the switch is not directly connected (an indirect link) has failed (that is, the designated bridge has lost its connection to the root bridge). Under normal spanning-tree rules, the switch ignores inferior BPDUs for the configured maximum aging time, as specified by the *agingtime* variable of the **set spantree maxage** command.

The switch tries to determine if it has an alternate path to the root bridge. If the inferior BPDU arrives on a blocked port, the root port and other blocked ports on the switch become alternate paths to the root bridge. (Self-looped ports are not considered alternate paths to the root bridge.) If the inferior BPDU arrives on the root port, all blocked ports become alternate paths to the root bridge. If the inferior BPDU arrives on the root port and there are no blocked ports, the switch assumes that it has lost connectivity to the root bridge, causes the maximum aging time on the root to expire, and becomes the root switch according to normal spanning-tree rules.

If the switch has alternate paths to the root bridge, it uses these alternate paths to transmit a new kind of PDU called the Root Link Query PDU out all alternate paths to the root bridge. If the switch determines that it still has an alternate path to the root, it causes the maximum aging time on the ports on which it received the inferior BPDU to expire. If all the alternate paths to the root bridge indicate that the switch has lost connectivity to the root bridge, the switch causes the maximum aging times on the ports on which it received an inferior BPDU to expire. If one or more alternate paths can still connect to the root bridge, the switch makes all ports on which it received an inferior BPDU its designated ports and moves them out of the blocking state (if they were in the blocking state) through the listening and learning states and into the forwarding state.

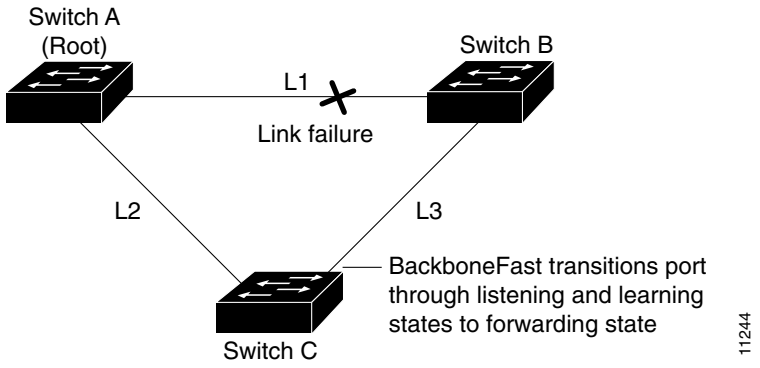
Figure 9-3 shows an example topology with no link failures. Switch A, the root switch, connects directly to Switch B over link L1 and to Switch C over link L2. The port on Switch C that connects directly to Switch B is in the blocking state.

Figure 9-3 BackboneFast Example Before Indirect Link Failure



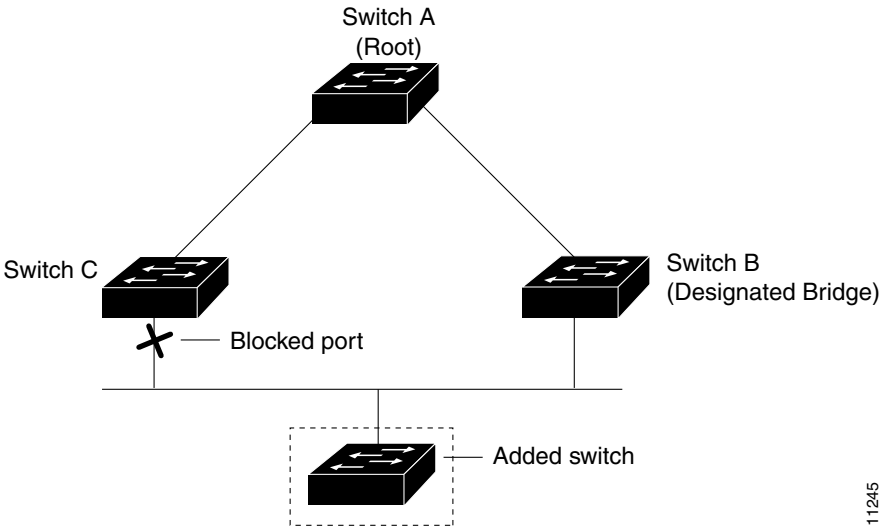
If link L1 fails, Switch C detects this failure as an indirect failure, because it is not connected directly to link L1. Switch B no longer has a path to the root switch. BackboneFast allows the blocked port on Switch C to move immediately to the listening state without waiting for the maximum aging time for the port to expire. BackboneFast then transitions the port on Switch C to the forwarding state, providing a path from Switch B to Switch A. This switchover takes approximately 30 seconds. Figure 9-4 shows how BackboneFast reconfigures the topology to account for the failure of link L1.

Figure 9-4 BackboneFast Example After Indirect Link Failure



If a new switch is introduced into a shared-medium topology, BackboneFast is not activated. Figure 9-5 shows a shared-medium topology in which a new switch is added. The new switch begins sending inferior BPDUs that say it is the root switch. However, the other switches ignore these inferior BPDUs and the new switch learns that Switch B is the designated bridge to Switch A, the root switch.

Figure 9-5 Adding a Switch in a Shared-Medium Topology



Understanding How Loop Guard Works

Unidirectional link failures may cause a root port or alternate port to become designated as root if BPDUs are absent. Some software failures may introduce temporary loops in the network. Loop guard checks if a root port or an alternate root port receives BPDUs. If the port is not receiving BPDUs, loop guard puts the port into an inconsistent state until it starts receiving BPDUs again. Loop guard isolates the failure and lets spanning tree converge to a stable topology without the failed link or bridge.

You can enable loop guard on a per-port basis. When you enable loop guard, it is automatically applied to all of the active instances or VLANs to which that port belongs. When you disable loop guard, it is disabled for the specified ports. Disabling loop guard moves all loop-inconsistent ports to the listening state.

If you enable loop guard on a channel and the first link becomes unidirectional, loop guard blocks the entire channel until the affected port is removed from the channel. Figure 9-6 shows loop guard in a triangle switch configuration.

Figure 9-6 Triangle Switch Configuration with Loop Guard

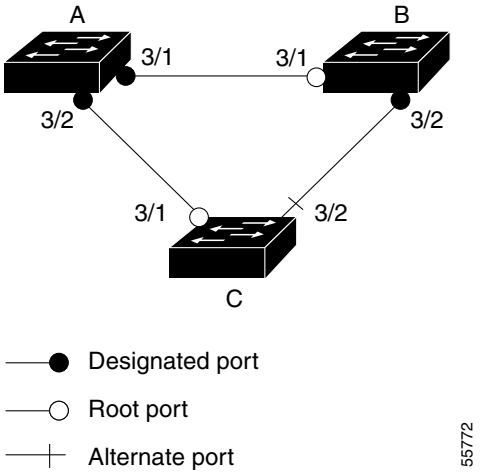


Figure 9-6 illustrates the following configuration:

- Switches A and B are distribution switches.
- Switch C is an access switch.
- Loop guard is enabled on ports 3/1 and 3/2 on Switches A, B, and C.

Use loop guard only in topologies where there are blocked ports. Topologies that have no blocked ports, which are loop free, do not need to enable this feature. Enabling loop guard on a root switch has no effect but provides protection when a root switch becomes a nonroot switch.

Follow these guidelines when using loop guard:

- You cannot enable loop guard on PortFast-enabled or dynamic VLAN ports.
- You cannot enable PortFast on loop guard-enabled ports.
- You cannot enable loop guard if root guard is enabled.

Loop guard interacts with other features as follows:

- Loop guard does not affect the functionality of UplinkFast or BackboneFast.
- Do not enable loop guard on ports that are connected to a shared link.

Note We recommend that you enable loop guard on root ports and alternate root ports on access switches.

- Root guard forces a port to be always designated as the root port. Loop guard is effective only if the port is a root port or an alternate port. You cannot enable loop guard and root guard on a port at the same time.
- PortFast transitions a port into a forwarding state immediately when a link is established. Because a PortFast-enabled port will not be a root port or alternate port, loop guard and PortFast cannot be configured on the same port. Assigning dynamic VLAN membership for the port requires that the port is PortFast enabled. You cannot configure a loop guard-enabled port with dynamic VLAN membership.

- If your network has a type-inconsistent port or a PVID-inconsistent port, all BPDUs are dropped until the misconfiguration is corrected. The port transitions out of the inconsistent state after the message age expires. Loop guard ignores the message age expiration on type-inconsistent ports and PVID-inconsistent ports. If the port is already blocked by loop guard, misconfigured BPDUs that are received on the port make loop guard recover, but the port is moved into the type-inconsistent state or PVID-inconsistent state.
- In high-availability switch configurations, if a port is put into the blocked state by loop guard, it remains blocked even after a switchover to the redundant supervisor engine. The newly activated supervisor engine recovers the port only after receiving a BPDU on that port.
- Loop guard uses the ports known to spanning tree. Loop guard can take advantage of logical ports that are provided by the Port Aggregation Protocol (PAgP). However, to form a channel, all the physical ports that are grouped in the channel must have compatible configurations. PAgP enforces uniform configurations of root guard or loop guard on all the physical ports to form a channel.

  These caveats apply to loop guard:

  - Spanning tree always chooses the first operational port in the channel to send the BPDUs. If that link becomes unidirectional, loop guard blocks the channel, even if other links in the channel are functioning properly.
  - If a set of ports that are already blocked by loop guard are grouped together to form a channel, spanning tree loses all the state information for those ports and the new channel port may obtain the forwarding state with a designated role.
  - If a channel is blocked by loop guard and the channel breaks, spanning tree loses all the state information. The individual physical ports may obtain the forwarding state with the designated role, even if one or more of the links that formed the channel are unidirectional.

  Note You can enable UniDirectional Link Detection (UDLD) to help isolate the link failure. A loop may occur until UDLD detects the failure, but loop guard will not be able to detect it.

- Loop guard has no effect on a disabled spanning tree instance or a VLAN.

Configuring PortFast on the Switch

These sections describe how to configure spanning-tree PortFast on the switch:

Enabling PortFast on an Access Port

Caution You can use PortFast to connect a single end station or a switch port to a switch port. If you enable PortFast on a port that is connected to another Layer 2 device, such as a switch, you might create network loops.

To enable PortFast on a switch port, perform this task in privileged mode:

	Task	Command
Step 1	Enable PortFast on a switch port that is connected to a single workstation, switch, or server.	**set spantree portfast** *mod_num/port_num* **enable** | **disable**
Step 2	Verify the PortFast setting on a switch port.	**show spantree** [*mod_num/port_num*] [*vlan*]

This example shows how to enable PortFast on port 1 of module 4 and verify the configuration; the PortFast status is shown in the "Fast-Start" column:

```
Console> (enable) set spantree portfast 4/1 enable
Warning:Connecting Layer 2 devices to a fast start port can cause
temporary spanning tree loops. Use with caution.

Spantree port  4/1 fast start enabled.
Console> (enable) show spantree 4/1
Port      Vlan  Port-State     Cost   Priority  Fast-Start  Group-method
--------- ----  -------------  -----  --------  ----------  ------------
 4/1      1     blocking          19        20   enabled
 4/1      100   forwarding        10        20   enabled
 4/1      521   blocking          19        20   enabled
 4/1      522   blocking          19        20   enabled
 4/1      523   blocking          19        20   enabled
 4/1      524   blocking          19        20   enabled
 4/1      1003  not-connected     19        20   enabled
 4/1      1005  not-connected     19         4   enabled
Console> (enable)
```

Note If the designation for a port is displayed as edge, that port is also a PortFast port. See the "Edge Ports" section on page 7-23.

Enabling Spanning-Tree PortFast on a Trunk Port

Caution You can use PortFast to connect a single end station or a switch port to a switch port. If you enable PortFast on a port that is connected to another Layer 2 device, such as a switch, you might create network loops.

To enable PortFast on a trunk port, perform this task in privileged mode:

	Task	Command
Step 1	Enable PortFast on a trunk port that is connected to a single workstation, switch, or server.	**set spantree portfast** *mod_num/port_num* **enable trunk** **Note** If you enter the **set spantree portfast** command on a trunk port without using the **trunk** keyword, the trunk port will stay in disable mode.
Step 2	Verify the PortFast setting on a trunk port.	**show spantree portfast** [*mod_num/port_num*]

This example shows how to enable PortFast on port 1 of module 4 of a trunk port, bring the trunk port to a forwarding state, and verify the configuration (the PortFast status is shown in the "Fast-Start" column):

```
Console> (enable) set spantree portfast 4/1 enable trunk
Warning:Connecting Layer 2 devices to a fast start port can cause
temporary spanning tree loops. Use with caution.

Spantree port  4/1 fast start enabled.
Console> (enable) show spantree 4/1
Port                     Vlan Port-State    Cost      Prio Portfast
Channel_id
------------------------ ---- ------------ --------- ---- ------------------
 4/1                     1    blocking              4   32 enabled  0
 4/1                     100  forwarding            4   32 enabled  0
 4/1                     521  blocking              4   32 enabled  0
 4/1                     524  blocking              4   32 enabled  0
 4/1                     1003 not-connected         4   32 enabled  0
 4/1                     1005 not-connected         4   32 enabled  0
Console> (enable) show spantree portfast 4/1
Portfast:enable trunk
Portfast BPDU guard is disabled.
Portfast BPDU filter is disabled.
Console>
```

Note When PortFast is enabled between two switches, the system will verify that there are no loops in the network before bringing the blocking trunk to a forwarding state.

Disabling PortFast

To disable PortFast on a switch or trunk port, perform this task in privileged mode:

	Task	Command
Step 1	Disable PortFast on a switch port.	**set spantree portfast** *mod_num/port_num* **disable**
Step 2	Verify the PortFast setting.	**show spantree** *mod_num/port_num*

This example shows how to disable PortFast on port 1 of module 4:

```
Console> (enable) set spantree portfast 4/1 disable
Spantree port 4/1 fast start disabled.
Console> (enable)
```

Resetting PortFast

To reset PortFast on a switch or trunk port to its default settings, perform this task in privileged mode:

	Task	Command
Step 1	Reset PortFast to its default settings on a switch port.	**set spantree portfast** *mod_num/port_num* **default**
Step 2	Verify the PortFast setting.	**show spantree** *mod_num/port_num*

This example shows how to reset PortFast to its default settings on port 1 of module 4:

```
Console> (enable) set spantree portfast 4/1 default

Spantree port  4/1 fast start set to default.

Console> (enable) show spantree portfast 4/1
Portfast:default
Portfast BPDU guard is disabled.
Portfast BPDU filter is disabled.
Console> (enable)
```

Configuring PortFast BPDU Guard on the Switch

These sections describe how to configure PortFast BPDU guard on the switch:

- Enabling PortFast BPDU Guard, page 9-11
- Disabling PortFast BPDU Guard, page 9-12

Enabling PortFast BPDU Guard

The PortFast feature is configured on an individual port and the PortFast BPDU guard option is configured either globally or on a per-port basis.

When you disable PortFast on a port, PortFast BPDU guard becomes inactive. Port configuration overrides global configuration unless port configuration is set to default. If port configuration is set to default, global configuration is checked. If the port configuration is enabled, the port configuration is used and the global configuration is not used.

To enable PortFast BPDU guard on a nontrunking switch port, perform this task in privileged mode:

	Task	Command
Step 1	Enable BPDU guard on an individual port.	**set spantree portfast bpdu-guard** *mod/port* [**disable** | **enable** | **default**]
Step 2	Verify the PortFast BPDU guard setting.	**show spantree summary**

This example shows how to enable PortFast BPDU guard on the switch and verify the configuration in the Per VLAN Spanning Tree + (PVST+) mode:

Note For additional PVST+ information, see Chapter 7, "Configuring Spanning Tree."

```
Console> (enable) set spantree portfast bpdu-guard 6/1 enable
Spantree port  6/1 bpdu guard enabled.
Console> (enable)
Console> (enable) show spantree summary
Root switch for vlans: none.
Portfast bpdu-guard enabled for bridge.
Uplinkfast disabled for bridge.
Backbonefast disabled for bridge.

Vlan  Blocking Listening Learning Forwarding STP Active
----- -------- --------- -------- ---------- ----------
   1         0         0        0          4          4
   2         0         0        0          4          4
   3         0         0        0          4          4
   4         0         0        0          4          4
   5         0         0        0          4          4
   6         0         0        0          4          4
  10         0         0        0          4          4
  20         0         0        0          4          4
  50         0         0        0          4          4
 100         0         0        0          4          4
 152         0         0        0          4          4
 200         0         0        0          5          5
 300         0         0        0          4          4
 400         0         0        0          4          4
 500         0         0        0          4          4
 521         0         0        0          4          4
 524         0         0        0          4          4
 570         0         0        0          4          4
 801         0         0        0          0          0
 802         0         0        0          0          0
 850         0         0        0          4          4
 917         0         0        0          4          4
 999         0         0        0          4          4
1003         0         0        0          0          0
1005         0         0        0          0          0

      Blocking Listening Learning Forwarding STP Active
----- -------- --------- -------- ---------- ----------
Total        0         0        0         85         85
```

Disabling PortFast BPDU Guard

To disable PortFast BPDU guard on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Disable PortFast BPDU guard on the switch.	**set spantree portfast bpdu-guard** *mod/port* [**disable** | **enable** | **default**]
Step 2	Verify the PortFast BPDU guard setting.	**show spantree summary**

This example shows how to disable PortFast BPDU guard on the switch and verify the configuration:

```
Console> (enable) set spantree portfast bpdu-guard disable
Spantree portfast bpdu-guard disabled on this switch.
Console> (enable) show spantree summary
Summary of connected spanning tree ports by vlan

Portfast bpdu-guard disabled for bridge.
Uplinkfast disabled for bridge.
Backbonefast disabled for bridge.

Vlan  Blocking Listening Learning Forwarding STP Active
----- -------- --------- -------- ---------- ----------
   1         0         0        0          4          4
   2         0         0        0          4          4
   3         0         0        0          4          4
   4         0         0        0          4          4
   5         0         0        0          4          4
   6         0         0        0          4          4
  10         0         0        0          4          4
  20         0         0        0          4          4
  50         0         0        0          4          4
 100         0         0        0          4          4
 152         0         0        0          4          4
 200         0         0        0          5          5
 300         0         0        0          4          4
 400         0         0        0          4          4
 500         0         0        0          4          4
 521         0         0        0          4          4
 524         0         0        0          4          4
 570         0         0        0          4          4
 801         0         0        0          0          0
 802         0         0        0          0          0
 850         0         0        0          4          4
 917         0         0        0          4          4
 999         0         0        0          4          4
1003         0         0        0          0          0
1005         0         0        0          0          0

      Blocking Listening Learning Forwarding STP Active
----- -------- --------- -------- ---------- ----------
Total        0         0        0         85         85
Console> (enable)
```

Configuring PortFast BPDU Filtering on the Switch

These sections describe how to configure PortFast BPDU filtering on the switch:

- Enabling PortFast BPDU Filtering, page 9-14
- Disabling PortFast BPDU Filtering, page 9-15

Enabling PortFast BPDU Filtering

To enable PortFast BPDU filtering on a nontrunking port, perform this task in privileged mode:

	Task	Command
Step 1	Set the BPDU filter state on the port.	**set spantree portfast bpdu-filter** *mod/port* [**disable** | **enable** | **default**]
Step 2	Verify the PortFast BPDU filter setting.	**show spantree summary**

BPDU filtering is set to default on each port. This example shows how to enable PortFast BPDU filtering on the port and verify the configuration in PVST+ mode:

Note For PVST+ information, see Chapter 7, "Configuring Spanning Tree."

```
Console> (enable) set spantree portfast bpdu-filter 6/1 enable
Warning:Ports enabled with bpdu filter will not send BPDUs and drop all
received BPDUs. You may cause loops in the bridged network if you misuse
this feature.

Console> (enable) show spantree summary
Root switch for vlans: none.
Portfast bpdu-filter enabled for bridge.
Uplinkfast disabled for bridge.
Backbonefast disabled for bridge.

Vlan  Blocking Listening Learning Forwarding STP Active
----- -------- --------- -------- ---------- ----------
   1         0         0        0          4          4
   2         0         0        0          4          4
   3         0         0        0          4          4
   4         0         0        0          4          4
   5         0         0        0          4          4
   6         0         0        0          4          4
 850         0         0        0          4          4
 917         0         0        0          4          4
 999         0         0        0          4          4
1003         0         0        0          0          0
1005         0         0        0          0          0

      Blocking Listening Learning Forwarding STP Active
----- -------- --------- -------- ---------- ----------
Total        0         0        0         85         85
Console> (enable)
```

Disabling PortFast BPDU Filtering

To disable PortFast BPDU filtering on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Disable PortFast BPDU filtering on the switch.	**set spantree portfast bpdu-filter disable**
Step 2	Verify the PortFast BPDU filter setting.	**show spantree summary** **show portfast**

This example shows how to disable PortFast BPDU filtering on the switch and verify the configuration:

```
Console> (enable) set spantree portfast bpdu-filter disable
Spantree portfast bpdu-filter disabled on this switch.
Console> (enable) show spantree summary
Summary of connected spanning tree ports by vlan

Portfast bpdu-filter disabled for bridge.
Uplinkfast disabled for bridge.
Backbonefast disabled for bridge.

Vlan  Blocking Listening Learning Forwarding STP Active
----- -------- --------- -------- ---------- ----------
   1         0         0        0          4          4
   2         0         0        0          4          4
   3         0         0        0          4          4
   4         0         0        0          4          4
   5         0         0        0          4          4
   6         0         0        0          4          4
  10         0         0        0          4          4
.
.
 802         0         0        0          0          0
 850         0         0        0          4          4
 917         0         0        0          4          4
 999         0         0        0          4          4
1003         0         0        0          0          0
1005         0         0        0          0          0

      Blocking Listening Learning Forwarding STP Active
----- -------- --------- -------- ---------- ----------
Total        0         0        0         85         85
Console> (enable)
```

Configuring UplinkFast on the Switch

You can configure UplinkFast for PVST+ or for Multi-Instance Spanning Tree Protocol (MISTP). The command is the same but the output may be slightly different.

Note For additional MISTP information, see Chapter 7, “Configuring Spanning Tree.”

These sections describe how to configure UplinkFast on the switch:

Enabling UplinkFast

The **set spantree uplinkfast enable** command increases the path cost of all ports on the switch, making it unlikely that the switch will become the root switch. The *station_update_rate* value represents the number of multicast packets that are transmitted per 100 milliseconds (the default is 15 packets per millisecond).

Note When you enable the **set spantree uplinkfast** command, it affects all VLANs on the switch. You cannot configure UplinkFast on an individual VLAN.

To enable UplinkFast on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Enable UplinkFast on the switch.	**set spantree uplinkfast enable** [**rate** *station_update_rate*] [**all-protocols off** | **on**]
Step 2	Verify that UplinkFast is enabled.	**show spantree uplinkfast** [{**mistp-instance** [*instances*]}| *vlans*]

With PVST+ mode enabled, this example shows how to enable UplinkFast with a station-update rate of 40 packets per 100 milliseconds and verify that UplinkFast is enabled:

```
Console> (enable) set spantree uplinkfast enable
VLANs 1-4094 bridge priority set to 49152.
The port cost and portvlancost of all ports set to above 3000.
Station update rate set to 15 packets/100ms.
uplinkfast all-protocols field set to off.
uplinkfast enabled for bridge.
Console> (enable) show spantree uplinkfast 1 100 521-524
Station update rate set to 15 packets/100ms.
uplinkfast all-protocols field set to off.
VLAN          port list
-----------------------------------------------
1             1/1(fwd),1/2
100           1/2(fwd)
521           1/1(fwd),1/2
522           1/1(fwd),1/2
523           1/1(fwd),1/2
524           1/1(fwd),1/2
Console> (enable)
```

This example shows how to display the UplinkFast settings for all VLANs:

```
Console> show spantree uplinkfast
Station update rate set to 15 packets/100ms.
uplinkfast all-protocols field set to off.
VLAN port list
-----------------------------------------------
1-20   1/1(fwd),1/2-1/5
21-50  1/9(fwd), 1/6-1/8, 1/10-1/12
51-100 2/1(fwd), 2/12
Console> (enable)
```

With MISTP mode enabled, this example shows the output when you enable UplinkFast:

```
Console> (enable) set spantree uplinkfast enable
Instances 1-16 bridge priority set to 49152.
The port cost and portinstancecost of all ports set to above 10000000.
Station update rate set to 15 packets/100ms.
uplinkfast all-protocols field set to off.
uplinkfast enabled for bridge.
Console> (enable)
```

This example shows how to display the UplinkFast settings for a specific instance:

```
Console> show spantree uplinkfast mistp-instance 1
Station update rate set to 15 packets/100ms.
uplinkfast all-protocols field set to off.
Inst   port list
-------------------------------------------------
1      4/1(fwd)
Console> (enable)
```

Disabling UplinkFast

The **set spantree uplinkfast disable** command disables UplinkFast on the switch, but the switch priority and port cost values are not reset to the factory defaults.

Note When you enter the **set spantree uplinkfast disable** command, it affects all VLANs on the switch. You cannot disable UplinkFast on an individual VLAN.

To disable UplinkFast on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Disable UplinkFast on the switch.	**set spantree uplinkfast disable**
Step 2	Verify that UplinkFast is disabled.	**show spantree uplinkfast**

With PVST+ mode enabled, this example shows how to disable UplinkFast on the switch and verify the configuration:

```
Console> (enable) set spantree uplinkfast disable
Uplinkfast disabled for switch.
Use clear spantree uplinkfast to return stp parameters to default.
Console> (enable) show spantree uplinkfast
Station update rate set to 15 packets/100ms.
uplinkfast all-protocols field set to off.
VLAN          port list
-------------------------------------------------
1             1/1(fwd),1/2
100           1/2(fwd)
521           1/1(fwd),1/2
522           1/1(fwd),1/2
523           1/1(fwd),1/2
524           1/1(fwd),1/2
Console> (enable)
```

Configuring BackboneFast on the Switch

These sections describe how to configure BackboneFast:

Enabling BackboneFast

Note For BackboneFast to work, you must enable it on all switches in the network. BackboneFast is not supported on Token Ring VLANs. This feature is supported for use with third-party switches.

To enable BackboneFast on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Enable BackboneFast on the switch.	**set spantree backbonefast enable**
Step 2	Verify that BackboneFast is enabled.	**show spantree backbonefast**

This example shows how to enable BackboneFast on the switch and verify the configuration:

```
Console> (enable) set spantree backbonefast enable
Backbonefast enabled for all VLANs
Console> (enable) show spantree backbonefast
Backbonefast is enabled.
Console> (enable)
```

Displaying BackboneFast Statistics

To display BackboneFast statistics, perform this task in privileged mode:

Task	Command
Display BackboneFast statistics.	**show spantree summary**

This example shows how to display BackboneFast statistics:

```
Console> (enable) show spantree summary
Summary of connected spanning tree ports by vlan

Uplinkfast disabled for bridge.
Backbonefast enabled for bridge.
```

```
Vlan  Blocking Listening Learning Forwarding STP Active
----- -------- --------- -------- ---------- ----------
   1         0         0        0          1          1

      Blocking Listening Learning Forwarding STP Active
----- -------- --------- -------- ---------- ----------
Total        0         0        0          1          1
BackboneFast statistics
-----------------------
Number of inferior BPDUs received (all VLANs)  : 0
Number of RLQ req PDUs received (all VLANs)    : 0
Number of RLQ res PDUs received (all VLANs)    : 0
Number of RLQ req PDUs transmitted (all VLANs) : 0
Number of RLQ res PDUs transmitted (all VLANs) : 0
Console> (enable)
```

Disabling BackboneFast

To disable BackboneFast on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Disable BackboneFast on the switch.	**set spantree backbonefast disable**
Step 2	Verify that BackboneFast is disabled.	**show spantree backbonefast**

This example shows how to disable BackboneFast on the switch and verify the configuration:

```
Console> (enable) set spantree backbonefast disable
Backbonefast enabled for all VLANs
Console> (enable) show spantree backbonefast
Backbonefast is disable.
Console> (enable)
```

Configuring Loop Guard on the Switch

These sections describe how to configure BackboneFast:

- Enabling Loop Guard, page 9-19
- Disabling Loop Guard, page 9-20

Enabling Loop Guard

Use the **set spantree guard** command to enable or disable the spanning-tree loop guard on a per-port basis.

To enable loop guard on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Enable loop guard on a port.	**set spantree guard loop** *mod/port*
Step 2	Verify that loop guard is enabled.	**show spantree guard** {*mod/port* | *vlan*} **mistp-instance** *instance*

This example shows how to enable loop guard:

```
Console> (enable) set spantree guard loop 5/1
Rootguard is enabled on port 5/1, enabling loopguard will disable rootguard on this port.
Do you want to continue (y/n) [n]? y
Loopguard on port 5/1 is enabled.
Console> (enable)
```

Disabling Loop Guard

To disable loop guard on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Disable loop guard on a port.	**set spantree guard none** *mod/port*
Step 2	Verify that loop guard is disabled.	**show spantree guard** {*mod/port* | *vlan*} **mistp-instance** *instance*

This example shows how to disable loop guard:

```
Console> (enable) set spantree guard none 5/1
Rootguard is disabled on port 5/1, disabling loopguard will disable rootguard on this
port.
Do you want to continue (y/n) [n]? y
Loopguard on port 5/1 is disabled.
Console> (enable)
```

CHAPTER 10

Configuring VTP

This chapter describes how to configure the VLAN Trunking Protocol (VTP) on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Understanding How VTP Version 1 and Version 2 Work

VTP is a Layer 2 messaging protocol that maintains VLAN configuration consistency by managing the addition, deletion, and renaming of VLANs on a network-wide basis. VTP minimizes misconfigurations and configuration inconsistencies that can result in a number of problems, such as duplicate VLAN names, incorrect VLAN-type specifications, and security violations.

You can use VTP to manage VLANs 1–1005 in your network. (VTP version 1 and VTP version 2 do not support VLANs 1025–4094.) With VTP, you can make configuration changes centrally on one switch and have those changes automatically communicated to all the other switches in the network.

Note For complete information on configuring VLANs, see Chapter 11, "Configuring VLANs."

These sections describe how VTP works:

Understanding the VTP Domain

A VTP domain (also called a VLAN management domain) is made up of one or more interconnected switches that share the same VTP domain name. A switch can be configured to be in one and only one VTP domain. You make global VLAN configuration changes for the domain using either the command-line interface (CLI) or Simple Network Management Protocol (SNMP).

By default, the switch is in VTP server mode and is in the no-management domain state until the switch receives an advertisement for a domain over a trunk link or you configure a management domain. You cannot create or modify VLANs on a VTP server until the management domain name is specified or learned.

If the switch receives a VTP advertisement over a trunk link, it inherits the management domain name and the VTP configuration revision number. The switch ignores advertisements with a different management domain name or an earlier configuration revision number.

If you configure the switch as VTP transparent, you can create and modify VLANs but the changes affect only the individual switch.

When you make a change to the VLAN configuration on a VTP server, the change is propagated to all switches in the VTP domain. VTP advertisements are transmitted out all trunk connections, including Inter-Switch Link (ISL), IEEE 802.1Q, IEEE 802.10, and ATM LAN Emulation (LANE).

VTP maps VLANs dynamically across multiple LAN types with unique names and internal index associations. Mapping eliminates excessive device administration that is required from network administrators.

Understanding VTP Modes

You can configure a switch to operate in any one of these VTP modes:

- Server—In VTP server mode, you can create, modify, and delete VLANs and specify other configuration parameters (such as VTP version and VTP pruning) for the entire VTP domain. VTP servers advertise their VLAN configuration to other switches in the same VTP domain and synchronize their VLAN configuration with other switches based on advertisements received over trunk links. VTP server is the default mode.
- Client—VTP clients behave the same way as VTP servers, but you cannot create, change, or delete VLANs on a VTP client.

- Transparent—VTP transparent switches do not participate in VTP. A VTP transparent switch does not advertise its VLAN configuration and does not synchronize its VLAN configuration based on received advertisements. However, in VTP version 2, transparent switches do forward VTP advertisements that they receive out their trunk ports.
- Off—In the three modes described above, VTP advertisements are received and transmitted as soon as the switch enters the management domain state. In the VTP off mode, the switch behaves the same as in VTP transparent mode with the exception that VTP advertisements are not forwarded.

Understanding VTP Advertisements

Each switch in the VTP domain sends periodic advertisements out each trunk port to a reserved multicast address. VTP advertisements are received by neighboring switches, which update their VTP and VLAN configurations as necessary.

The following global configuration information is distributed in VTP advertisements:

- VLAN IDs (ISL and 802.1Q)
- Emulated LAN names (for ATM LANE)
- 802.10 SAID values (FDDI)
- VTP domain name
- VTP configuration revision number
- VLAN configuration, including the maximum transmission unit (MTU) size for each VLAN
- Frame format

Understanding VTP Version 2

If you use VTP in your network, you must decide whether to use VTP version 1, version 2, or version 3 (for details on version 3, see the "Understanding How VTP Version 3 Works" section on page 10-12).

Note If you are using VTP in a Token Ring environment, you must use version 2.

VTP version 2 supports the following features that are not supported in version 1:

- Token Ring support—VTP version 2 supports Token Ring LAN switching and VLANs (Token Ring Bridge Relay Function [TrBRF] and Token Ring Concentrator Relay Function [TrCRF]). For more information about Token Ring VLANs, see Chapter 11, "Configuring VLANs."
- Unrecognized Type-Length-Value (TLV) Support—A VTP server or client propagates configuration changes to its other trunks even for TLVs it is not able to parse. The unrecognized TLV is saved in NVRAM.
- Version-Dependent Transparent Mode—In VTP version 1, a VTP transparent switch inspects VTP messages for the domain name and version and forwards a message only if the version and domain name match. Since only one domain is supported in the supervisor engine software, VTP version 2 forwards VTP messages in transparent mode without checking the version.

- Consistency Checks—In VTP version 2, VLAN consistency checks (such as VLAN names and values) are performed only when you enter new information through the CLI or SNMP. Consistency checks are not performed when new information is obtained from a VTP message, or when information is read from NVRAM. If the digest on a received VTP message is correct, its information is accepted without consistency checks.

Understanding VTP Pruning

Note Enabling VTP pruning on a VTP version 3 switch enables pruning only on the switch that you enable it on. VTP pruning is not propagated as it is with VTP version 1 and VTP version 2.

VTP pruning enhances network bandwidth use by reducing unnecessary flooded traffic, such as broadcast, multicast, unknown, and flooded unicast packets. VTP pruning increases available bandwidth by restricting flooded traffic to those trunk links that the traffic must use to access the appropriate network devices. By default, VTP pruning is disabled.

Make sure that all devices in the management domain support VTP pruning before enabling it. VTP pruning is supported in supervisor engine software release 5.1(1) and later releases.

Note If you use routers to route between emulated LANs, you should disable VTP pruning in the VTP management domain that contains the switches with ATM LANE modules installed (VTP pruning messages are sent over the ATM LANE module because it is a trunk). You can also disable pruning for the LANE VLANs by using the **clear vtp pruneeligible** command on all switches with ATM LANE modules.

Figure 10-1 shows a switched network without VTP pruning enabled. Port 1 on Switch 1 and port 2 on Switch 4 are assigned to the Red VLAN. A broadcast is sent from the host that is connected to Switch 1. Switch 1 floods the broadcast and every switch in the network receives it even though Switches 3, 5, and 6 have no ports in the Red VLAN.

Figure 10-1 Flooding Traffic without VTP Pruning

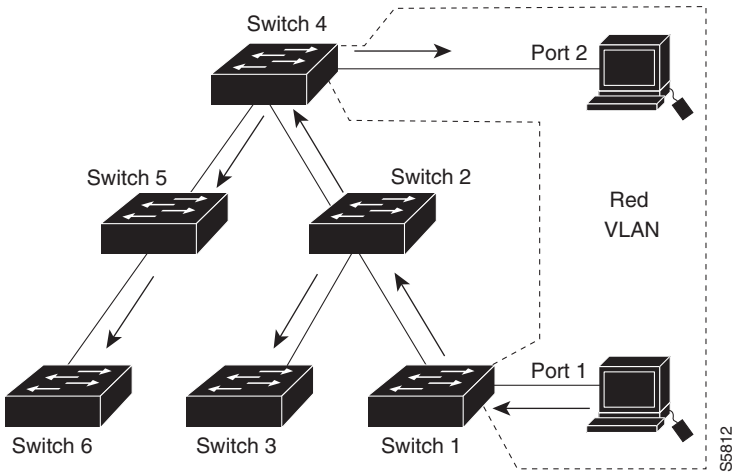
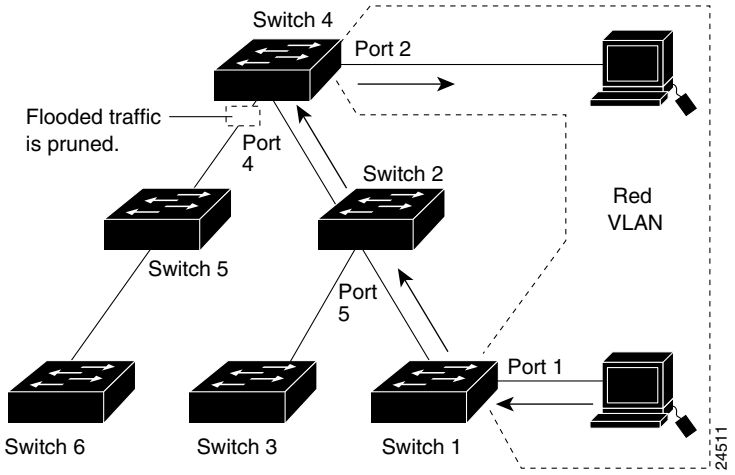


Figure 10-2 shows the same switched network with VTP pruning enabled. The broadcast traffic from Switch 1 is not forwarded to Switches 3, 5, and 6 because traffic for the Red VLAN has been pruned on the links indicated (port 5 on Switch 2 and port 4 on Switch 4).

Enabling VTP pruning on a VTP server enables pruning for the entire management domain. VTP pruning takes effect several seconds after you enable it. By default, VLANs 2–1000 are pruning eligible. VTP pruning does not prune traffic from VLANs that are pruning ineligible. VLAN 1 is always pruning ineligible; traffic from VLAN 1 cannot be pruned.

To make a VLAN pruning ineligible, enter the **clear vtp pruneeligible** command. To make a VLAN pruning eligible again, enter the **set vtp pruneeligible** command. You can set VLAN pruning eligibility regardless of whether VTP pruning is enabled or disabled for the domain. Pruning eligibility always applies to the local device only, not for the entire VTP domain.

Figure 10-2 Flooding Traffic with VTP Pruning



Default VTP Version 1 and Version 2 Configuration

Table 10-1 shows the default VTP configuration.

Table 10-1 VTP Default Configuration

Feature	Default Value
VTP domain name	Null
VTP mode	Server
VTP version 2 enable state	Version 1 is enabled (version 2 is disabled)
VTP password	None
VTP pruning	Disabled

VTP Version 1 and Version 2 Configuration Guidelines

This section describes the guidelines for implementing VTP in your network:

- All switches in a VTP domain must run the same VTP version.
- You must configure a password on each switch in the management domain when you are in secure mode.

Caution If you configure VTP in secure mode, the management domain will not function properly if you do not assign a management domain password to each switch in the domain.

- A VTP version 2-capable switch can operate in the same VTP domain as a switch running VTP version 1 if VTP version 2 is disabled on the VTP version 2-capable switch (VTP version 2 is disabled by default).
- Do not enable VTP version 2 on a switch unless all of the switches in the same VTP domain are version 2 capable. When you enable VTP version 2 on a switch, all of the version 2-capable switches in the domain enable VTP version 2.
- In a Token Ring environment, you must enable VTP version 2 for Token Ring VLAN switching to function properly.
- Enabling or disabling VTP pruning on a VTP server enables or disables VTP pruning for the entire management domain.
- Making VLANs pruning eligible or pruning ineligible on a switch affects pruning eligibility for those VLANs on that device only (not on all switches in the VTP domain).
- With software release 8.1(1), all VTP versions can be configured on a per-port basis. See the "VTP Version 3 Per-Port Configuration" section on page 10-14.

Configuring VTP Version 1 and Version 2

These sections describe how to configure VTP:

Configuring a VTP Server

When a switch is in VTP server mode, you can change the VLAN configuration and have it propagate throughout the network.

To configure the switch as a VTP server, perform this task in privileged mode:

	Task	Command
Step 1	Define the VTP domain name.	**set vtp domain** *name*
Step 2	Place the switch in VTP server mode.	**set vtp mode server**
Step 3	(Optional) Set a password for the VTP domain.	**set vtp passwd** *passwd*
Step 4	Verify the VTP configuration.	**show vtp domain**

This example shows how to configure the switch as a VTP server and verify the configuration:

```
Console> (enable) set vtp domain Lab_Network
VTP domain Lab_Network modified
Console> (enable) set vtp mode server
Changing VTP mode for all features
VTP domain Lab_Network modified
Console> (enable) show vtp domain
Version      : running VTP2 (VTP3 capable)
Domain Name  : Lab_Network                        Password  : configured (hidden)
Notifications: disabled                           Updater ID: 172.20.52.19

Feature        Mode           Revision
-------------- -------------- -----------
VLAN           Server         0

Pruning             : disabled
VLANs prune eligible: 2-1000
Console> (enable)
```

Configuring a VTP Client

When a switch is in VTP client mode, you cannot change the VLAN configuration on the switch. The client switch receives VTP updates from a VTP server in the management domain and modifies its configuration accordingly.

To configure the switch as a VTP client, perform this task in privileged mode:

	Task	Command
Step 1	Define the VTP domain name.	**set vtp domain** *name*
Step 2	Place the switch in VTP client mode.	**set vtp mode client**
Step 3	Verify the VTP configuration.	**show vtp domain**

This example shows how to configure the switch as a VTP client and verify the configuration:

```
Console> (enable) set vtp domain Lab_Network
VTP domain Lab_Network modified
Console> (enable) set vtp mode client
Changing VTP mode for all features
VTP domain Lab_Network modified
Console> (enable) show vtp domain
Version      : running VTP2 (VTP3 capable)
Domain Name  : Lab_Network                        Password  : configured (hidden)
Notifications: disabled                           Updater ID: 172.20.52.19

Feature        Mode           Revision
-------------- -------------- -----------
```

```
VLAN           Client         0

Pruning             : disabled
VLANs prune eligible: 2-1000
Console> (enable)
```

Configuring VTP (VTP Transparent Mode)

When you configure the switch as VTP transparent, you disable VTP on the switch. A VTP transparent switch does not send VTP updates and does not act on VTP updates that are received from other switches. However, a VTP transparent switch running VTP version 2 does forward received VTP advertisements out all of its trunk links.

Note Network devices in VTP transparent mode do not send VTP join messages. On Catalyst 6500 series switches with trunk connections to network devices in VTP transparent mode, configure the VLANs that are used by the transparent-mode network devices or that need to be carried across trunks as pruning ineligible (use the **clear vtp pruneeligible** command).

To disable VTP on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Disable VTP on the switch by configuring it for VTP transparent mode.	**set vtp mode transparent**
Step 2	Verify the VTP configuration.	**show vtp domain**

This example shows how to configure the switch as VTP transparent and verify the configuration:

```
Console> (enable) set vtp mode transparent
Changing VTP mode for all features
VTP domain Lab_Net modified
Console> (enable) show vtp domain
Version      : running VTP2 (VTP3 capable)
Domain Name  : Lab_Network                    Password  : configured (hidden)
Notifications: disabled                       Updater ID: 172.20.52.19

Feature        Mode           Revision
-------------- -------------- -----------
VLAN           Transparent    0

Pruning             : disabled
VLANs prune eligible: 2-1000
Console> (enable)
```

Disabling VTP Using the Off Mode

When you disable VTP using the off mode, the switch behaves the same as in VTP transparent mode with the exception that VTP advertisements are not forwarded.

To disable VTP using the off mode, perform this task in privileged mode:

	Task	Command
Step 1	Disable VTP using the off mode.	**set vtp mode off**
Step 2	Verify the VTP configuration.	**show vtp domain**

This example shows how to disable VTP using the off mode:

```
Console> (enable) set vtp mode off
Changing VTP mode for all features
VTP domain Lab_Net modified
Console> (enable) show vtp domain
Version      : running VTP2 (VTP3 capable)
Domain Name  : Lab_Network                        Password  : configured (hidden)
Notifications: disabled                           Updater ID: 172.20.52.19

Feature        Mode           Revision
-------------- -------------- -----------
VLAN           Off            0

Pruning             : disabled
VLANs prune eligible: 2-1000
Console> (enable)
```

Enabling VTP Version 2

VTP version 2 is disabled by default on VTP version 2-capable switches. When you enable VTP version 2 on a switch, every VTP version 2-capable switch in the VTP domain will enable version 2 as well.

Caution VTP version 1 and VTP version 2 are not interoperable on switches in the same VTP domain. Every switch in the VTP domain must use the same VTP version. Do not enable VTP version 2 unless every switch in the VTP domain supports version 2.

Note In a Token Ring environment, you must enable VTP version 2 for Token Ring VLAN switching to function properly.

To enable VTP version 2, perform this task in privileged mode:

	Task	Command
Step 1	Enable VTP version 2 on the switch.	**set vtp version 2**
Step 2	Verify that VTP version 2 is enabled.	**show vtp domain**

This example shows how to enable VTP version 2 and verify the configuration:

```
Console> (enable) set vtp version 2
This command will enable VTP version 2 function in the entire management domain.
All devices in the management domain should be version2-capable before enabling.
Do you want to continue (y/n) [n]? y
VTP domain server modified
Console> (enable) show vtp domain
```

```
Version      : running VTP2 (VTP3 capable)
Domain Name  : Lab_Network                      Password  : configured (hidden)
Notifications: disabled                         Updater ID: 172.20.52.19

Feature        Mode           Revision
-------------- -------------- -----------
VLAN           Off            0

Pruning             : disabled
VLANs prune eligible: 2-1000
Console> (enable)
```

Disabling VTP Version 2

To disable VTP version 2, perform this task in privileged mode:

	Task	Command
Step 1	Disable VTP version 2.	**set vtp version 1**
Step 2	Verify that VTP version 2 is disabled.	**show vtp domain**

This example shows how to disable VTP version 2:

```
Console> (enable) set vtp version 1
This command will enable VTP version 1 function in the entire management domain.
Warning: trbrf & trcrf vlans will not work properly in this version.
Do you want to continue (y/n) [n]? y
VTP domain Lab_Network modified
Console> (enable) show vtp domain
Version      : running VTP1 (VTP3 capable)
Domain Name  : Lab_Network                      Password  : configured (hidden)
Notifications: disabled                         Updater ID: 172.20.52.19

Feature        Mode           Revision
-------------- -------------- -----------
VLAN           Off            0

Pruning             : disabled
VLANs prune eligible: 2-1000
Console> (enable)
```

Enabling VTP Pruning

To enable VTP pruning, perform this task in privileged mode:

	Task	Command
Step 1	Enable VTP pruning in the management domain.	**set vtp pruning enable**
Step 2	(Optional) Make specific VLANs pruning ineligible on the device. (By default, VLANs 2–1000 are pruning eligible.)	**clear vtp pruneeligible** *vlan_range*
Step 3	(Optional) Make specific VLANs pruning eligible on the device.	**set vtp pruneeligible** *vlan_range*

	Task	Command
Step 4	Verify the VTP pruning configuration.	**show vtp domain**
Step 5	Verify that the appropriate VLANs are being pruned on trunk ports.	**show trunk**

This example shows how to enable VTP pruning in the management domain and how to make VLANs 2–99, 250–255, and 501–1000 pruning eligible on the particular device:

```
Console> (enable) set vtp pruning enable
Cannot modify pruning mode unless in VTP SERVER mode.

Console> (enable) set vtp mode server
Changing VTP mode for all features
VTP domain Lab_Network modified

Console> (enable) set vtp pruning enable
This command will enable the pruning function in the entire management domain.
All devices in the management domain should be pruning-capable before enabling.
Do you want to continue (y/n) [n]? y
VTP domain Lab_Network modified

Console> (enable) clear vtp pruneeligible 100-500
Vlans 1,100-500,1001-1023 will not be pruned on this device.
VTP domain Lab_Network modified.

Console> (enable) set vtp pruneeligible 250-255
Vlans 2-99,250-255,501-1000,1024-4094 eligible for pruning on this device.
VTP domain Lab_Network modified.

Console> (enable) show vtp domain
Version      : running VTP1 (VTP3 capable)
Domain Name  : Lab_Network                    Password  : configured (hidden)
Notifications: disabled                       Updater ID: 172.20.52.19

Feature        Mode           Revision
-------------- -------------- -----------
VLAN           Server         1

Pruning             : enabled
VLANs prune eligible: 2-99,250-255,501-1000

Console> (enable) show trunk
* - indicates vtp domain mismatch
# - indicates dot1q-all-tagged enabled on the port
Port      Mode         Encapsulation  Status        Native vlan
--------  -----------  -------------  ------------  -----------
16/1      nonegotiate  isl            trunking      1

Port      Vlans allowed on trunk
--------  ---------------------------------------------------------------------
16/1      1-1005,1025-4094

Port      Vlans allowed and active in management domain
--------  ---------------------------------------------------------------------
16/1

Port      Vlans in spanning tree forwarding state and not pruned
--------  ---------------------------------------------------------------------
16/1
Console> (enable)
```

Disabling VTP Pruning

To disable VTP pruning, perform this task in privileged mode:

	Task	Command
Step 1	Disable VTP pruning in the management domain.	**set vtp pruning disable**
Step 2	Verify that VTP pruning is disabled.	**show vtp domain**

This example shows how to disable VTP pruning in the management domain:

```
Console> (enable) set vtp pruning disable
This command will disable the pruning function in the entire management domain.
Do you want to continue (y/n) [n]? y
VTP domain Lab_Network modified
Console> (enable)
```

Displaying VTP Statistics

To display VTP statistics, including the VTP advertisements that are sent and received and VTP errors, perform this task:

Task	Command
Display VTP statistics for the switch.	**show vtp statistics**

This example shows how to display VTP statistics on the switch:

```
Console> (enable) show vtp statistics
VTP statistics:
summary advts received          0
subset  advts received          0
request advts received          0
summary advts transmitted       7843
subset  advts transmitted       4
request advts transmitted       20
No of config revision errors    0
No of config digest errors      0


VTP pruning statistics:

Trunk    Join Transmitted Join Received Summary advts received from GVRP PDU
                                        non-pruning-capable device  Received
-------- ---------------- ------------- --------------------------- ----------
16/1     75               0             0                           0
Console> (enable)
```

Understanding How VTP Version 3 Works

VTP version 3 differs from earlier VTP versions in that it does not directly handle VLANs. VTP version 3 is a protocol that is only responsible for distributing a list of opaque databases over an administrative domain. When enabled, VTP version 3 provides these enhancements to previous VTP versions:

- Support for extended VLANs.
- Support for the creation and advertising of private VLANs.
- Support for VLAN instances and MST mapping propagation instances.
- Improved server authentication.
- Protection from the "wrong" database accidentally being inserted into a VTP domain.
- Interaction with VTP version 1 and VTP version 2.
- Ability to be configured on a per-port basis.

Note With software release 8.1(1), all VTP versions can be configured on a per-port basis.

- Provides the ability to propagate the VLAN database *and* other databases. VTP version 3 is a collection of protocol *instances*, with each instance handling one database that is associated with a given feature. VTP version 3 handles the configuration propagation of multiple databases (features) independent of one another by running multiple instances of the protocol.

Note In software releases 8.1(x) and 8.2(x), the only supported database propagation is for the VLAN database. In software release 8.3(1), support is added to propagate the MST database.

These sections describe VTP version 3:

- VTP Version 3 Authentication, page 10-13
- VTP Version 3 Per-Port Configuration, page 10-14
- VTP Version 3 Domains, Modes, and Partitions, page 10-14
- VTP Version 3 Modes, page 10-17
- VTP Version 3 Databases, page 10-19

VTP Version 3 Authentication

VTP version 3 introduces an enhancement to the handling of VTP passwords. VTP version 3 allows the configuration of a *primary server*. A VTP version 3 server cannot make any configuration changes in the domain without first becoming the primary server for the domain. The VTP version 3 authentication enhancements are as follows:

- If no password is configured or if a password is configured the same way as in VTP version 1 or VTP version 2 (without using the **hidden** or **secret** keywords), the following occurs:
  - A switch can become the primary server and configure the domain with no restriction.
  - The password appears in the configuration.

  This enhancement is equivalent to the existing VTP version 1 and VTP version 2 levels of security.
- If a password is configured as hidden using the **hidden** password configuration option, the following occurs:
  - The password does not appear in plain text in the configuration; the *secret* hexadecimal format of the password is saved in the configuration.

– If you try to configure the switch as a primary server, you are prompted for the password. If your password matches the secret password, the switch becomes a primary server allowing you to configure the domain.

For more information on configuring the passwords, see the "Configuring VTP Version 3 Passwords" section on page 10-26.

VTP Version 3 Per-Port Configuration

Note With software release 8.1(1), all VTP versions can be configured on a per-port basis.

VTP version 3 allows you to disable the protocol on a per-port basis. If a trunk is connected to a switch or server that is not trusted and is not supposed to interact with the VTP domain, it is now possible to drop incoming VTP packets and prevent VTP advertisements on a particular trunk. This configuration option has no impact on other protocols.

For more information on the per-port configuration options, see the "Disabling VTP Version 3 on a Per-Port Basis" section on page 10-28.

VTP Version 3 Domains, Modes, and Partitions

This section describes how the domains, modes, and partitions are handled in VTP version 3 as compared to VTP versions 2 and 3:

- A VTP version 3 server can be configured as primary or secondary.
- The VTP version 3 modes (server, client, and transparent) are specific to a VTP instance.
- A VTP version 3 domain can be partitioned.

For more information about these features, see these sections:

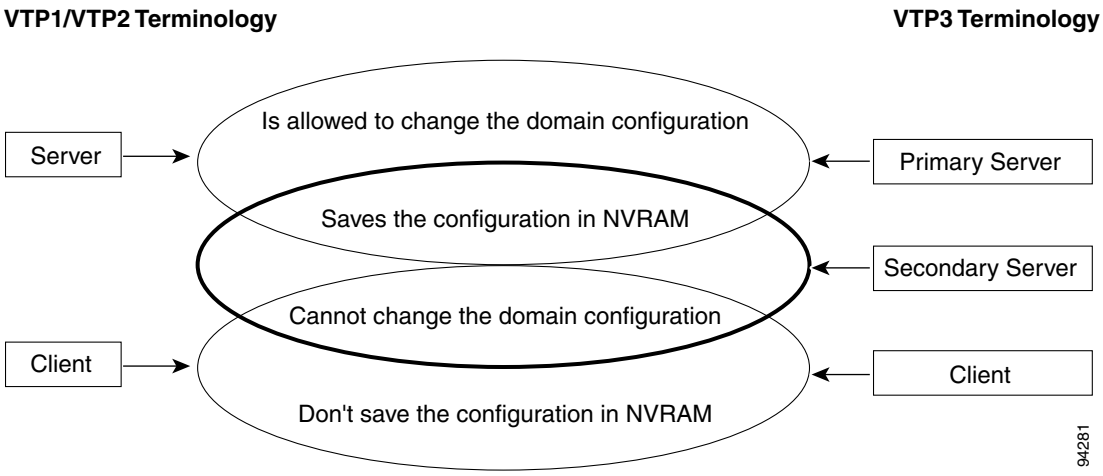
- Primary Servers, Secondary Servers, and Clients, page 10-14
- Partitioned VTP Domains, page 10-15
- Reconfiguring a Partitioned VTP Domain, page 10-16

Primary Servers, Secondary Servers, and Clients

In previous VTP implementations, the VTP server could modify and store the VTP domain configuration in NVRAM, and a VTP client could only receive the configuration from the network and could not save or modify it.

In VTP version 3, the primary server functions the same way as the VTP version 1 and version 2 servers, and the secondary server can store the configuration of the domain but cannot modify it. The concept of client is unchanged in VTP version 3 (see Figure 10-3). The main distinction in VTP version 3 is that the server, client, and transparent modes are specific to a VTP instance. For example, in VTP version 3, it is possible for a switch to be a primary server for one instance and a client for another instance.

Figure 10-3 VTP Version 3: Primary Servers, Secondary Servers, and Clients



Partitioned VTP Domains

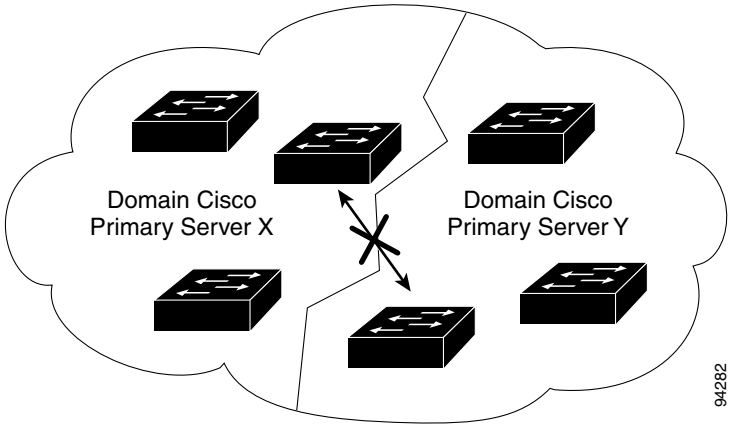
VTP version 3 restricts the configuration rights for a domain to a unique primary server, as follows:

- VTP configuration is possible only on a primary server.
- The identifier (ID) of the primary server that generated the database is attached to the VTP advertisements.
- A VTP switch keeps the ID of a primary server and accepts VTP database updates from its current primary server only.

Because the ID of a primary server is always sent with the VTP configuration, any switch that has a configuration also knows the corresponding primary server. As in VTP version 1 and VTP version 2, the switches that do not have a VTP configuration accept the first configuration that they receive (if it passes the optional authentication scheme that is described in the "VTP Version 3 Authentication" section on page 10-13). VTP version 3 switches *lock* on the primary server that generated their configuration and only listen to further VTP database updates from this primary server. This process differs significantly from VTP version 1 and VTP version 2 where a switch would always accept a superior configuration from a neighbor in the same domain. A VTP version 3 switch accepts only a superior configuration that is from the same domain *and* that is generated by the same primary server.

Ideally, there should be only one primary server in a VTP version 3 domain, but if there are several, the domain is partitioned in groups following the update of their respective primary server (see Figure 10-4). In Figure 10-4, the Cisco VTP domain is partitioned between switches accepting server X or server Y as a primary server. The switches that are from different partitions do not exchange database information even though they are part of the same domain. If server X changes the VTP configuration, only the left partition of the network accepts it.

Figure 10-4 VTP Version 3: Partitioned VTP Domain



Partitions exist because of discrepancies in the domain configuration that cannot automatically be resolved by VTP. Partitions are the result of a misconfiguration or an independent configuration of a temporarily disconnected part of the domain. This behavior of VTP version 3 protects the domain from accepting a conflicting configuration after the insertion of a misconfigured switch. If a new switch is added to a domain, it will not propagate its configuration until you manually designate it as the new primary server.

The primary server for a VLAN instance can be a different server than the primary server that is set for an MST instance. Using two primary servers in this case does not cause partitions.

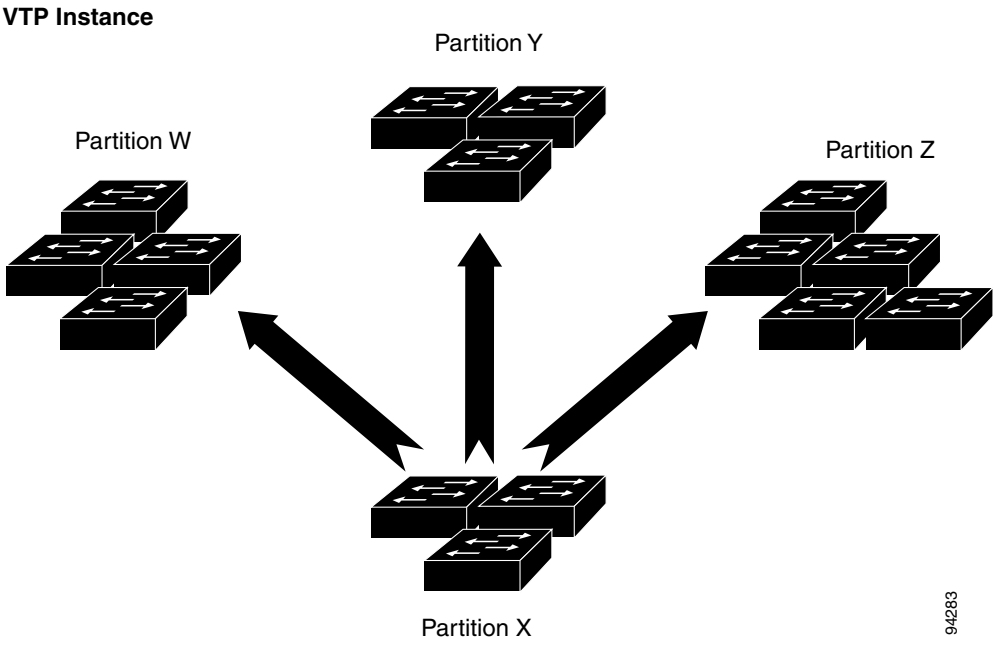
For information on using the *takeover* mechanism to reconfigure partitioned VTP domains, see the "Reconfiguring a Partitioned VTP Domain" section on page 10-16.

Reconfiguring a Partitioned VTP Domain

Partitioning of a VTP domain is specific to the instance; one instance may be partitioned while another might not be partitioned. In VTP version 3, you are required to remove any partitions because the protocol cannot determine which primary server has the final, desired configuration. Figure 10-5 shows a VTP domain that has been divided into four partitions for one specific VTP instance.

In Figure 10-5, server X has the correct configuration for the domain. To reconfigure this partitioned VTP domain, you need to issue a takeover message from server X to the entire domain, advertising server X as the new primary server for this specific instance. All switches in the domain will lock onto primary server X and accept only the instance configuration updates that are initiated by server X. All switches in the domain synchronize their VTP configuration to server X for that instance.

Figure 10-5 VTP Version 3: Reconfiguring a Partitioned VTP Domain



Initiating a takeover is a critical operation due to the following reasons:

- The takeover erases conflicting configurations that are potentially stored on other primary servers in the VTP domain. VTP lists all the switches with conflicting configurations (when you enter the **show vtp conflicts** command) and prompts you for confirmation before taking over (a server has conflicting information if it belongs to the same VTP domain but has a different primary server).
- The takeover leaves this switch (server X in Figure 10-5) as the only primary server controlling the VTP domain.

If you have a hidden password configured, you need to reenter the password to do a takeover. Switches refuse the takeover request if they are not correctly authenticated. If no authentication is enabled, any server is able to take over.

After a takeover, there should be only one primary server controlling the entire VTP domain for a particular instance. If this is not the case, it might be due to the following:

- Some switches may be temporarily disconnected and unreachable when the takeover message is sent.
- The takeover message might be lost on some links (however, the takeover messages are repeated to reduce this risk).

In both cases, you can correct the problem by issuing additional takeover messages.

For more information on configuring a takeover, see the "Configuring a VTP Version 3 Takeover" section on page 10-27.

VTP Version 3 Modes

The default mode for VTP is version 1, server mode. The off mode can be exited only after you configure a VTP domain name on the switch. The "domain discovery" that is used in VTP version 1 and VTP version 2 is not available in VTP version 3.

Switches running VTP version 3 have the following common characteristics:

- They accept only VTP packets from the same VTP domain.
- If they do not have a primary server, they accept the primary server that is associated with the first VTP database that they receive for any instance.
- They accept only a database with a higher revision number from their current primary server.
- If they have a password configured (whether hidden or not hidden), they accept only a new database or a takeover message if it contains the correct password.

VTP version 3 modes are described in the following sections:

- Client Mode, page 10-18
- Server Mode, page 10-18
- Transparent and VTP Off Modes, page 10-19

For more information on configuring modes, see the "Changing VTP Version 3 Modes" section on page 10-23.

Client Mode

VTP version 3 clients are similar to VTP version 1 and VTP version 2 clients as follows:

- A VTP client accepts a VTP configuration from the network but cannot generate or alter the configuration.
- A VTP client stores the VTP configuration that it receives in RAM (not NVRAM). When a VTP client boots, it needs to reacquire the entire configuration that is propagated by VTP, including the identity of the primary server.
- A VTP client that cannot store the entire VTP configuration that is received in an instance to RAM, immediately transitions to transparent mode.

Server Mode

Primary and secondary servers are two types of servers that may exist on a VLAN or VTP instance in the VTP domain.

Secondary Server

When a switch is configured to be a server, it becomes a secondary server by default. As a secondary server, a VTP version 3 switch behaves as a client with the following exceptions:

- A secondary server immediately stores the information that is received through VTP version 3 in NVRAM. This NVRAM is part of the running configuration or startup configuration.
- At startup, a secondary server that has a configuration in NVRAM starts advertising the configuration. The main purpose of a VTP secondary server is to back up the configuration that is propagated over the network.
- Similar to a client, a VTP secondary server cannot modify the VTP configuration.
- A VTP server reverts to client mode if it cannot store the configuration in NVRAM.
- A VTP version 3 secondary server can issue a takeover to become a primary server.

Primary Server

The primary server can initiate or change the VTP configuration. To reach the primary server state, you must issue a successful takeover from the switch. The takeover is propagated to the entire domain. All other potential primary servers in the domain resign to secondary server mode to ensure that there is only one primary server in the VTP domain.

You only need the primary server when the VTP configuration for any instance needs to be modified. A VTP domain can operate with no active primary server because the secondary servers ensure persistence of the configuration over reloads. The primary server state is exited due to the following reasons:

- A switch reload.
- A high-availability switchover between the active and redundant supervisor engines.
- A takeover from another server.
- A change in the mode configuration.
- Any VTP domain configuration change (such as version, domain name, or domain password).

Transparent and VTP Off Modes

In VTP version 3, the transparent mode is specific to the instance. The off mode in VTP version 3 is similar to the previous VTP versions and is not specific to an instance. In both modes, you are allowed to configure locally the features that VTP is controlling. This configuration also appears in the running configuration (if applicable). The feature stores its local configuration in the same NVRAM block that is used by VTP. All NVRAM handling for the feature happens through VTP whether or not the switch is transparent to the feature. In VTP transparent mode, all VTP messages that are received by the switch are still flooded. In VTP off mode, the VTP messages are dropped on the trunks.

VTP Version 3 Databases

VTP version 1 and VTP version 2 are tied to VLAN information. VTP version 3 is designed to distribute any kind of configuration (referred to as a database) over a VTP domain.

Note In software releases 8.1(x) and 8.2(x), the only supported database propagation is for the VLAN database. In software release 8.3(1), support is added to propagate the MST database.

These sections describe the VTP version 3 databases:

- Valid Databases, page 10-19
- Database Revision Number, page 10-20
- Interaction with VTP Version 1 and VTP Version 2, page 10-20
- Limitations, page 10-21

Valid Databases

A switch advertises a database only if it is valid. The only way to validate a database is to become the primary server. If a switch modifies a database that has been generated by a primary server (this situation is possible in off or transparent modes), the database is invalid. An invalid database is applied only locally on a switch and is overwritten by any database that is received on the network if the switch is a VTP client or server. Some examples of valid and invalid databases are as follows:

- When you move from VTP version 1 to VTP version 3, the VLAN database and MST database are not deleted but they are marked invalid because they have been generated by a VTP version 1 server, not by a VTP version 3 primary server.
- If you move a VTP version 3 server with a valid database to transparent mode, you can configure the VLAN database and MST database, but as soon as the database is modified, it becomes invalid. This situation prevents the switch from going back to server mode and advertising the database because the valid database that is received from the network overwrites the changes made while in transparent mode. If a server moves to transparent mode and then back to server mode with no changes to the database configuration, its database is still valid.
- If you modify a database on a primary server (such as a VLAN configuration), the database stays valid and is advertised to the rest of the domain. In any mode, when you configure a domain-related parameter (such as the domain name, VTP version, and the authentication method [password]), all the databases become invalidated. In addition to invalidating the databases, configuring a domain-related parameter also reverts a primary server to a secondary server.
- When you change a domain parameter, the switch is inserted into a new domain. To prevent the wrong database from accidentally being inserted into a VTP domain, you cannot insert a switch as a primary server into a new domain because it could potentially erase a valid configuration. Because it has an invalid database, a newly inserted switch in a domain immediately accepts the network configuration instead of erasing it.

Database Revision Number

Each VTP instance is associated with a database revision number. The database revision number is incremented when the value of the database that is covered by the advertised checksum is modified.

When a device receives a VTP advertisement from the same primary server for an instance in the same domain, the following occurs:

- If the database revision number in the advertisement is less than that of the receiving device, the advertisement is ignored and a summary advertisement with the current revision number is transmitted on the trunk on which the original advertisement was received.
- If the database revision number in the advertisement is the same as that of the receiving device, the following occurs:
  - If the checksum of the advertisement is exactly the same as the checksum of the current configuration known to the device, then no action is taken.
  - If the checksum of the advertisement is not exactly the same as the checksum of the current configuration known to the device, the device's configuration is unaffected, but the device indicates to the database manager that a configuration error condition has occurred.
- If the database revision number in the advertisement is greater than that of the receiving device, and the advertisement's checksum and configuration information match, the receiving switch requests the exact subset of databases for which it is not up to date.

The VTP advertisement is regenerated on each of the trunk ports of the device but not on the trunk port on which it was received.

Interaction with VTP Version 1 and VTP Version 2

VTP version 3 interacts with VTP version 1 and VTP version 2 switches as follows:

Note You should configure VTP version 1 and VTP version 2 switches as clients to allow them to work properly with VTP version 3. See the "Limitations" section on page 10-21 for more information.

- A VTP version 3 switch can detect VTP version 1 and VTP version 2 switches and send a scaled-down version of its database on a per-trunk basis in VTP version 2 format only. VTP version 1 switches move to VTP version 2 mode without any configuration assistance.
- A VTP version 3 switch never sends any VTP version 2 packets on a trunk unless it first receives a legacy VTP version 1 or VTP version 2 packet on the trunk. This situation forces legacy neighboring switches to keep advertising their presence on the link. If a VTP version 3 switch does not receive a legacy packet on a trunk for a certain period of time, it is considered to be a VTP version 3-only trunk and does not advertise a scaled-down version of the VLAN database or MST database on the trunk anymore.
- Even when advertising a VTP version 2 database on a trunk, VTP version 3 keeps sending VTP version 3 updates through the port. This situation allows two kinds of neighbors to coexist on the trunk.
- A VTP version 3 switch can modify reserved VLANs 1002–1005; however, these VLANs are set to their default in the scaled-down database in VTP version 2 format.
- A VTP version 3 switch never accepts a configuration from a VTP version 1 or VTP version 2 neighbor.

Limitations

The limitations of VTP version 3 are as follows:

- Two VTP version 3 regions can communicate only over a VTP version 1 and VTP version 2 region in transparent mode.
- Leaving a server in a VTP version 2 region so that it will receive its VTP information from a VTP version 3 region could cause a problem. If a configuration change occurs in the VTP version 1 and VTP version 2 region, the revision of the database may become higher than the one that is generated by the VTP version 3 region, and the updates from the VTP version 3 region may be ignored.

  Note We recommend that you set all switches in the VTP version 1 and VTP version 2 region to client and reset their revision number (do a reload or change the domain name back and forth).

- A VTP version 2 region that is connected to two different VTP version 3 regions may receive contradictory information and keep swapping its database to the VTP version 3 region that has the highest revision number at any given time. We do not recommend this type of configuration.
- Enabling VTP pruning on a VTP version 3 switch enables pruning only on the switch that you enable it on. VTP pruning is not propagated as it is with VTP version 1 and VTP version 2.

Default VTP Version 3 Configuration

Table 10-2 shows the default VTP version 3 configuration.

Table 10-2 VTP Version 3 Default Configuration

Feature	Default Value
VTP domain name	Null
VTP mode	Server
VTP version 3 enable state	Version 1 is enabled
VTP password	None
VTP pruning	Disabled

Configuring VTP Version 3

These sections describe how to configure VTP version 3:

Enabling VTP Version 3

Use the **set vtp version** *version_number* command to specify the VTP version. By default, the VTP version is version 1 and the VTP mode is server mode. You must specify a domain before selecting a VTP version or VTP mode.

To enable VTP version 3, perform this task in privileged mode:

	Task	Command
Step 1	Enable VTP version 3 on the switch.	**set vtp version 3**
Step 2	Verify that VTP version 3 is enabled.	**show vtp domain**

This example shows how to enable VTP version 3 and verify the configuration:

```
Console> (enable) set vtp version 3
VTP version 3 cannot be enabled on a switch with No Domain.
Console> (enable) set vtp domain ENG
VTP domain ENG modified
Console> (enable) set vtp version 3
VTP version 3 Server/Client for VLANDB requires Reduced Mac Address feature to
be enabled (use "set spantree macreduction enable" command)
Console> (enable) set spantree macreduction enable
MAC address reduction enabled
Console> (enable) set vtp version 3
This command will enable VTP version 3 on this switch.
Do you want to continue (y/n) [n]? y
VTP3 domain ENG modified
Console> (enable) show vtp domain
```

```
Version      : running VTP3
Domain Name  : ENG                                Password  : configured
Notifications: disabled                           Switch ID : 00d0.004c.1800

Feature        Mode           Revision    Primary ID     Primary Description
-------------- -------------- ----------- -------------- ----------------------
VLAN           Server         0           0000.0000.0000
MST            Transparent
UNKNOWN        Transparent

Pruning             : disabled
VLANs prune eligible: 2-1000
Console> (enable)
```

Changing VTP Version 3 Modes

Note For more information on VTP version modes, see the "VTP Version 3 Modes" section on page 10-17.

Each database is propagated by an instance of the VTP protocol. As these instances are independent, they can operate in different modes. The **set vtp mode** command allows you to set the mode for a particular VTP instance. The VTP instance is identified by the name of the feature to which it applies. The **set vtp mode** command has been extended to include a *feature* that you specify to identify the database to which the command applies. The **unknown** keyword allows you to configure the behavior of the switch databases that it cannot interpret. (These databases will be features handled by future extensions of VTP version 3.) If you enter the **set vtp mode transparent unknown** command, the packets for the unknown features are flooded through the switch. If you enter the **set vtp mode off unknown** command, the packets are dropped. The "unknown" feature can only be configured with off or transparent modes. The default mode is off for all databases. The mode of the VLAN database and MST database are preserved when VTP versions are changed.

Note In software releases 8.1(x) and 8.2(x), the only supported database propagation is for the VLAN database; therefore, there are no "unknown" databases. In software release 8.3(1), support is added to propagate the MST database.

Configuring a VTP Version 3 Server

When a switch is in VTP version 3 server mode, you can change the VLAN configuration and have it propagate throughout the network. To configure the switch as a VTP version 3 server, perform this task in privileged mode:

	Task	Command
Step 1	Define the VTP domain name.	**set vtp domain** *name*
Step 2	Place the switch in VTP server mode.	**set vtp mode server** {**vlan** | **mst** | **unknown**}
Step 3	(Optional) Set a password for the VTP domain.	**set vtp passwd** *passwd*
Step 4	Verify the VTP configuration.	**show vtp domain**

This example shows how to configure the switch as a VTP VLAN server and verify the configuration:

```
Console> (enable) set vtp mode server vlan
Changing VTP mode for vlan feature
```

```
VTP3 domain map1 modified
Console> (enable) show vtp domain
Version      : running VTP3
Domain Name  : ENG                              Password  : configured
Notifications: disabled                          Switch ID : 00d0.004c.1800

Feature        Mode           Revision    Primary ID     Primary Description
-------------- -------------- ----------- -------------- ----------------------
VLAN           Server         0           0000.0000.0000
MST            Transparent
UNKNOWN        Transparent

Pruning             : disabled
VLANs prune eligible: 2-1000
Console> (enable)
```

This example shows how to configure the switch as a VTP MST server and verify the configuration:

```
Console> (enable) set vtp mode server mst
Changing VTP mode for mst feature
VTP3 domain ENG modified
Console> (enable) show vtp domain
Version      : running VTP3
Domain Name  : ENG                              Password  : configured
Notifications: disabled                          Switch ID : 00d0.004c.1800

Feature        Mode           Revision    Primary ID     Primary Description
-------------- -------------- ----------- -------------- ----------------------
VLAN           Server         0           0000.0000.0000
MST            Server         0           0000.0000.0000
UNKNOWN        Transparent

Pruning             : disabled
VLANs prune eligible: 2-1000
Console> (enable)
```

Configuring a VTP Version 3 Client

When a switch is in VTP client mode, you cannot change the VLAN configuration on the switch. The client switch receives VTP updates from a VTP server in the management domain and modifies its configuration accordingly.

To configure the switch as a VTP version 3 client, perform this task in privileged mode:

	Task	Command
Step 1	Define the VTP domain name.	**set vtp domain** *name*
Step 2	Place the switch in VTP client mode.	**set vtp mode client** [**vlan** | **mst** | **unknown**]
Step 3	Verify the VTP configuration.	**show vtp domain**

This example shows how to configure the switch as a VTP version 3 VLAN client and verify the configuration:

```
Console> (enable) set vtp mode client vlan
Changing VTP mode for vlan feature
VTP3 domain ENG modified
Console> (enable) show vtp domain
Version      : running VTP3
Domain Name  : ENG                              Password  : configured
Notifications: disabled                          Switch ID : 00d0.004c.1800
```

```
Feature        Mode           Revision    Primary ID     Primary Description
-------------- -------------- ----------- -------------- ----------------------
VLAN           Client         0           0000.0000.0000
MST            Server         0           0000.0000.0000
UNKNOWN        Transparent

Pruning             : disabled
VLANs prune eligible: 2-1000
Console> (enable)
```

Configuring VTP Version 3 Transparent Mode

When you configure the switch as VTP transparent, you disable VTP on the switch. A VTP transparent switch does not send VTP updates and does not act on VTP updates that are received from other switches.

Note Network devices in VTP transparent mode do not send VTP join messages. On Catalyst 6500 series switches with trunk connections to network devices in VTP transparent mode, you should configure the VLANs that are used by the transparent-mode network devices or that need to be carried across trunks as pruning ineligible (use the **clear vtp pruneeligible** command).

To disable VTP on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Disable VTP on the switch by configuring it for VTP transparent mode.	**set vtp mode transparent** [**vlan** | **mst** | **unknown**]
Step 2	Verify the VTP configuration.	**show vtp domain**

This example shows how to configure the switch as VTP VLAN transparent and verify the configuration:

```
Console> (enable) set vtp mode transparent vlan
Changing VTP mode for vlan feature
VTP3 domain ENG modified
Console> (enable) show vtp domain
Version      : running VTP3
Domain Name  : ENG                             Password  : configured
Notifications: disabled                        Switch ID : 00d0.004c.1800

Feature        Mode           Revision    Primary ID     Primary Description
-------------- -------------- ----------- -------------- ----------------------
VLAN           Transparent
MST            Server         0           0000.0000.0000
UNKNOWN        Transparent

Pruning             : disabled
VLANs prune eligible: 2-1000
Console> (enable)
```

Disabling VTP Using the Off Mode

When you disable VTP using the off mode, the switch behaves the same as in VTP transparent mode with the exception that VTP advertisements are not forwarded.

To disable VTP using the off mode, perform this task in privileged mode:

	Task	Command
Step 1	Disable VTP using the off mode.	**set vtp mode off**
Step 2	Verify the VTP configuration.	**show vtp domain**

This example shows how to disable VTP using the off mode:

```
Console> (enable) set vtp mode off
Changing VTP mode for all features
VTP3 domain server modified
```

Note Because there is only the VLAN database in software releases 8.1(x) and 8.2(x), using the **set vtp mode off** command without specifying the **vlan** keyword results in the same configuration as using the **vlan** keyword. Note that in software release 8.3(1), support is added to propagate the MST database.

```
Console> (enable) show vtp domain
Version      : running VTP3
Domain Name  : ENG                              Password  : configured
Notifications: disabled                          Switch ID : 00d0.004c.1800

Feature        Mode            Revision    Primary ID     Primary Description
-------------- --------------- ----------- -------------- ----------------------
VLAN           Off
MST            Off
UNKNOWN        Transparent

Pruning             : disabled
VLANs prune eligible: 2-1000
Console> (enable)
```

Configuring VTP Version 3 Passwords

Note For more information on passwords, see the "VTP Version 3 Authentication" section on page 10-13.

In VTP version 3, you can hide the VTP password from the configuration by adding the **hidden** keyword to the password configuration. When you use the **hidden** keyword, the hexadecimal secret key that is generated from the password is shown in the configuration instead of the password in plain text. If you configure a password with the **hidden** keyword, you need to reenter the password to issue a takeover (for details on configuring a takeover, see the "Configuring a VTP Version 3 Takeover" section on page 10-27).

Two different formats of the **set vtp passwd** command can be shown in the configuration: A plain text password or an encrypted hexadecimal secret value. These two formats are exclusive; if you configure a plain text password, it replaces a current secret password, and similarly, if you paste a secret password into the configuration, the initial password is removed.

To configure VTP passwords, perform this task in privileged mode:

	Task	Command
Step 1	Configure a VTP password.	**set vtp passwd** *passwd* {**hidden** | **secret**}
Step 2	Verify the VTP password.	**show config**

This example shows how to configure a VTP password and verify the configuration:

```
Console> (enable) set vtp passwd toto
Generating the secret associated to the password.
VTP3 domain server modified
Console> (enable) show config
.
.
.
set vtp passwd toto
.
.
.
Console> (enable) set vtp passwd toto hidden
Generating the secret associated to the password.
The VTP password will not be shown in the configuration.
VTP3 domain server modified

Console> (enable) show config
.
.
.
set vtp passwd 9fbdf74b43a2815037c1b33aa00445e2 secret
.
.
.
Console> (enable) set vtp passwd toto secret
VTP secret has to be 32 characters in length
Console> (enable)
```

This example shows how to copy the secret, hexadecimal value from the configuration, paste it into the command line, and verify the configuration:

```
Console> (enable) set vtp passwd 9fbdf74b43a2815037c1b33aa00445e2 secret
Setting secret.
VTP3 domain server modified
Console> (enable) show config
.
.
.
set vtp passwd 9fbdf74b43a2815037c1b33aa00445e2 secret
.
.
.
```

Configuring a VTP Version 3 Takeover

Note For more information on takeovers, see the "Reconfiguring a Partitioned VTP Domain" section on page 10-16.

Use the **set vtp primary** [*feature*] [**force**] command to configure a takeover. A takeover allows a secondary server to become a primary server and propagates the primary server's configuration to the entire VTP domain, removing any partitions if applicable.

Note If a password was configured using the **hidden** keyword, you are prompted to reenter it.

If you do not specify the **force** keyword, the switch tries to discover some conflicting servers in the domain. Conflicting servers follow a different primary server than the one in the configuration of the local switch. You are prompted by the local switch for confirmation before proceeding with the takeover. The prompting is necessary because taking over the domain involves overwriting the configuration of any conflicting servers.

If you do not specify the optional *feature* argument, the local switch sends a takeover message for each database for which it is a secondary or a primary server. If you specify a database, the switch takes over only those databases that are associated with the specified feature.

To configure a takeover, perform this task in privileged mode:

	Task	Command
Step 1	Configure a takeover.	**set vtp primary** [*feature*] [**force**]
Step 2	Verify the takeover.	**show vtp domain**

This example shows how to configure a takeover from a secondary switch that has a hidden password configured and verify the configuration:

```
Console> (enable) set vtp primary force
Switch can become primary server for vlan feature only when configured as a server
Switch can become primary server for mst feature only when configured as a server
Console> (enable) set vtp mode server mst
Changing VTP mode for mst feature
VTP3 domain ENG modified
Console> (enable) set vtp mode server vlan
Changing VTP mode for vlan feature
VTP3 domain ENG modified
Console> (enable) set vtp primary force
This switch is becoming primary server for feature vlan.
This switch is becoming primary server for feature mst.
Do you want to continue (y/n) [n]? y
Console> (enable) show vtp domain
Version      : running VTP3
Domain Name  : ENG                              Password  : configured
Notifications: disabled                         Switch ID : 00d0.004c.1800

Feature        Mode           Revision    Primary ID     Primary Description
-------------- -------------- ----------- -------------- ----------------------
VLAN           Primary Server 1           00d0.004c.1800
MST            Primary Server 1           00d0.004c.1800
UNKNOWN        Transparent

Pruning             : disabled
VLANs prune eligible: 2-1000
Console> (enable)
```

Disabling VTP Version 3 on a Per-Port Basis

Note For more information on disabling VTP version 3 on a per-port basis, see the "VTP Version 3 Per-Port Configuration" section on page 10-14.

Use the **set port vtp** *mod/port* {**enable** | **disable**} command to enable or disable all VTP interaction on a per-port basis. This capability might be used on trunks leading to nontrusted hosts. When a port is disabled, no VTP packets are sent on the port, and any VTP packets that are received on the port are dropped. By default, VTP is enabled and advertisements are received and sent on all trunks.

To disable VTP on a per-port basis, perform this task in privileged mode:

	Task	Command
Step 1	Disable VTP on a per-port basis.	**set port vtp** *mod/port* {**enable** | **disable**}
Step 2	Verify the change.	**show port vtp**

This example shows how to disable VTP on a per-port basis and verify the configuration:

```
Console> (enable) set port vtp 3/1-2 disable
VTP is disabled on ports 3/1-2.
Console> (enable) show port vtp 3
Port      VTP Status
--------  ----------
 3/1      disabled
 3/2      disabled
 3/3      enabled
 3/4      enabled
Console> (enable)
```

VTP Version 3 show Commands

Use the **show vtp** {**conflicts** | **devices** | **domain** | **statistics**} command to show other **devices** in the domain or devices in the domain with conflicting (**conflicts**) configurations. Use the **domain** keyword to display information that is specific to the VTP domain. Use the **statistics** keyword to display VTP statistics. Switches in transparent or off mode are not part of the VTP domain and do not respond to requests. In addition, the clients or servers that do not have a valid database do not respond to requests.

CHAPTER 11

Configuring VLANs

This chapter describes how to configure VLANs for the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Understanding How VLANs Work

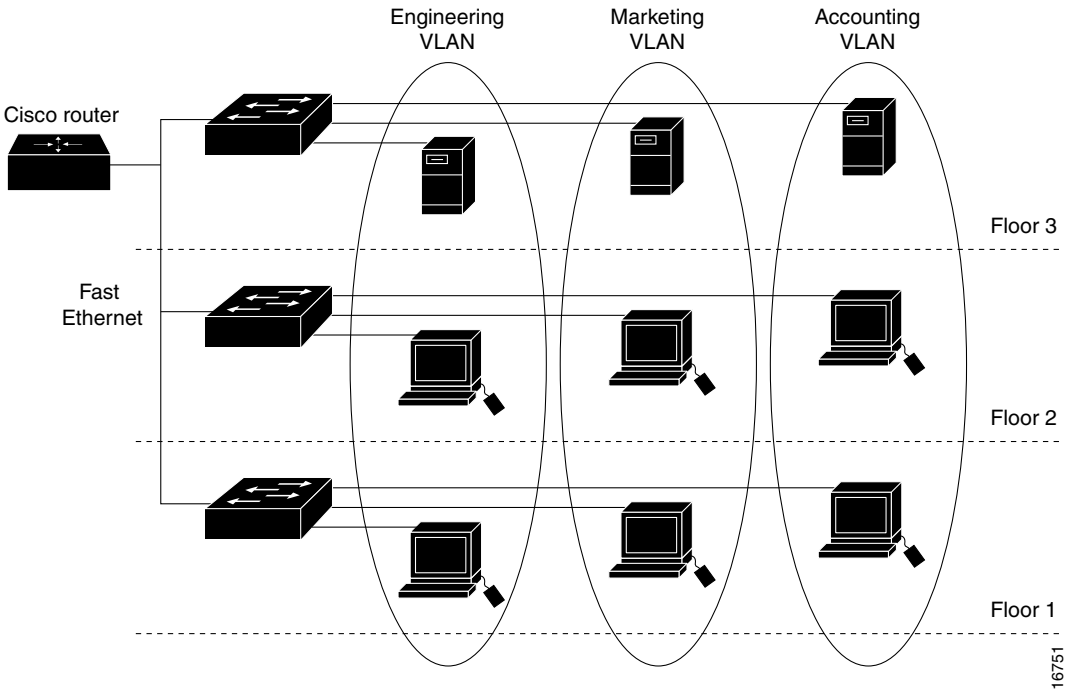
A VLAN is a group of end stations with a common set of requirements, independent of their physical location. A VLAN has the same attributes as a physical LAN but allows you to group the end stations even if they are not located physically on the same LAN segment.

A VLAN allows you to group the ports on a switch to limit the unicast, multicast, and broadcast traffic flooding. The flooded traffic that originates from a particular VLAN is flooded only out the ports that belong to that VLAN.

Figure 11-1 shows an example of VLANs that are segmented into logically defined networks.

These sections describe VLANs:

Figure 11-1 VLANs as Logically Defined Networks



VLANs are often associated with the IP subnetworks. For example, all the end stations in a particular IP subnet belong to the same VLAN. The traffic between the VLANs must be routed. Port VLAN membership on the switch is assigned manually on a port-by-port basis. When you assign the switch ports to the VLANs using this method, it is known as port-based, or static, VLAN membership.

The in-band (sc0) interface of a switch can be assigned to any VLAN, so that you can access another switch on the same VLAN directly without a router. Only one IP address at a time can be assigned to the in-band interface. If you change the IP address and assign the interface to a different VLAN, the previous IP address and VLAN assignment are overwritten.

VLAN Ranges

Catalyst 6500 series switches support 4096 VLANs in accordance with the IEEE 802.1Q standard. These VLANs are organized into two ranges; you use each range slightly differently. Some of these VLANs are propagated to other switches in the network when you use a management protocol, such as the VLAN Trunking Protocol (VTP). Other VLANs are not propagated and you must configure them on each applicable switch.

VLANs are divided into the following two ranges:

- Normal-range VLANs: 1–1023

- Extended-range VLANs: 1024–4094

Note With VTP version 3, you can manage VLANs 1006–4094. These VLANs are propagated with VTP version 3.

Configurable VLAN Parameters

Whenever you create or modify VLANs 2–1005, you can set the parameters as follows:

Note Ethernet VLANs 1 and 1025–4094 can use the defaults only.

Note With software release 8.3(1) and later releases, you can name all user VLANs. This capability is independent of any VTP version or mode.

- VLAN number
- VLAN name
- VLAN type: Ethernet, FDDI, FDDINET, Token Ring Bridge Relay Function (TrBRF), or Token Ring Concentrator Relay Function (TrCRF)
- VLAN state: active or suspended
- Multi-Instance Spanning Tree Protocol (MISTP) instance
- Private VLAN type: primary, isolated, community, two-way community, or none
- Security Association Identifier (SAID)
- Maximum transmission unit (MTU) for the VLAN
- Ring number for FDDI and TrCRF VLANs
- Bridge identification number for TrBRF VLANs
- Parent VLAN number for TrCRF VLANs
- STP type for TrCRF VLANs: IEEE, IBM, or auto
- VLAN to use when translating from one VLAN media type to another (VLANs 1–1005 only); requires a different VLAN number for each media type
- Source routing bridge mode for Token Ring VLANs: source-routing bridge (SRB) or source-routing transparent bridge (SRT)
- Backup for TrCRF VLAN
- Maximum hops VLAN All-Routes Explorer frames (ARE) and Spanning Tree Explorer frames (STE) for Token Ring
- Remote Switched Port Analyzer (RSPAN)

Default VLAN Configuration

Table 11-1 shows the default VLAN configuration for the Catalyst 6500 series switches.

Table 11-1 VLAN Default Configuration

Feature	Default Value
Native (default) VLAN	VLAN 1
Port VLAN assignments	All ports assigned to VLAN 1 Token Ring ports assigned to VLAN 1003 (trcrf-default)
VLAN state	Active
MTU size	1500 bytes 4472 bytes for Token Ring VLANs
SAID value	100,000 plus the VLAN number (for example, the SAID for VLAN 8 is 100008, and the SAID for VLAN 4050 is 104050)
Pruning eligibility	VLANs 2–1000 are pruning eligible; VLANs 1025-4094 are not pruning eligible
MAC address reduction	Disabled
Spanning-tree mode	PVST+
Default FDDI VLAN	VLAN 1002
Default FDDI NET VLAN	VLAN 1004
Default Token Ring TrBRF VLAN	VLAN 1005 (trbrf-default) with bridge number 0F
Default Token Ring TrCRF VLAN	VLAN 1003 (trcrf-default)
Spanning Tree Protocol (STP) version for TrBRF VLAN	IBM
VLAN port-provisioning verification	Disabled
TrCRF bridge mode	SRB
Remote switched port analyzer (RSPAN)	Disabled

Configuring VLANs on the Switch

These sections describe how to configure user VLANs 1–4094:

- Normal-Range VLAN Configuration Guidelines, page 11-5
- Creating Normal-Range VLANs, page 11-5
- Modifying Normal-Range VLANs, page 11-6

Note You cannot configure or modify normal-range VLAN 1.

Normal-Range VLAN Configuration Guidelines

This section describes the guidelines for creating and modifying the user VLANs in your network:

- The default VLAN type is Ethernet; if you do not specify a VLAN type, the VLAN will be an Ethernet VLAN.
- If you wish to use VTP to maintain global VLAN configuration information on your network, configure VTP before you create any normal-range VLANs. See Chapter 10, "Configuring VTP" for configuring VTP. (You cannot use VTP to manage extended-range VLANs 1025–4094.)

Note With VTP version 3, you can manage VLANs 1006–4094. These VLANs are propagated with VTP version 3.

- The FlexWAN modules and routed ports automatically allocate a number of VLANs for their own use, starting at VLAN 1025. If you use these devices, you must allow for the number of VLANs required.

Creating Normal-Range VLANs

You can create one VLAN at a time or you can create a range of VLANs with a single command. If you create a range of VLANs, you cannot specify a name; the VLAN names must be unique.

To create a normal-range VLAN, perform this task in privileged mode:

	Task	Command
Step 1	Create a normal-range Ethernet VLAN.	**set vlan** *vlan* [**name** *name*] [**said** *said*] [**mtu** *mtu*] [**translation** *vlan*]
Step 2	Verify the VLAN configuration.	**show vlan** [*vlan*]

This example shows how to create the normal-range VLANs and verify the configuration when the switch is in Per VLAN Spanning Tree + (PVST+) mode:

```
Console> (enable) set vlan 500-520
Vlan 500 configuration successful
Vlan 501 configuration successful
Vlan 502 configuration successful
Vlan 503 configuration successful
Vlan 520 configuration successful

Console> (enable) show vlan 500-520
VLAN Name                             Status    IfIndex Mod/Ports, Vlans
---- -------------------------------- --------- ------- ------------------------
500                                    active    342
501                                    active    343
502                                    active    344
503                                    active    345
520                                    active    362
```

```
VLAN Type  SAID       MTU   Parent RingNo BrdgNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ ------ ---- -------- ------ ------
500  enet  100500     1500  -      -      -      -    -        0      0
501  enet  100501     1500  -      -      -      -    -        0      0
502  enet  100502     1500  -      -      -      -    -        0      0
503  enet  100503     1500  -      -      -      -    -        0      0
520  enet  100520     1500  -      -      -      -    -        0      0
VLAN AREHops STEHops Backup CRF
---- ------- ------- ----------
Console> (enable)
```

Modifying Normal-Range VLANs

To modify the VLAN parameters on an existing normal-range VLAN, perform this task in privileged mode:

	Task	Command
Step 1	Modify an existing normal-range VLAN.	**set vlan** *vlan* [**name** *name*] [**state** {**active** | **suspend**}] [**said** *said*] [**mtu** *mtu*] [**translation** *vlan*]
Step 2	Verify the VLAN configuration.	**show vlan** [*vlan*]

Configuring Extended-Range VLANs on the Switch

These sections explain how to configure extended-range VLANs 1025–4094:

Extended-Range VLAN Configuration Guidelines

This section describes the guidelines for creating extended-range VLANs 1024–4094:

- You can create only Ethernet-type VLANs in the extended range.
- You must enable MAC address reduction in order to use the extended-range VLANs.
- You can only create and delete the extended-range VLANs from the CLI or SNMP.
- You cannot use VTP to manage these VLANs; they must be statically configured on each switch.

Note With VTP version 3, you can manage VLANs 1006–4094. These VLANs are propagated with VTP version 3. For configuration purposes, the extended range consists of VLANs 1025–4094.

- You cannot use the extended-range VLANs if you have dot1q-to-isl mappings.

- You can configure the private VLAN parameters and RSPAN for the extended-range VLANs; however, all other parameters for the extended-range VLANs use the system defaults only.
- The switch may allocate a block of VLANs from the extended range for internal purposes; for example, the switch may allocate the VLANs for the routed ports or FlexWAN modules. The block of VLANs is always allocated starting from VLAN 1006 up. If you have any VLANs within the range that are required by the FlexWAN module, all of the VLANs that are required will not be allocated, because the VLANs are never allocated from the user's VLAN area.

Caution The FlexWAN modules and routed ports automatically allocate a sequential block of internal VLANs starting at VLAN 1006. If you use these devices, you *must* allow the required number of VLANs for them. If not enough VLANs are available for the FlexWAN module, some ports may not work. Refer to the *Catalyst 6500 Series and Cisco 7600 Series Router FlexWAN Module Installation and Configuration Note* for more information.

Caution If you move a FlexWAN module from one slot to another on the same switch, it will allocate another block of VLANs without deleting the previous block. You should reboot the switch if you move the FlexWAN module.

Creating Extended-Range VLANs

To create the extended-range VLANs, you must first enable MAC address reduction, which provides the IDs for the extended-range VLANs. After you enable MAC address reduction, you cannot disable it as long as any extended-range VLANs exist.

Note If you wish to use the extended-range VLANs and you have existing 802.1Q-to-ISL mappings in your system, you must delete the mappings. See the "Deleting 802.1Q-to-ISL VLAN Mappings" section on page 11-10 for more information.

Note With software release 8.1(1) and later releases, you can name the extended-range VLANs. This capability is independent of any VTP version or mode.

To enable MAC address reduction and create an Ethernet VLAN in the extended range, perform this task in privileged mode:

	Task	Command
Step 1	Enable MAC address reduction.	**set spantree macreduction** {**enable** | **disable**}
Step 2	Create a VLAN.	**set vlan** *vlan*
Step 3	Verify the VLAN configuration.	**show vlan** [*vlan*]

This example shows how to enable MAC address reduction and create an extended-range Ethernet VLAN:

```
Console> (enable) set spantree macreduction enable
MAC address reduction enabled
Console> (enable) set vlan 2000
Vlan 2000 configuration successful
Console> (enable) show vlan 2000
VLAN Name                             Status    IfIndex Mod/Ports, Vlans
---- -------------------------------- --------- ------- ------------------------
2000 VLAN2000                         active    61

VLAN Type  SAID       MTU   Parent RingNo BrdgNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ ------ ---- -------- ------ ------
2000 enet  102000     1500  -      -      -      -    -        0      0

VLAN Inst DynCreated  RSPAN
---- ---- ---------- --------
2000 -    static     disabled
Console> (enable)
```

This example shows how to display a summary of active, suspended, and extended VLANs:

```
Console> (enable) show vlan summary

Vlan status    Count  Vlans
-------------  -----  ---------------------------------------------------
VTP Active       504   1-100,102-500,1000,1002-1005

VTP Suspended      1   101

Extended           1   2000
Console> (enable)
```

Mapping VLANs to VLANs

Note To configure the VLAN mappings on a per-port or per-ASIC basis, see the "Configuring VLAN Mappings on a Per-Port or Per-ASIC Basis" section on page 11-14.

Note With software release 8.3(1) and later releases, the global VLAN mapping feature is not needed because ISL trunks now support the entire VLAN range (1 to 4094).

You can map the VLANs from the 802.1Q trunks that are connected to the VLANs on the non-Cisco devices to the ISL trunks that are connected to the other VLANs on the Catalyst 6500 series switches.

Note If you map the 802.1Q VLANs to the ISL VLANs, you can retain the mappings from a previous Catalyst 6500 series software release but you cannot use the extended-range VLANs.

This section describes how to map the VLANs to VLANs:

Mapping 802.1Q VLANs to ISL VLANs

Your network might have non-Cisco devices that are connected to the Catalyst 6500 series switches through the 802.1Q trunks.

The valid range of the user-configured Inter-Switch Link (ISL) VLANs is 1–1000 (and 1002–1005) and 1025–4094. The valid range of VLANs that is specified in the IEEE 802.1Q standard is 0–4095. In a network environment with the non-Cisco devices that are connected to the Cisco switches through the 802.1Q trunks, you can map the 802.1Q VLAN numbers that are greater than 1000 to the ISL VLAN numbers. If you use any VLANs in the extended range (1025–4094) for dot1q mappings, you cannot use any of the extended-range VLANs for any other purpose.

The 802.1Q VLANs in the range 1–1000 are automatically mapped to the corresponding ISL VLAN. The 802.1Q VLAN numbers greater than 1000 must be mapped to an ISL VLAN in order to be recognized and forwarded by the Cisco switches.

These restrictions apply when mapping the 802.1Q VLANs to the ISL VLANs:

- The global VLAN mapping feature and the per-port/per-ASIC VLAN mapping features (see the "Configuring VLAN Mappings on a Per-Port or Per-ASIC Basis" section on page 11-14) are mutually exclusive; only one feature can be enabled at any time.
- If there are any extended-range VLANs present on the switch, you cannot map any new 802.1Q VLANs-to-ISL VLANs.
- You can configure up to eight 802.1Q-to-ISL VLAN mappings on the switch.
- You can only map the 802.1Q VLANs to the Ethernet-type ISL VLANs.
- Do not enter the native VLAN of any 802.1Q trunk in the mapping table.
- When you map an 802.1Q VLAN to an ISL VLAN, the traffic on the 802.1Q VLAN corresponding to the mapped ISL VLAN is blocked. For example, if you map 802.1Q VLAN 2000 to ISL VLAN 200, the traffic on 802.1Q VLAN 200 is blocked.
- The VLAN mappings are local to each switch. Make sure that you configure the same VLAN mappings on all appropriate switches in the network.

To map an 802.1Q VLAN to an ISL VLAN, perform this task in privileged mode:

	Task	Command
Step 1	Map an 802.1Q VLAN to an ISL Ethernet VLAN. The valid range for *dot1q_vlan* is 1001–4095. The valid range for *isl_vlan* is 1–1000.	**set vlan mapping dot1q** *dot1q_vlan* **isl** *isl_vlan*
Step 2	Verify the VLAN mapping.	**show vlan mapping**

This example shows how to map 802.1Q VLANs 2000, 3000, and 4000 to ISL VLANs 200, 300, and 400, and verify the configuration:

```
Console> (enable) set vlan mapping dot1q 2000 isl 200
Vlan mapping successful
Console> (enable) set vlan mapping dot1q 3000 isl 300
Vlan mapping successful
```

```
Console> (enable) set vlan mapping dot1q 4000 isl 400
Vlan mapping successful
Console> (enable) show vlan mapping
802.1q vlan     ISL vlan        Effective
-----------------------------------------
2000            200             true
3000            300             true
4000            400             true
Console> (enable)
```

Deleting 802.1Q-to-ISL VLAN Mappings

To delete an 802.1Q-to-ISL VLAN mapping, perform this task in privileged mode:

	Task	Command
Step 1	Delete an 802.1Q-to-ISL VLAN mapping.	**clear vlan mapping dot1q** {*dot1q_vlan* | **all**}
Step 2	Verify the VLAN mapping.	**show vlan mapping**

This example shows how to delete the VLAN mapping for 802.1Q VLAN 2000:

```
Console> (enable) clear vlan mapping dot1q 2000
Vlan 2000 mapping entry deleted
Console> (enable)
```

This example shows how to delete all 802.1Q-to-ISL VLAN mappings:

```
Console> (enable) clear vlan mapping dot1q all
All vlan mapping entries deleted
Console> (enable)
```

Allocating Internal VLANs

The VLANs are classified as either user VLANs or internal VLANs. A user VLAN can be any VLAN from 1–4094 created by a user. The internal VLANs are the VLANs that are used by the software features that require the dedicated VLANs in order to function. The internal VLANs are allocated by the VLAN Manager as needed using VLANs 1006–4094. The internal VLANs are allocated in ascending order, starting at VLAN 1006. You should assign the user VLANs as close to VLAN 4094 as possible in order to avoid conflicts between the user VLANs and the internal VLANs.

Note Because the number of available VLANs is fixed, make sure that a sufficient number of VLANs remains available for internal VLAN allocation after you have assigned the user VLANs.

Assigning Switch Ports to a VLAN

A VLAN that is created in a management domain remains unused until you assign one or more switch ports to the VLAN. You can create a new VLAN and then specify the module and ports later, or you can create the VLAN and specify the module and ports in a single step.

Note Make sure that you assign the switch ports to a VLAN of the proper type. For example, assign the Ethernet, Fast Ethernet, and Gigabit Ethernet ports to the Ethernet-type VLANs.

To assign one or more switch ports to a VLAN, perform this task in privileged mode:

	Task	Command
Step 1	Assign one or more switch ports to a VLAN.	**set vlan** *vlan mod/port*
Step 2	Verify the port VLAN membership.	**show vlan** [*vlan*] **show port** [*mod*[*/port*]]

This example shows how to assign the switch ports to a VLAN and verify the assignment:

```
Console> (enable) set vlan 560 4/10
VLAN 560 modified.
VLAN 1 modified.
VLAN  Mod/Ports
---- ------------------------
560   4/10

Console> (enable) show vlan 560
VLAN Name                             Status    IfIndex Mod/Ports, Vlans
---- -------------------------------- --------- ------- -------------------------
560  Engineering                      active    348     4/10
VLAN Type  SAID       MTU   Parent RingNo BrdgNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ ------ ---- -------- ------ ------
560  enet  100560     1500  -      -      -      -    -        0      0
VLAN AREHops STEHops Backup CRF
---- ------- ------- ----------
Console> (enable) show port 4/10
Port  Name               Status     Vlan       Duplex Speed Type
----- ------------------ ---------- ---------- ------ ----- ------------
4/10                     connected  560        a-half a-100 10/100BaseTX

Port  AuxiliaryVlan AuxVlan-Status
----- ------------- --------------
 4/10  none          none

.
.
.

Last-Time-Cleared
--------------------------
Tue Jun 6 2000, 16:45:18
Console> (enable)
```

Enabling or Disabling VLAN Port-Provisioning Verification

When VLAN port-provisioning verification is enabled, you must specify the VLAN name *in addition to* the VLAN number when assigning the switch ports to the VLANs. Because you are required to specify both the VLAN name and the VLAN number, this verification feature helps to ensure that the ports are not inadvertently placed in the wrong VLAN.

When the feature is enabled, you can still create new VLANs by entering the **set vlan** *vlan mod/port* command but you cannot add additional ports to the VLAN without specifying both the VLAN number and the VLAN name. The feature does not affect assigning ports to VLANs using other features such as SNMP, dynamic VLANs, and 802.1X. VLAN port-provisioning verification is disabled by default.

To enable or disable VLAN port-provisioning verification, perform this task in privileged mode:

	Task	Command
Step 1	Enable or disable VLAN port-provisioning verification.	**set vlan verify-port-provisioning** {**enable** | **disable**}
Step 2	Verify the VLAN port-provisioning verification status.	**show vlan verify-port-provisioning**

This example shows how to enable VLAN port-provisioning verification:

```
Console> (enable) set vlan verify-port-provisioning enable
vlan verify-port-provisioning feature enabled
Console> (enable)
```

This example shows how to verify the status of VLAN port-provisioning verification:

```
Console> (enable) show vlan verify-port-provisioning
Vlan Verify Port Provisioning feature enabled
Console> (enable)
```

This example shows how to create VLAN 150 and add port 3/16 (with VLAN port-provisioning verification enabled):

```
Console> (enable) set vlan 150 3/16
Vlan 150 configuration successful
VLAN 150 modified.
VLAN 1 modified.
VLAN  Mod/Ports
---- ----------------------
150   3/16
Console> (enable)
```

This example shows what happens when you try to add port 3/17 to VLAN 150 with VLAN port-provisioning verification enabled:

```
Console> (enable) set vlan 150 3/17
Port Provisioning Verification is enabled on the switch.
To move port(s) into the VLAN, use 'set vlan <vlan> <port> <vlan_name>' command.
Console> (enable)
```

This example shows how to add port 3/17 to VLAN 150 with VLAN port-provisioning verification enabled:

```
Console> (enable) set vlan 150 name Eng
VTP advertisements transmitting temporarily stopped,
and will resume after the command finishes.
Vlan 150 configuration successful
Console> (enable)

Console> (enable) set vlan 150 3/17 Eng
VLAN 150 modified.
VLAN 1 modified.
VLAN  Mod/Ports
---- ----------------------
150   3/16-17
Console> (enable)
```

Deleting a VLAN

This section describes the guidelines for deleting the VLANs:

- When you delete a normal-range Ethernet VLAN in VTP server mode, the VLAN is removed from all switches in the VTP domain.
- When you delete a normal-range VLAN in VTP transparent mode, the VLAN is deleted only on the current switch.
- You can delete an extended-range VLAN only on the switch where it was created.
- You cannot delete the default VLANs.
- To delete a Token Ring TrBRF VLAN, you must first reassign its child TrCRFs to another parent TrBRF, or delete the child TrCRFs.

Caution When you delete a VLAN, any ports that are assigned to that VLAN become inactive. Such ports remain associated with the VLAN (and are inactive) until you assign them to a new VLAN.

You can delete a single VLAN or a range of VLANs. To delete a VLAN on the switch, perform this task in privileged mode:

Task	Command
Delete a VLAN.	**clear vlan** *vlan*

This example shows how to delete a VLAN (in this case, the switch is a VTP server):

```
Console> (enable) clear vlan 500
This command will deactivate all ports on vlan(s) 500
Do you want to continue(y/n) [n]?y
Vlan 500 deleted
Console> (enable)

This command will deactivate all ports on vlan(s) 10
All ports on normal range vlan(s) 10
will be deactivated in the entire management domain.
Do you want to continue(y/n) [n]?
```

Configuring VLAN Mappings on a Per-Port or Per-ASIC Basis

These sections describe how to configure VLAN mapping on a per-port or per-ASIC basis:

Understanding VLAN Mapping

With software release 8.4(1) and later releases, VLAN mapping has been enhanced to allow you to map *any* type of VLAN to any other type of VLAN without any VLAN range restrictions. VLAN mapping is now configurable on a per-port or per-ASIC basis.

Note Before software release 8.4(1), VLAN mapping was configured globally. For detailed information, see the "Mapping VLANs to VLANs" section on page 11-8.

Configuration Guidelines and Restrictions

This section describes the configuration guidelines and restrictions for configuring VLAN mapping:

- With VLAN mapping, you have the following options depending on the type of ASIC on the switching module or supervisor engine (for the individual module ASIC specifics, see Table 11-2):
  - VLAN mapping is not supported.
  - Per-port VLAN mapping is supported.
  - Per-ASIC VLAN mapping *without* the ability to enable or disable VLAN mapping on an individual port basis is supported.
  - Per-ASIC VLAN mapping *with* the ability to enable or disable VLAN mapping on an individual port basis is supported.
- If a module does not support per-port VLAN mapping and supports only per-ASIC VLAN mapping, VLAN mapping is applied to all the ports in the ASIC. If you change the mapping for any port in the ASIC, the change is applied to all the ports in the ASIC.
- Global VLAN mapping

  The global VLAN mapping feature (see the "Mapping VLANs to VLANs" section on page 11-8) and the per-port/per-ASIC VLAN mapping features are mutually exclusive; only one feature can be enabled at any time.

  If global VLAN mapping is configured for any of the VLANs and you try to configure per-port/per-ASIC VLAN mapping, the command is rejected and an error message is displayed. Conversely, if per-port/per-ASIC VLAN mapping is configured for any of the VLANs and you try to configure global VLAN mapping, the command is rejected and an error message is displayed.

Global VLAN mapping supports a maximum of eight VLANs. If VLAN X is mapped to VLAN Y, VLAN Y is mapped to a discarded VLAN internally. Per-port/per-ASIC VLAN mapping does not work that way. If VLAN X is mapped to VLAN Y, all the internally switched traffic to a port on VLAN Y is mapped to VLAN X.

- VLAN mapping is applied in both directions. For example, if port P has a VLAN mapping of VLAN x to VLAN y, all the traffic received by port P on VLAN X is mapped and processed in VLAN Y, and all the traffic internally tagged with VLAN Y that leaves port P, is tagged with VLAN X.
- EtherChannel

  VLAN mapping is supported on EtherChannels, both PAgP and LACP. If you enable or disable VLAN mapping on one port of the channel, the feature is enabled or disabled on all the ports in the channel. Similarly, if you configure a VLAN mapping on one port in the channel, the mapping is applied to all ports in the channel.

  All the ports in the EtherChannel must have the same port ASIC capability in terms of VLAN mapping. If you try to configure a VLAN mapping on an EtherChannel where some of the ports in the channel do not have the same port ASIC capabilities, the command is rejected.
- SPAN and RSPAN

  If per-port VLAN mapping is enabled on a port, the port ASIC changes the source VLAN to the translated VLAN. Any SPAN configuration works on the translated VLAN.

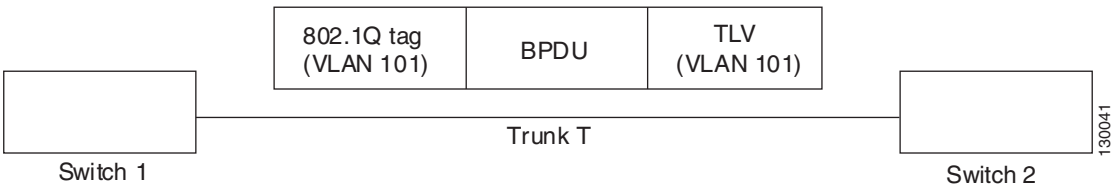
  The RSPAN VLAN cannot be translated; you must not configure the RSPAN VLAN to be mapped to any VLAN. Similarly, the translated VLAN cannot be used as an RSPAN VLAN.
- Spanning tree

  In the PVST+ implementation, spanning-tree BPDUs are tagged with a TLV of “VLAN ID” on each trunk port. This TLV helps spanning tree in determining the port VLAN ID consistency. In PVST+ and Rapid-PVST+, this VLAN ID is equal to the spanning-tree instance number (the VLAN ID).

  With Shared Spanning Tree Protocol (SSTP), be careful when per-port/per-ASIC VLAN mapping is enabled on a port. For example, in Figure 11-2, switch 1 and switch 2 are connected using trunk T that carries VLAN 101. On switch 2, per-port/per-ASIC VLAN mapping is enabled on trunk port P and one of the mappings is VLAN 101 to VLAN 202. As shown in Figure 11-2, on the trunk link, the BPDU has the 802.1Q VLAN and the TLV VLAN as VLAN 101. When this BPDU reaches port P, its 802.1Q VLAN is changed to VLAN 202 because of the mapping but the TLV VLAN still remains VLAN 101. When the BPDU reaches the spanning-tree process, spanning tree concludes that the VLAN 101 BPDU is received on VLAN 202 and thinks that it is inconsistent and reports this as an inconsistent port.

  To correct this problem, the spanning tree processes this BPDU in VLAN 202 and the TLV VLAN is mapped to the translated VLAN and checked for consistency. When that occurs, the spanning-tree instance 101 of switch 1 is merged with the spanning-tree instance 202 of switch 2. This process is also done on the transmit side.

Figure 11-2 Understanding VLAN Mapping and Spanning Tree



Tip Before designing your spanning-tree topology, you should take into account the way in which VLANs are merged. You should clear the source VLAN from the port on which VLAN mapping is enabled and clear the translated VLAN from the neighboring end. Doing this ensures that the source VLAN of the customer port and the translated VLAN of the provider port are merged.

Table 11-2 Per-Module Port ASIC VLAN Mapping Capabilities

Module	Maximum Number of Per-Port VLAN Mappings Supported	Capabilities/Limitations
WS-X6548-RJ-45 WS-X6548-RJ-21 WS-X6148X2-RJ-45 WS-X6148X2-45AF WS-X6196-RJ-21[1]	32	Per-ASIC VLAN mapping. Mapping can be enabled or disabled on a per-port basis on ISL trunks. Mapping is always on for 802.1Q trunks and there is no way to disable it. Mapping is supported for ISL and 802.1Q trunks.
WS-X6K-S2U-MSFC2 WS-X6K-S2-MSFC2 WS-X6K-S2-PFC2 WS-SUP720-3B WS-SUP720-3BXL WS-SUP720 WS-X6516A-GBIC[2] WS-X6516-GE-TX	32	Per-ASIC VLAN mapping. Mapping can be enabled or disabled on individual ports in the ASIC. Supports any-to-any type of VLAN translation. Supported only on 802.1Q trunks.[3]
WS-X6748-SFP[4] WS-X6724-SFP WS-X6748-GE-TX	128	Per-ASIC VLAN mapping. Mapping can be enabled or disabled on individual ports in the ASIC. Supports any-to-any type of VLAN translation. Mapping is supported for ISL and 802.1Q trunks.
WS-X6148A-GE-TX WS-X6148A-GE-45A WS-X6148-FE-SFP WS-X6148A-RJ-45 WS-X6148A-45AF WS-X6704-10GE[5]	8	Per-port VLAN mapping. Supports any-to-any type of VLAN translation. Mapping is supported for ISL and 802.1Q trunks.
WS-X6502-10GE	16	Per-port VLAN mapping. Supports any-to-any type of VLAN translation. Supported only on 802.1Q trunks.
WS-SUP32-GE-3B	16	Per-port VLAN mapping. Supports any-to-any type of VLAN translation. Mapping is supported for ISL and 802.1Q trunks.

1. WS-X6196-RJ-21 does not have per-ASIC VLAN mapping. VLAN mapping is per-two ASICs: Ports 1 through 96 (instead of only 48 ports per ASIC).
2. WS-X6516A-GBIC does not have per-ASIC VLAN mapping. VLAN mapping is per-two ASICs: Ports 1 through 8 and ports 9 through 16 (instead of only 4 ports per ASIC).
3. The ASICs in these modules have the following limitation: When dot1q-all-tagged is disabled, VLAN translation does not occur for packets transmitted on the native VLAN.

4. WS-X6748-SFP does not have per-ASIC VLAN mapping. VLAN mapping is per-two ASICs: Ports 1 through 24 and ports 25 through 48 (instead of only 12 ports per ASIC).
5. WS-X6704-10GE: Mapping can be enabled or disabled on individual ports in the ASIC. 128 per-port VLAN mappings supported.

Enabling or Disabling VLAN Mapping on an Individual Port

Note Before using the **set port vlan-mapping** command to configure VLAN mapping on an individual port, you must enable port VLAN mapping by entering the **set port vlan-mapping** *mod/port* **enable** command.

Enter the **set port vlan-mapping** *mod/port* {**enable** | **disable**} command to enable or disable VLAN mapping on an individual port. VLAN translation occurs only when the mapping is enabled and the port is trunking. For the ASICs that support VLAN mapping only on a per-ASIC basis, but with the ability to enable or disable VLAN mapping on an individual port basis, this command is applied to the port configuration only and not to the ASIC. If you disable VLAN mapping, the mapping is still preserved. VLAN mapping is disabled by default.

To enable or disable VLAN mapping on an individual port, perform this task in privileged mode:

	Task	Command
Step 1	Enable or disable VLAN mapping on an individual port.	**set port vlan-mapping** *mod/port* {**enable** | **disable**}
Step 2	Display VLAN mapping configuration.	**show port vlan-mapping** [*mod* | *mod/port*]

This example shows how to enable VLAN mapping on an individual port:

```
Console>(enable) set port vlan-mapping 7/1 enable
VLAN mapping enabled on port 7/1.
Console>(enable)
```

Configuring VLAN Mapping on an Individual Port

Note Before using the **set port vlan-mapping** command, you must enable the port VLAN mapping by entering the **set port vlan-mapping** *mod/port* **enable** command.

Note The source VLAN is the trunk VLAN (external to the switch) and the translated VLAN is internal to the switch.

Enter the **set port vlan-mapping** *mod/port source-vlan-id translated-vlan-id* command to configure VLAN mapping on an individual port. This command causes the traffic on the *source-vlan-id* to be translated to the *translated-vlan-id*. All traffic that is internally tagged with the *translated-vlan-id* is tagged with the *source-vlan-id* before leaving the port. The VLAN translation occurs only if the port is trunking. This command accepts the full range of ports.

To configure VLAN mapping on an individual port, perform this task in privileged mode:

	Task	Command
Step 1	Enable the port VLAN mapping.	**set port vlan-mapping** *mod/port* {**enable** | **disable**}
Step 2	Configure VLAN mapping on an individual port.	**set port vlan-mapping** *mod/port source-vlan-id translated-vlan-id*
Step 3	Display VLAN mapping configuration.	**show port vlan-mapping** [*mod* | *mod/port*]

This example shows how to enable the port VLAN mapping and configure VLAN mapping on an individual port. In this example, module 7 is the 48-port 10/100/1000 switching module (WS-X6748-GE-TX). This module supports per-ASIC VLAN mapping; 1 ASIC supports 12 ports.

```
Console>(enable) set port vlan-mapping 7/1 enable
VLAN mapping enabled on port 7/1.
Console>(enable) set port vlan-mapping 7/1 2002 3003
VLAN 2002 mapped to VLAN 3003 on ports 7.1. 7/1-12.
Console>(enable) show port vlan-mapping 7/1
Mod/Port Source VLAN Translated VLAN State       Max Allowed (Current) Entries
-------- ----------- --------------- ----------- ----------------------------
7/1      2002        3003            Enabled     128 (1)
Console>(enable)
```

In this example, module 5 is the 1-port 10GBASE-E serial 10-Gigabit Ethernet module (WS-X6502-10GE). This module supports per-port VLAN mapping.

```
Console>(enable) set port vlan-mapping 5/1 2002 3003
VLAN 2002 mapped to VLAN 3003 on port 5/1.
Console>(enable)
```

In this example, module 7 is the 48-port 10/100/1000 switching module (WS-X6748-GE-TX). This module supports per-ASIC VLAN mapping; 1 ASIC supports 12 ports. In this example, ports 7/1-4 are part of an EtherChannel.

```
Console>(enable) set port vlan-mapping 7/1 2002 3003
VLAN 2002 mapped to VLAN 3003 on ports 7/1-12.
Console>(enable)
```

In this example, module 7 and module 8 are the 48-port 10/100/1000 switching modules (WS-X6748-GE-TX). These modules support per-ASIC VLAN mapping; 1 ASIC supports 12 ports. In this example, ports 7/1-4 and ports 8/1-4 are part of an EtherChannel.

```
Console>(enable) set port vlan-mapping 7/1 2002 3003
VLAN 2002 mapped to VLAN 3003 on ports 7/1-12,8/1-12.
Console>(enable)
```

Clearing the VLAN Mapping

Enter the **clear port vlan-mapping** command to clear the VLAN mapping on an individual port, on all ports, or on a specific source VLAN ID. On some modules, VLAN mapping is supported on a per-ASIC basis; the mapping is not stored on a per-port basis. For these modules, entering the **clear port vlan-mapping** *mod/port* command clears the VLAN mapping on all ports on the ASIC. When you enter a *source_vlan_id* argument, only the VLAN mapping for that source VLAN is cleared from the VLAN mapping table of the specified port or ASIC (if ASIC-based port).

To clear VLAN mapping, perform this task in privileged mode:

Task	Command
Clear VLAN mapping.	**clear port vlan-mapping** *mod/port* **all** **clear port vlan-mapping** *mod/port* [*source-vlan-id*] **clear port vlan-mapping all**

This example shows how to clear the VLAN mapping from port 7/1:

```
Console>(enable) clear port vlan-mapping 7/1 2002
VLAN mapping for VLAN 2002 removed from port 7/1-12.
Console>(enable)
```

Displaying the VLAN Mapping Information

Enter the **show port vlan-mapping** [*mod* | *mod/port*] command to display the VLAN mapping information.

To display VLAN mapping information, perform this task in normal mode:

Task	Command
Display the VLAN mapping information.	**show port vlan-mapping** [*mod*	*mod/port*]

This example shows how to display the VLAN mapping information for port 7/1:

```
Console>(enable) show port vlan-mapping 7/1
Mod/Port Source VLAN Translated VLAN State       Max Allowed (Current) Entries
-------- ----------- --------------- ----------- ------------------------------
7/1      2002        3003            Enabled     128 (1)
Console>(enable)
```

Note Enter the **show port capabilities** [*mod* | *mod/port*] command to display the mapping type (per port or per ASIC) for each port. This command also displays the maximum allowed mappings for each port.

Configuring Private VLANs on the Switch

These sections describe how the private VLANs work:

- Understanding How Private VLANs Work, page 11-20
- Private VLAN Configuration Guidelines, page 11-21
- Creating a Primary Private VLAN, page 11-25
- Viewing the Port Capability of a Private VLAN Port, page 11-27
- Deleting a Private VLAN, page 11-28
- Deleting an Isolated, Community, or Two-Way Community VLAN, page 11-29
- Deleting a Private VLAN Mapping, page 11-29
- Private VLAN Support on the MSFC, page 11-30

Understanding How Private VLANs Work

The private VLANs provide the Layer-2 isolation between the ports within the same private VLAN on the Catalyst 6500 series switches. The ports that belong to a private VLAN are associated with a common set of supporting VLANs that are used to create the private VLAN structure.

The three types of private VLAN ports are as follows:

- Promiscuous—This port communicates with all other private VLAN ports and is the port that you use to communicate with routers, LocalDirector, backup servers, and administrative workstations.

  Note If a broadcast or multicast packet comes from the promiscuous port, it is sent to all the ports in the private VLAN domain, that is, to all the community and isolated ports.

- Isolated—This port has complete Layer 2 separation from the other ports within the same private VLAN with the exception of the promiscuous port.
- Community—These ports communicate among themselves and with their promiscuous ports. These ports are isolated at Layer 2 from all other ports in other communities or isolated ports within their private VLAN.

Privacy is granted at Layer 2 by blocking the outgoing traffic to all isolated ports. All isolated ports are assigned to an isolated VLAN where this hardware function occurs. The traffic that is received from an isolated port is forwarded to all promiscuous ports only.

A private VLAN has four distinct classifications: a single primary VLAN, a single isolated VLAN, and a series of community or two-way community VLANs.

You must define each supporting VLAN within a private VLAN structure before you can configure the private VLAN:

- Primary VLAN—Conveys the incoming traffic from the promiscuous port to all other promiscuous, isolated, community, and two-way community ports.
- Isolated VLAN—Used by the isolated ports to communicate to the promiscuous ports. The traffic from an isolated port is blocked on all adjacent ports within its private VLAN and can only be received by its promiscuous ports.
- Community VLAN—A unidirectional VLAN that is used by a group of community ports to communicate among themselves and transmit the traffic to outside the private VLAN through the designated promiscuous port.
- Two-way community VLAN—A bidirectional VLAN that is used by a group of community ports to communicate among themselves and to and from the community ports from and to the Multilayer Switch Feature Card (MSFC).

  Note With software release 6.2(1) and later releases, you can use the two-way community VLANs to perform an inverse mapping from the primary VLAN to the secondary VLAN when the traffic crosses the boundary of a private VLAN through an MSFC promiscuous port. Both the outbound and inbound traffic can be carried on the same VLAN allowing the VLAN-based features such as the VLAN access control lists (VACLs) to be applied in both directions on a per-community (per-customer) basis.

To create a private VLAN, you assign two or more normal VLANs in the normal VLAN range: one VLAN is designated as a primary VLAN, and a second VLAN is designated as either an isolated, community, or two-way community VLAN. If you choose, you can then designate additional VLANs as separate isolated, community, or two-way community VLANs in this private VLAN. After designating the VLANs, you must bind them together and associate them to the promiscuous port.

You can extend the private VLANs across multiple Ethernet switches by trunking the primary, isolated, and any community or two-way community VLANs to the other switches that support the private VLANs.

In an Ethernet-switched environment, you can assign an individual VLAN and associated IP subnet to each individual or common group of stations. The servers require only the ability to communicate with a default gateway to gain access to the end points outside the VLAN itself. By incorporating these stations, regardless of ownership, into one private VLAN, you can do the following:

- Designate the server ports as isolated to prevent any interserver communication at Layer 2.
- Designate the ports to which the default gateway(s), backup server, or LocalDirector are attached as promiscuous to allow all stations to have access to these gateways.
- Reduce VLAN consumption. You only need to allocate one IP subnet to the entire group of stations because all stations reside in one common private VLAN.

On an MSFC port or a nontrunk promiscuous port, you can remap as many isolated or community VLANs as desired; however, while a nontrunk promiscuous port can remap to only one primary VLAN, an MSFC port can only connect an MSFC router. With a nontrunk promiscuous port, you can connect a wide range of devices as "access points" to a private VLAN. For example, you can connect a nontrunk promiscuous port to the "server port" of a LocalDirector to remap a number of isolated or community VLANs to the server VLAN so that the LocalDirector can load balance the servers that are present in the isolated or community VLANs, or you can use a nontrunk promiscuous port to monitor and/or back up all the private VLAN servers from an administration workstation.

Note A two-way community VLAN can be mapped only on the MSFC promiscuous port (it cannot be mapped on nontrunk or other types of promiscuous ports).

Private VLAN Configuration Guidelines

This section describes the guidelines for configuring private VLANs:

Note In this section, the term *community VLAN* is used for both the unidirectional community VLANs and two-way community VLANs unless specifically differentiated.

Note If VLAN port-provisioning verification is enabled, you must specify the VLAN name *in addition to* the VLAN number when assigning the switch ports to the primary and secondary VLANs. For more information, see the "Enabling or Disabling VLAN Port-Provisioning Verification" section on page 11-12.

- Designate one VLAN as the primary VLAN.
- You have the option of designating one VLAN as an isolated VLAN, but you can use only one isolated VLAN.

- You have the option of using the private VLAN communities, but you need to designate a community VLAN for each community.
- Bind the isolated and/or community VLAN(s) to the primary VLAN and assign the isolated or community ports. You will achieve these results:
  - Isolated/community VLAN spanning-tree properties are set to those of the primary VLAN.
  - VLAN membership becomes static.
  - The access ports become the host ports.
  - BPDU guard protection is activated.
- Set up the automatic VLAN translation that maps the isolated and community VLANs to the primary VLAN on the promiscuous port(s). Set the nontrunk ports or the MSFC ports as promiscuous ports.
- You must set VTP to transparent mode.

Note This restriction does not apply with VTP version 3.

- After you configure a private VLAN, you cannot change the VTP mode to client or server mode, because VTP does not support the private VLAN types and mapping propagation.
- You can configure the VLANs as primary, isolated, or community only if no access ports are currently assigned to the VLAN. Enter the **show port** command to verify that the VLAN has no access ports that are assigned to it.
- A primary VLAN can have one isolated VLAN and/or multiple communities that are associated with it.
- An isolated or community VLAN can have only one primary VLAN that is associated with it.
- The private VLANs can use VLANs 2–1000 and 1025–4096.
- If you delete either the primary or secondary VLAN, the ports that are associated with the VLAN become inactive.
- When configuring the private VLANs, note the hardware and software interactions as follows:
  - You cannot use the inband port, sc0, in a private VLAN.

Note With software release 6.3(1) and later releases, you can configure the sc0 port as a private VLAN port; however, you cannot configure the sc0 port as a promiscuous port.

  - You cannot set the private VLAN ports to trunking mode, channeling, or have dynamic VLAN memberships, with the exception of the MSFC ports that always have trunking activated.
  - You cannot set the ports belonging to the same ASIC where one port is set to trunking or promiscuous mode or is a SPAN destination and another port is set to isolated or community port for the modules listed in Table 11-3. (Note that a promiscuous port can be defined in the same ASIC as a trunk port but not within the same ASIC as an isolated or community port.)

  If you attempt such a configuration, a warning message displays and the command is rejected.

Note Software release 8.6(1) and later releases provide support for configuring 802.1X with private VLANs. For more information, see the "Configuring 802.1X Authentication with Private VLANs" section on page 40-41.

Table 11-3 Modules with Ports Listed by ASIC Groups

Module Number	Description	Ports by ASIC
WS-X6224-100FX-MT	24-port 100BASE-FX multimode, MT-RJ	Ports 1–12 Ports 13–24
WS-X6324-100FX-SM WS-X6324-100FX-MM	24-port 100BASE-FX single mode or multimode, MT-RJ	Ports 1–12 Ports 13–24
WS-X6024-10FL-MT	24-port 10BASE-FL, MT-RJ	Ports 1–12 Ports 13–24
WS-X6248-TEL WS-X6248A-TEL WS-X6348-RJ-21(V) WS-X6148-RJ-21(V) WS-X6148-21AF	48-port 10/100BASE-TX, RJ-21	Ports 1–12 Ports 13–24 Ports 25–36 Ports 37–48
WS-X6348-RJ-45 WS-X6348-RJ-45(V) WS-X6248-RJ-45 WS-X6248A-RJ-45 WS-X6148-RJ-45(V) WS-X6148-45AF	48-port 10/100BASE-TX, RJ-45	Ports 1–12 Ports 13–24 Ports 25–36 Ports 37–48
WS-X6148-GE-TX WS-X6148V-GE-TX WS-X6148-GE-45AF WS-X6548-GE-TX WS-X6548V-GE-TX WS-X6548-GE-45AF	48-port 10/100/1000BASE-TX, RJ-45	Ports 1–8 Ports 9–16 Ports 17–24 Ports 25–32 Ports 33–40 Ports 41–48

- The isolated and community ports should run the BPDU guard features to prevent the spanning-tree loops due to misconfigurations.
- The primary VLANs and associated isolated/community VLANs must have the same spanning-tree configuration. This configuration maintains the consistent spanning-tree topologies between the associated primary, isolated, and community VLANs and avoids possible connectivity loss. These priorities and parameters automatically propagate from the primary VLAN to the isolated and community VLANs.
- You can create the private VLANs that run in MISTP mode as follows:
  - If you disable MISTP, any change to the configuration of a primary VLAN propagates to all corresponding isolated and community VLANs, and you cannot change the isolated or community VLANs.
  - If you enable MISTP, you can only configure the MISTP instance with the primary VLAN. The changes will be applied to the primary VLAN and will propagate to the isolated and community VLANs.

- In the networks with some switches using MAC address reduction, and others not using MAC address reduction, the STP parameters do not necessarily propagate to ensure that the spanning-tree topologies match. You should manually check the STP configuration to ensure that the primary, isolated, and community VLANs' spanning-tree topologies match.
- If you enable MAC address reduction on a Catalyst 6500 series switch, you might want to enable MAC address reduction on all the switches in your network to ensure that the STP topologies of the private VLANs match. Otherwise, in a network where private VLANs are configured, if you enable MAC address reduction on some switches and disable it on others (mixed environment), you will have to use the default bridge priorities to make sure that the root bridge is *common* to the primary VLAN and to all its associated isolated and community VLANs. Be consistent with the ranges that are employed by the MAC address reduction feature regardless of whether it is enabled on the system. MAC address reduction allows only discrete levels and uses *all* intermediate values internally as a range. You should disable a root bridge with private VLANs and MAC address reduction, and configure the root bridge with any priority higher than the highest priority *range* that is used by any nonroot bridge.
- BPDU guard mode is system wide and is enabled after you add the first port to a private VLAN.
- You cannot configure a destination SPAN port as a private VLAN port and vice versa.
- A source SPAN port can belong to a private VLAN.
- You can use VLAN-based SPAN (VSPAN) to span the primary, isolated, and community VLANs together, or use SPAN on only one VLAN to separately monitor the egress or ingress traffic.
- You cannot use a remote SPAN VLAN (RSPAN) for a private VLAN.
- You cannot enable EtherChannel on the isolated, community, or promiscuous ports.
- You can apply the different VACLs and the quality of service (QoS) ACLs to the primary, isolated, and community VLANs.

Note For information on configuring the ACLs, see the "Configuring ACLs on Private VLANs" section on page 15-43.

- You need to configure the output ACLs on both the two-way community VLANs and the primary VLAN in order to be applied to all outgoing traffic from the MSFC.
- If you map a Cisco IOS ACL to a primary VLAN, the Cisco IOS ACL automatically maps to the associated isolated and community VLANs.
- You cannot map the Cisco IOS ACLs to an isolated or community VLAN.
- You cannot use policy-based routing (PBR) on a private VLAN interface. You get an error message if you try to apply a policy to a private VLAN interface using the **ip policy route-map** *route_map_name* command.
- You cannot set a VLAN to a private VLAN if the VLAN has the dynamic access control entries (ACEs) configured.
- You can stop the Layer 3 switching on an isolated or community VLAN by destroying the binding of that VLAN with its primary VLAN. Deleting the corresponding mapping is not sufficient.

Creating a Primary Private VLAN

To create a primary private VLAN, perform this task in privileged mode:

	Task	Command
Step 1	Create the primary private VLAN.	**set vlan** *vlan* **pvlan-type primary**
Step 2	Set the isolated, community, or two-way community VLAN(s).	**set vlan** *vlan* **pvlan-type** {**isolated** | **community** | **twoway-community**}
Step 3	Bind the isolated, community, or two-way community VLAN(s) to the primary VLAN.	**set pvlan** *primary_vlan* {*isolated_vlan* | *community_vlan* | *twoway_community_vlan*}
Step 4	Associate the isolated, community, or two-way community port(s) to the primary private VLAN.	**set pvlan** *primary_vlan* {*isolated_vlan* | *community_vlan* | *twoway_community_vlan*} [*mod/ports* | **sc0**]
Step 5	Map the isolated, community, or two-way community VLAN to the primary private VLAN on the promiscuous port.	**set pvlan mapping** *primary_vlan* {*isolated_vlan* | *community_vlan* | *twoway_community_vlan*} *mod/ports*
Step 6	Verify the primary private VLAN configuration.	**show pvlan** [*vlan*] **show pvlan mapping**

Note You can bind the isolated, community, or two-way community port(s) and associated isolated, community, or two-way community VLANs to the private VLAN by entering the **set pvlan** *primary_vlan* {*isolated_vlan* | *community_vlan* | *twoway_community_vlan*} *mod/port* command.

Note The ports do not have to be on the same switch as long as the switches are trunk connected and the private VLAN has not been removed from the trunk.

Note If you are using the MSFC for your promiscuous port in your private VLAN, use 15/1 as the MSFC *mod/por*t number if the supervisor engine is in slot 1, or use 16/1 if the supervisor engine is in slot 2.

Note You must enter the **set pvlan** command everywhere that a private VLAN needs to be created, which includes the switches with the isolated, community, or two-way community ports, the switches with the promiscuous ports, and all *intermediate* switches that need to carry the private VLANs on their trunks. On the edge switches that do not have any isolated, community, two-way community, or promiscuous ports (typically, the access switches with no private ports), you do not need to create the private VLANs and you can prune the private VLANs from the trunks for security reasons.

This example shows how to specify VLAN 7 as the primary VLAN:

```
Console> (enable) set vlan 7 pvlan-type primary
Vlan 7 configuration successful
Console> (enable)
```

This example shows how to specify VLAN 901 as the isolated VLAN and VLANs 902 and 903 as community VLANs:

```
Console> (enable) set vlan 901 pvlan-type isolated
Vlan 901 configuration successful
Console> (enable) set vlan 902 pvlan-type community
Vlan 902 configuration successful
Console> (enable) set vlan 903 pvlan-type community
Vlan 903 configuration successful
Console> (enable)
```

This example shows how to bind VLAN 901 to primary VLAN 7 and assign port 4/3 as the isolated port:

```
Console> (enable) set pvlan 7 901 4/3
Successfully set the following ports to Private Vlan 7,901: 4/3
Console> (enable)
```

This example shows how to bind VLAN 902 to primary VLAN 7 and assign ports 4/4 through 4/6 as the community port:

```
Console> (enable) set pvlan 7 902 4/4-6
Successfully set the following ports to Private Vlan 7,902:4/4-6
Console> (enable)
```

This example shows how to bind VLAN 903 to primary VLAN 7 and assign ports 4/7 through 4/9 as the community ports:

```
Console> (enable) set pvlan 7 903
Successfully set association between 7 and 903.
Console> (enable) set pvlan 7 903 4/7-9
Successfully set the following ports to Private Vlan 7,903:4/7-9
Console> (enable)
```

This example shows how to map the isolated/community VLAN to the primary VLAN on the promiscuous port, 3/1, for each isolated or community VLAN:

```
Console> (enable) set pvlan mapping 7 901 3/1
Successfully set mapping between 7 and 901 on 3/1
Console> (enable) set pvlan mapping 7 902 3/1
Successfully set mapping between 7 and 902 on 3/1
Console> (enable) set pvlan mapping 7 903 3/1
Successfully set mapping between 7 and 903 on 3/1
```

This example shows how to verify the private VLAN configuration:

```
Console> (enable) show vlan 7
VLAN Name                             Status    IfIndex Mod/Ports, Vlans
---- -------------------------------- --------- ------- ------------------------
7    VLAN0007                         active    35      4/4-6
VLAN Type  SAID       MTU   Parent RingNo BrdgNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ ------ ---- -------- ------ ------
7    enet  100010     1500  -      -      -      -    -        0      0
VLAN DynCreated  RSPAN
---- ---------- --------
7    static     disabled
VLAN AREHops STEHops Backup CRF 1q VLAN
---- ------- ------- ---------- -------
Primary Secondary Secondary-Type    Ports
------- --------- ----------------- ------------------
7       901        Isolated          4/3
7       902        Community         4/4-6
7       903        Community         4/7-9
```

```
Console> (enable) show vlan 902
VLAN Name                             Status    IfIndex Mod/Ports, Vlans
---- -------------------------------- --------- ------- ------------------------
902  VLAN0007                         active    38      4/4-6
VLAN Type  SAID       MTU   Parent RingNo BrdgNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ ------ ---- -------- ------ ------
7    enet  100010     1500  -      -      -      -    -        0      0
VLAN DynCreated  RSPAN
---- ---------- --------
7    static     disabled
VLAN AREHops STEHops Backup CRF 1q VLAN
---- ------- ------- ---------- -------
Primary Secondary Secondary-Type    Ports
------- --------- ----------------- -----------------
7       902       Isolated          4/4-6

Console> (enable) show pvlan
Primary Secondary Secondary-Type Ports
------- --------- -------------- ------------
7       901       isolated       4/3
7       902       community      4/4-6
7       903       community      4/7-9

Console> (enable) show pvlan mapping
Port  Primary  Secondary
----- -------- ----------
3/1    7        901-903
Console> (enable) show port
Port  Name               Status     Vlan       Duplex Speed Type
----- ------------------ ---------- ---------- ------ ----- ------------
...truncated output...
 4/3                     notconnect 7,901       half   100 100BaseFX MM
 4/4                     notconnect 7,902       half   100 100BaseFX MM
 4/5                     notconnect 7,902       half   100 100BaseFX MM
 4/6                     notconnect 7,902       half   100 100BaseFX MM
 4/7                     notconnect 7,903       half   100 100BaseFX MM
 4/8                     notconnect 7,903       half   100 100BaseFX MM
 4/9                     notconnect 7,903       half   100 100BaseFX MM
... truncated output...
```

Viewing the Port Capability of a Private VLAN Port

You can view the port capability of a port in a private VLAN by entering the **show pvlan capability** *mod/port* command.

This example shows the port capability for several ports in the following configuration:

```
Console> (enable) set pvlan 10 20
Console> (enable) set pvlan mapping 10 20 3/1
Console> (enable) set pvlan mapping 10 20 5/2
Console> (enable) set trunk 5/1 desirable isl 1-1005,1025-4094

Console> (enable) show pvlan capability 5/20
Ports 5/13 - 5/24 are in the same ASIC range as port 5/20.

Port 5/20 can be made a private vlan port.

Console> (enable) show pvlan
Primary Secondary Secondary-Type Ports
------- --------- -------------- ------------
10      20        isolated
```

```
Console> (enable) show pvlan capability 3/1
Port 3/1 cannot be made a private vlan port due to:
-----------------------------------------------------
Promiscuous ports cannot be made private vlan ports.

Console> (enable) show pvlan capability 5/1
Ports 5/1 - 5/12 are in the same ASIC range as port 5/1.

Port 5/1 cannot be made a private vlan port due to:
-----------------------------------------------------
Trunking ports are not Private Vlan capable.
Conflict with Promiscuous port(s) : 5/2

Console> (enable) show pvlan capability 5/2
Ports 5/1 - 5/12 are in the same ASIC range as port 5/2.

Port 5/2 cannot be made a private vlan port due to:
-----------------------------------------------------
Promiscuous ports cannot be made private vlan ports.
Conflict with Trunking port(s) : 5/1

Console> (enable) show pvlan capability 5/3
Ports 5/1 - 5/12 are in the same ASIC range as port 5/3.

Port 5/3 cannot be made a private vlan port due to:
-----------------------------------------------------
Conflict with Promiscuous port(s) : 5/2
Conflict with Trunking port(s) : 5/1

Console> (enable) show pvlan capability 15/1
Port 15/1 cannot be made a private vlan port due to:
-----------------------------------------------------
Only ethernet ports can be added to private vlans.
```

Deleting a Private VLAN

You can delete a private VLAN by deleting the primary VLAN. If you delete a primary VLAN, all bindings to the primary VLAN are broken, all ports in the private VLAN become inactive, and any related mappings on the promiscuous port(s) are deleted.

To delete a private VLAN, perform this task in privileged mode:

Task	Command
Delete a primary VLAN.	**clear vlan** *primary_vlan*

This example shows how to delete primary VLAN 7:

```
Console> (enable) clear vlan 7
This command will de-activate all ports on vlan 7
Do you want to continue(y/n) [n]?y
Vlan 7 deleted
Console> (enable)
```

Deleting an Isolated, Community, or Two-Way Community VLAN

If you delete an isolated, community, or two-way community VLAN, the binding with the primary VLAN is broken, any isolated, community, or two-way community ports that are associated to the VLAN become inactive, and any related mappings on the promiscuous port(s) are deleted.

To delete a VLAN on the switch, perform this task in privileged mode:

Task	Command
Delete an isolated or community VLAN.	**clear vlan** {*isolated_vlan* | *community_vlan* | *twoway_community_vlan*}

This example shows how to delete community VLAN 902:

```
Console> (enable) clear vlan 902
This command will de-activate all ports on vlan 902
Do you want to continue(y/n) [n]?y
Vlan 902 deleted
Console> (enable)
```

Deleting a Private VLAN Mapping

If you delete the private VLAN mapping, the connectivity breaks between the isolated, community, or two-way community ports and the promiscuous port. If you delete all the mappings on a promiscuous port, the promiscuous port becomes inactive. When a private VLAN port is set to inactive, it displays "pvlan-" as its VLAN number in the **show port** output.

You might set a private VLAN port to inactive for the following reasons:

- The primary, isolated, community, or two-way community VLAN to which it belongs is cleared.
- All mappings from a non-MSFC promiscuous port are deleted.
- An error occurs when you are configuring a port as a private VLAN port.

To delete a port mapping from a private VLAN, perform this task in privileged mode:

Task	Command
Delete the port mapping from the private VLAN.	**clear pvlan mapping** primary_vlan {*isolated* | *community* | *twoway-community*} {*mod/ports*}

This example shows how to delete the mapping of VLANs 902 to 901, previously set on ports 3/2 through 3/5:

```
Console> (enable) clear pvlan mapping 901 902 3/2-5
Successfully cleared mapping between 901 and 902 on 3/2-5
Console> (enable)
```

Private VLAN Support on the MSFC

These items describe the private VLAN support on the MSFC:

- Enter the **show pvlan** command to display information about the private VLANs. The **show pvlan** command displays information about the private VLANs only when the primary private VLAN is up.
- Entering the **set pvlan mapping** or the **clear pvlan mapping** command on the supervisor engine generates the MSFC syslog messages. See the following for an example:

```
%PV-6-PV_MSG:Created a private vlan mapping, Primary 100, Secondary 101
%PV-6-PV_MSG:Created a private vlan mapping, Primary 200, Secondary 201
%PV-6-PV_MSG:Purged a private vlan mapping, Primary 100, Secondary 101
```

- Enter the **interface vlan** command to configure the Layer 3 parameters only for the primary private VLANs.
- On the supervisor engine, you cannot create the isolated or community VLANs using the VLAN numbers for which the **interface vlan** commands have been entered on the MSFC.
- The ARP entries that are learned on the Layer 3 private VLAN interfaces are sticky ARP entries (we recommend that you display and verify the private VLAN interface ARP entries).
- For security reasons, the private VLAN interface sticky ARP entries do not age out. Connecting new equipment with the same IP address generates a message and the ARP entry is not created.
- Because the private VLAN interface ARP entries do not age out, you must manually remove the private VLAN interface ARP entries if a MAC address changes.
- You can add or remove the private VLAN ARP entries manually as follows:

```
obelix-rp(config)# no arp 11.1.3.30
IP ARP:Deleting Sticky ARP entry 11.1.3.30

obelix-rp(config)# arp 11.1.3.30 0000.5403.2356 arpa
IP ARP:Overwriting Sticky ARP entry 11.1.3.30, hw:00d0.bb09.266e by hw:0000.5403.2356
```

- Some commands clear and recreate the private VLAN mapping as follows:

```
obelix-rp(config)# xns routing
obelix-rp(config)#
%PV-6-PV_MSG:Purged a private vlan mapping, Primary 100, Secondary 101
%PV-6-PV_MSG:Purged a private vlan mapping, Primary 100, Secondary 102
%PV-6-PV_MSG:Purged a private vlan mapping, Primary 100, Secondary 103
%PV-6-PV_MSG:Created a private vlan mapping, Primary 100, Secondary 101
%PV-6-PV_MSG:Created a private vlan mapping, Primary 100, Secondary 102
%PV-6-PV_MSG:Created a private vlan mapping, Primary 100, Secondary 103
```

Configuring FDDI VLANs on the Switch

To create a new FDDI VLAN, perform this task in privileged mode:

	Task	Command
Step 1	Create a new FDDI or FDDI NET-type VLAN.	**set vlan** *vlan* [**name** *name*] **type** {**fddi** | **fddinet**} [**said** *said*] [**mtu** *mtu*]
Step 2	Verify the VLAN configuration.	**show vlan** [*vlan*]

To modify the VLAN parameters on an existing FDDI VLAN, perform this task in privileged mode:

	Task	Command
Step 1	Modify an existing FDDI or FDDI NET-type VLAN.	**set vlan** *vlan* [**name** *name*] [**state** {**active** | **suspend**}] [**said** *said*] [**mtu** *mtu*]
Step 2	Verify the VLAN configuration.	**show vlan** [*vlan*]

Configuring Token Ring VLANs on the Switch

These sections describe the two Token Ring VLAN types that are supported on the switches running VTP version 2:

- Understanding How Token Ring TrBRF VLANs Work, page 11-31
- Understanding How Token Ring TrCRF VLANs Work, page 11-32
- Token Ring VLAN Configuration Guidelines, page 11-34
- Creating or Modifying a Token Ring TrBRF VLAN, page 11-34
- Creating or Modifying a Token Ring TrCRF VLAN, page 11-35

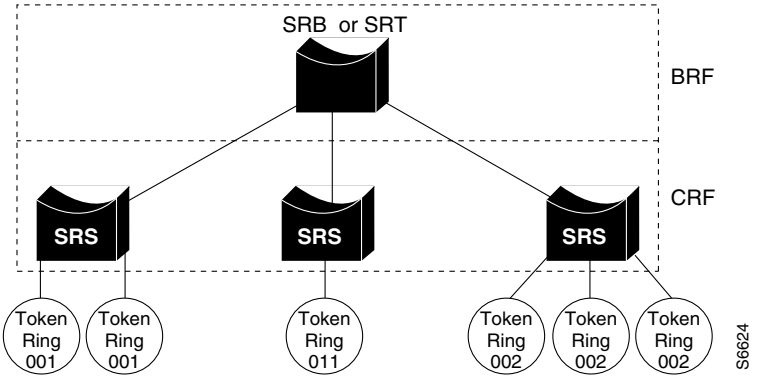
You must use VTP version 2 to configure and manage the Token Ring VLANs.

Note Catalyst 6500 series switches do not support the ISL-encapsulated Token Ring frames.

Understanding How Token Ring TrBRF VLANs Work

The Token Ring Bridge Relay Function (TrBRF) VLANs interconnect multiple the Token Ring Concentrator Relay Function (TrCRF) VLANs in a switched Token Ring network (see Figure 11-3). The TrBRF can be extended across a network of switches that are interconnected through the trunk links. The connection between the TrCRF and the TrBRF is referred to as a *logical port*.

Figure 11-3 Interconnected Token Ring TrBRF and TrCRF VLANs



For source routing, the switch appears as a single bridge between the logical rings. The TrBRF can function as a source-route bridge (SRB) or as a source-route transparent (SRT) bridge running either the IBM or IEEE STP. If SRB is used, you can define the duplicate MAC addresses on the different logical rings.

The Token Ring software runs an instance of STP for each TrBRF VLAN and each TrCRF VLAN. For the TrCRF VLANs, STP removes loops in the logical ring. For the TrBRF VLANs, STP interacts with the external bridges to remove the loops from the bridge topology, similar to STP operation on the Ethernet VLANs.

Caution Certain parent TrBRF STP and TrCRF bridge mode configurations can place the logical ports (the connection between the TrBRF and the TrCRF) of the TrBRF in a blocked state. For more information, see the "Default VLAN Configuration" section on page 11-3.

For source routing, the switch appears as a single bridge between the logical rings. The TrBRF can function as an SRB or SRT bridge running either the IBM or IEEE STP. If SRB is used, the duplicate MAC addresses can be defined on the different logical rings.

To accommodate the IBM System Network Architecture (SNA) traffic, you can use a combination of SRT and SRB modes. In a mixed mode, the TrBRF considers some ports (logical ports that are connected to TrCRFs) to operate in SRB mode while others operate in SRT mode.

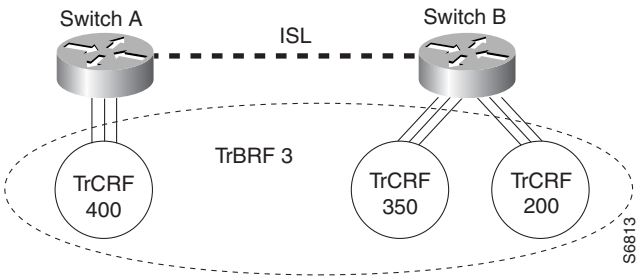
Understanding How Token Ring TrCRF VLANs Work

The TrCRF VLANs define the port groups with the same logical ring number. You can configure two TrCRF types in your network: undistributed and backup.

Typically, the TrCRFs are undistributed, which means that each TrCRF is limited to the ports on a single switch. Multiple undistributed TrCRFs on the same or separate switches can be associated with a single parent TrBRF (see Figure 11-4). The parent TrBRF acts as a multiport bridge, forwarding the traffic between the undistributed TrCRFs.

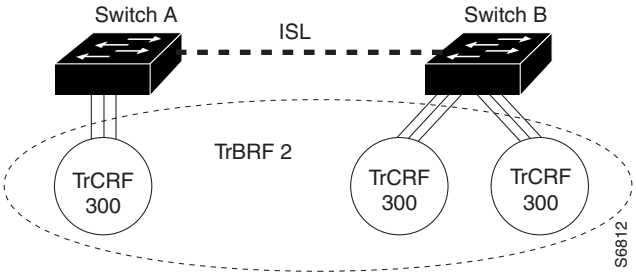
Note To pass data between the rings that are located on separate switches, you can associate the rings to the same TrBRF and configure the TrBRF for SRB.

Figure 11-4 Undistributed TrCRFs



Note By default, the Token Ring ports are associated with the default TrCRF (VLAN 1003, trcrf-default), which has the default TrBRF (VLAN 1005, trbrf-default) as its parent. In this configuration, a distributed TrCRF is possible (see Figure 11-5), and the traffic is passed between the default TrCRFs that are located on separate switches if the switches are connected through an ISL trunk.

Figure 11-5 Distributed TrCRF



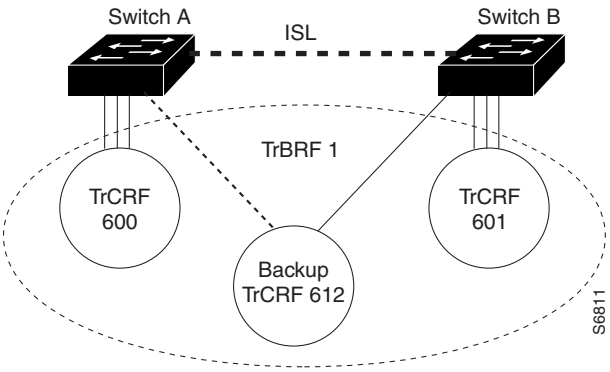
Within a TrCRF, source-route switching forwards the frames that are based on either the MAC addresses or the route descriptors. The entire VLAN can operate as a single ring, with frames that are switched between the ports within a single TrCRF.

You can specify the maximum hop count for the All-Routes and Spanning Tree Explorer frames for each TrCRF to limit the maximum number of hops that an explorer is allowed to traverse. If a port determines that the explorer frame it is receiving has traversed more than the number of specified hops, it does not forward the frame. The TrCRF determines the number of hops that an explorer has traversed based on the number of bridge hops in the route information field.

A backup TrCRF enables you to configure an alternate route for the traffic between the undistributed TrCRFs located on separate switches that are connected by a TrBRF if the ISL connection between the switches fails. Only one backup TrCRF for a TrBRF is allowed, and only one port per switch can belong to a backup TrCRF.

If the ISL connection between the switches fails, the port in the backup TrCRF on each affected switch automatically becomes active, rerouting the traffic between the undistributed TrCRFs through the backup TrCRF. When the ISL connection is reestablished, all but one port in the backup TrCRF is disabled. Figure 11-6 shows the backup TrCRF.

Figure 11-6 Backup TrCRF



Token Ring VLAN Configuration Guidelines

This section describes the guidelines for creating or modifying the Token Ring VLANs:

- For the Token Ring VLANs, the default TrBRF (VLAN 1005) can only be the parent of the default TrCRF (VLAN 1003). You cannot specify the default TrBRF as the parent of a user-configured TrCRF.
- You must configure a TrBRF before you configure the TrCRF; that is, the parent TrBRF VLAN you specify for the TrCRF must already exist.
- In a Token Ring environment, the logical ports of the TrBRF (the connection between the TrBRF and the TrCRF) are placed in a blocked state if either of these conditions exists:
  - The TrBRF is running the IBM STP, and the TrCRF is in SRT mode.
  - The TrBRF is running the IEEE STP, and the TrCRF is in SRB mode.

Creating or Modifying a Token Ring TrBRF VLAN

You must enable VTP version 2 before you create the Token Ring VLANs. For information on enabling VTP version 2, see Chapter 10, "Configuring VTP."

You must specify a bridge number when you create a new TrBRF.

To create a new Token Ring TrBRF VLAN, perform this task in privileged mode:

	Task	Command
Step 1	Create a new Token Ring TrBRF-type VLAN.	**set vlan** *vlan* [**name** *name*] **type trbrf** [**said** *said*] [**mtu** *mtu*] **bridge** *bridgeber* [**stp** {**ieee** | **ibm**}]
Step 2	Verify the VLAN configuration.	**show vlan** [*vlan*]

This example shows how to create a new Token Ring TrBRF VLAN and verify the configuration:

```
Console> (enable) set vlan 999 name TrBRF_999 type trbrf bridge a
Vlan 999 configuration successful
Console> (enable) show vlan 999
VLAN Name                             Status    IfIndex Mod/Ports, Vlans
---- -------------------------------- --------- ------- ------------------------
999  TrBRF_999                        active
VLAN Type  SAID       MTU   Parent RingNo BrdgNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ ------ ---- -------- ------ ------
999  trbrf 100999     4472  -      -      0xa    ibm  -        0      0
VLAN AREHops STEHops Backup CRF
---- ------- ------- ----------
Console> (enable)
```

To modify the VLAN parameters on an existing Token Ring TrBRF VLAN, perform this task in privileged mode:

	Task	Command
Step 1	Modify an existing Token Ring TrBRF-type VLAN.	**set vlan** *vlan* [**name** *name*] [**state** {**active** | **suspend**}] [**said** *said*] [**mtu** *mtu*] [**bridge** *bridgeber*] [**stp** {**ieee** | **ibm**}]
Step 2	Verify the VLAN configuration.	**show vlan** [*vlan*]

Creating or Modifying a Token Ring TrCRF VLAN

Note You must enable VTP version 2 before you create the Token Ring VLANs. For information on enabling VTP version 2, see Chapter 10, "Configuring VTP."

To create a new Token Ring TrCRF VLAN, perform this task in privileged mode:

	Task	Command
Step 1	Create a new Token Ring TrCRF-type VLAN.	**set vlan** *vlan* [**name** *name*] **type trcrf** [**said** *said*] [**mtu** *mtu*] {**ring** *hex_ringber* | **decring** *decimal_ringber*} **parent** *vlan*
Step 2	Verify the VLAN configuration.	**show vlan** [*vlan*]

Note You must specify a ring number (either in hexadecimal or in decimal) and a parent TrBRF VLAN when creating a new TrCRF.

This example shows how to create a Token Ring TrCRF VLAN and verify the configuration:

```
Console> (enable) set vlan 998 name TrCRF_998 type trcrf decring 10 parent 999
Vlan 998 configuration successful
Console> (enable) show vlan 998
VLAN Name                             Status    IfIndex Mod/Ports, Vlans
---- -------------------------------- --------- ------- -------------------------
998  TrCRF_998                        active    352
VLAN Type  SAID       MTU   Parent RingNo BrdgNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ ------ ---- -------- ------ ------
998  trcrf 100998     4472  999    0xa    -      -    srb      0      0
VLAN AREHops STEHops Backup CRF
---- ------- ------- ----------
998  7       7       off
Console> (enable)
```

To modify the VLAN parameters on an existing Token Ring TrCRF VLAN, perform this task in privileged mode:

	Task	Command
Step 1	Modify an existing Token Ring TrCRF VLAN.	**set vlan** *vlan* [**name** *name*] [**state** {**active** | **suspend**}] [**said** *said*] [**mtu** *mtu*] [**ring** *hex_ring*] [**decring** *decimal_ring*] [**bridge** *bridge*] [**parent** *vlan*]
Step 2	Verify the VLAN configuration.	**show vlan** [*vlan*]

To create a backup TrCRF, assign one port on each switch that the TrBRF traverses to the backup TrCRF.

To configure a TrCRF VLAN as a backup TrCRF, perform this task in privileged mode:

	Task	Command
Step 1	Configure a TrCRF VLAN as a backup TrCRF.	**set vlan** *vlan* **backupcrf on**
Step 2	Verify the VLAN configuration.	**show vlan** [*vlan*]

Caution If the backup TrCRF port is attached to a Token Ring multistation access unit (MSAU), it does not provide a backup path unless the ring speed and port mode are set by another device. We recommend that you configure the ring speed and port mode for the backup TrCRF.

To specify the maximum number of hops for the All-Routes Explorer frames or the Spanning Tree Explorer frames in the TrCRF, perform this task in privileged mode:

	Task	Command
Step 1	Specify the maximum number of hops for the All-Routes Explorer frames in the TrCRF.	**set vlan** *vlan* **aremaxhop** *hopcount*
Step 2	Specify the maximum number of hops for the Spanning Tree Explorer frames in the TrCRF.	**set vlan** *vlan* **stemaxhop** *hopcount*
Step 3	Verify the VLAN configuration.	**show vlan** [*vlan*]

This example shows how to limit the All-Routes Explorer frames and Spanning Tree Explorer frames to ten hops and how to verify the configuration:

```
Console> (enable) set vlan 998 aremaxhop 10 stemaxhop 10
Vlan 998 configuration successful
Console> (enable) show vlan 998
VLAN Name                             Status    IfIndex Mod/Ports, Vlans
---- -------------------------------- --------- ------- -------------------------

998  VLAN0998                         active    357


VLAN Type  SAID       MTU   Parent RingNo BrdgNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ ------ ---- -------- ------ ------
998  trcrf 100998     4472  999    0xff   -      -    srb      0      0


VLAN AREHops STEHops Backup CRF
---- ------- ------- ----------
998  10      10      off
Console> (enable)
```

Configuring VLANs for the Firewall Services Module

Enter the **set vlan** {*vlans*} **firewall-vlan** {*mod*} command to specify the VLANs that are secured by a Firewall Services Module (WS-SVC-FWM-1-K9). Enter the **show vlan firewall-vlan** *mod* command to display the VLANs that are secured by the Firewall Services Module.

To secure a range of VLANs on a Firewall Services Module, these conditions must be satisfied:

Note VLAN 1 cannot be secured to the Firewall Services Module.

1. The port membership must be defined for the VLANs, and the VLANs must be in the active state.
2. The VLANs cannot have a Layer 3 interface in the active state on the MSFC.
3. The VLANs cannot be reserved VLANs.

The VLANs that do not satisfy condition number 2 in the list above are discarded from the range of VLANs that you attempt to secure on the Firewall Services Module.

The VLANs that meet condition number 2 and condition number 3 but do not meet condition number 1 are stored in the supervisor engine database; these VLANs are sent to the Firewall Services Module as soon as they meet condition number 1.

This example shows how to secure a range of VLANs on a Firewall Services Module:

```
Console> (enable) set vlan 2-55 firewall-vlan 7
Console> (enable)
```

Enter the **set firewall multiple-vlan-interfaces** {**enable** | **disable**} command to set the multiple VLAN interface feature for a Firewall Services Module. Disabling the multiple VLAN interface feature sets the Firewall Services Module to single VLAN interface mode. The multiple VLAN interface feature is disabled by default. An example is as follows:

```
Console> (enable) set firewall multiple-vlan-interfaces  enable
This command will enable multiple-vlan-interfaces feature for all firewall
modules in the chassis.
It can result in traffic bypassing the firewall module.
Do you want to continue (y/n) [n]? y
multiple-vlan-interfaces feature enabled for firewall module 5.
Console> (enable)
```

With software release 8.4(1) and later releases, you can enter the **set vlan** {*vlan*} **firewall-vlan** {*mod*} **msfc-fwsm-interface** command to make the specified VLAN the secured interface between the MSFC and the Firewall Services Module. This command is available only in the single VLAN interface mode and cannot be entered when multiple VLAN interfaces are enabled. An example is as follows:

```
Console> (enable) set vlan 3 firewall-vlan 5 msfc-fwsm-interface
Vlan 3 declared as Secure Vlan interface for module 5
Vlan 3 declared secure for firewall module 5
Console> (enable)
```

Note For detailed Firewall Services Module configuration information, refer to the Firewall Services Module documentation at this URL:

http://www.cisco.com/en/US/products/hw/modules/ps2706/ps4452/tsd_products_support_model_home.html

CHAPTER 12

Configuring InterVLAN Routing

This chapter describes how to configure the Multilayer Switch Feature Card (MSFC) for interVLAN routing on the Catalyst 6500 series switches.

Note For complete syntax and usage for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Note Refer to the *FlexWAN Module Port Adapter Installation and Configuration Notes* for information about configuring routing on FlexWAN module interfaces.

Understanding How InterVLAN Routing Works

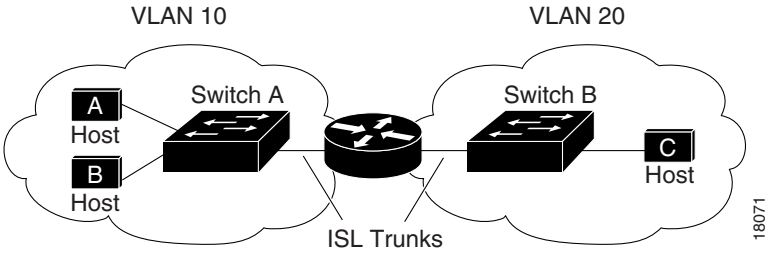
The network devices in different VLANs cannot communicate with one another without a router to forward traffic between the VLANs. In most network environments, the VLANs are associated with individual networks or subnetworks.

For example, in an IP network, each subnetwork is mapped to an individual VLAN. In an IPX network, each VLAN is mapped to an IPX network number.

Configuring the VLANs helps to control the size of the broadcast domain and keeps the local traffic local. When an end station in one VLAN needs to communicate with an end station in another VLAN, interVLAN communication is required. This communication is provided by interVLAN routing. You configure one or more routers to route traffic to the appropriate destination VLAN.

Figure 12-1 shows a basic interVLAN routing topology. Switch A is in VLAN 10 and Switch B is in VLAN 20. The router has an interface in each VLAN.

Figure 12-1 Basic InterVLAN Routing Topology



When Host A in VLAN 10 needs to communicate with Host B in VLAN 10, it sends a packet that is addressed to that host. Switch A forwards the packet directly to Host B without sending it to the router.

When Host A sends a packet to Host C in VLAN 20, Switch A forwards the packet to the router, which receives the traffic on the VLAN 10 interface. The router checks the routing table, determines the correct outgoing interface, and forwards the packet out the VLAN 20 interface to Switch B. Switch B receives the packet and forwards it to Host C.

Configuring InterVLAN Routing on the MSFC

Note This section is for users who are familiar with Cisco IOS software and have some experience configuring Cisco IOS routing. If you are not familiar with configuring Cisco routing, refer to the Cisco IOS documentation on Cisco.com.

These sections describe how to configure interVLAN routing on the MSFC:

MSFC Routing Configuration Guidelines

This section describes the guidelines (which consists of two main procedures) for configuring interVLAN routing on the MSFC:

1. Create and configure VLANs on the switch and assign VLAN membership to switch ports. For more information, see Chapter 11, "Configuring VLANs."
2. Create and configure VLAN interfaces for interVLAN routing on the MSFC. Configure a VLAN interface for each VLAN for which you want to route traffic.

The VLAN interfaces on the MSFC are virtual interfaces. However, you configure them the same as you do a physical router interface.

The MSFC3, MSFC2, MSFC2A, and MSFC support the same range of VLANs as the supervisor engine. MSFC3, MSFC2, and MSFC2A support up to 1,000 VLAN interfaces, and the MSFC supports up to 256 VLAN interfaces.

Configuring IP InterVLAN Routing on the MSFC

To configure interVLAN routing for IP, perform this task:

	Task	Command
Step 1	(Optional) Enable IP routing on the router[1].	Router(config)# **ip routing**
Step 2	(Optional) Specify an IP routing protocol[2].	Router(config)# **router** *ip_routing_protocol*
Step 3	Specify a VLAN interface on the MSFC.	Router(config)# **interface** *vlan-id*
Step 4	Assign an IP address to the VLAN.	Router(config-if)# **ip address** *n.n.n.n mask*
Step 5	Exit configuration mode.	Router(config-if)# **Ctrl-Z**

1. This step is necessary if you have multiple routers in the network.
2. This step is necessary if you enabled IP routing in Step 1. This step might include other commands, such as using the **network** router configuration command to specify the networks to route. Refer to the documentation for your router platform for detailed information on configuring routing protocols.

This example shows how to enable IP routing on the MSFC, create a VLAN interface, and assign the IP address to the interface:

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# ip routing
Router(config)# router rip
Router(config-router)# network 10.0.0.0
Router(config-router)# interface vlan 100
Router(config-if)# ip address 10.1.1.1 255.0.0.0
Router(config-if)# ^Z
Router#
```

Configuring IPX InterVLAN Routing on the MSFC

Note With Supervisor Engine 720 (MSFC3), IPX routing is done through the software.

To configure interVLAN routing for Internetwork Packet Exchange (IPX), perform this task:

	Task	Command
Step 1	(Optional) Enable IPX routing on the router[1].	Router(config)# **ipx routing**
Step 2	(Optional) Specify an IPX routing protocol[2].	Router(config)# **ipx router** *ipx_routing_protocol*
Step 3	Specify a VLAN interface on the MSFC.	Router(config)# **interface** *vlan-id*
Step 4	Assign a network number to the VLAN[3].	Router(config-if)# **ipx network** [*network* | **unnumbered**] **encapsulation** *encapsulation-type*
Step 5	Exit configuration mode.	Router(config-if)# **Ctrl-Z**

1. This step is necessary if you have multiple routers in the network.
2. This step is necessary if you enabled IPX routing in Step 1. This step might include other commands, such as using the **network** router configuration command to specify the networks to route. Refer to the documentation for your router platform for detailed information on configuring routing protocols.
3. This step enables IPX routing on the VLAN. When you enable IPX routing on the VLAN, you can also specify an encapsulation type.

This example shows how to enable IPX routing on the MSFC, create a VLAN interface, and assign an IPX network address to the interface:

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# ipx routing
Router(config)# ipx router rip
Router(config-ipx-router)# network all
Router(config-ipx-router)# interface vlan100
Router(config-if)# ipx network 100 encapsulation snap
Router(config-if)# ^Z
Router#
```

Configuring AppleTalk InterVLAN Routing on the MSFC

To configure interVLAN routing for AppleTalk, perform this task:

	Task	Command
Step 1	(Optional) Enable AppleTalk routing on the router[1].	Router(config)# **appletalk routing**
Step 2	Specify a VLAN interface on the MSFC.	Router(config)# **interface** *vlan-id*
Step 3	Assign a cable range to the VLAN.	Router(config-if)# **appletalk cable-range** *cable-range*
Step 4	Assign a zone name to the VLAN.	Router(config-if)# **appletalk zone** *zone-name*
Step 5	Exit configuration mode.	Router(config-if)# **Ctrl-Z**

1. This step is necessary if you have multiple routers in the network.

This example shows how to enable AppleTalk routing on the MSFC, create a VLAN interface, and assign an AppleTalk cable range and zone name to the interface:

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# appletalk routing
Router(config)# interface vlan100
Router(config-if)# appletalk cable-range 100-100
Router(config-if)# appletalk zone Engineering
Router(config-if)# ^Z
Router#
```

Configuring MSFC Features

These sections describe the MSFC features:

Local Proxy ARP

With Release 12.1(2)E or later releases, the Local Proxy Address Resolution Protocol (ARP) allows the MSFC to respond to the ARP requests for the IP addresses within a subnet where normally no routing is required. With local proxy ARP enabled, the MSFC responds to all the ARP requests for the IP addresses within the subnet and forwards all traffic between the hosts in the subnet. Use this feature only on the subnets where the hosts are intentionally prevented from communicating directly by the configuration on the switch to which they are connected.

Local proxy ARP is disabled by default. Enter the **ip local-proxy-arp** interface configuration command to enable local proxy ARP on an interface. Enter the **no ip local-proxy-arp** interface configuration command to disable the feature. The Internet Control Message Protocol (ICMP) redirects are disabled on the interfaces where local proxy ARP is enabled.

WCCP Layer 2 Redirection

Note Supervisor Engine 1 with the Policy Feature Card (PFC) supports this feature with Release 12.1(2)E or later releases. Supervisor Engine 2 with PFC2 supports this feature with Release 12.1(3a)E or later releases. WCCP Layer 2 redirection is not supported on Supervisor Engine 720 or Supervisor Engine 32.

Web Cache Communication Protocol (WCCP) Layer 2 redirection allows directly connected Cisco Cache Engines to use Layer 2 redirection, which is more efficient than Layer 3 redirection, through generic routing encapsulation (GRE). You can configure a directly connected Cache Engine to negotiate WCCP Layer 2 redirection. WCCP Layer 2 redirection requires no configuration on the MSFC. Enter the **show ip wccp web-cache detail** command to display which redirection method is in use for each cache. Follow these guidelines when using this feature:

- WCCP Layer 2 redirection sets the IP flow mask to full-flow mode.
- You can configure the Cisco Cache Engine software release 2.2 or later releases to use WCCP Layer 2 redirection.

- Layer 2 redirection takes place on the switch and is not visible to the MSFC. Entering the **show ip wccp web-cache detail** command on the MSFC displays statistics for only the first packet of a Layer 2 redirected flow, which provides an indication of how many flows, rather than packets, are using Layer 2 redirection. Entering the **show mls entries** command on the supervisor engine displays the other packets in the Layer 2 redirected flows.

Configure the Cisco IOS WCCP as described in the *Cisco IOS Configuration Fundamentals Configuration Guide* at this URL:
http://www.cisco.com/en/US/docs/ios/12_1/configfun/configuration/guide/fcd305.html

Autostate Feature

These MSFC autostate port-based modes are supported:

- Normal Autostate Mode, page 12-6
- Autostate Exclude Mode, page 12-6
- Autostate Track Mode, page 12-7

Normal Autostate Mode

Autostate shuts down (or brings up) the Layer 3 interfaces/subinterfaces on the MSFC and the Multilayer Switch Module (MSM) when the following port configuration changes occur on the switch:

- When the last port on a VLAN goes down, all the Layer 3 interfaces/subinterfaces on that VLAN shut down (are autostated) unless sc0 is on the VLAN or another router is in the chassis with an interface/subinterface in the VLAN.
- When the first port on the VLAN is brought back up, all the Layer 3 interfaces on that VLAN that were previously shut down are brought up.

The Catalyst 6500 series switch does not have knowledge of, or control over, the MSM or MSFC configuration (just as the switch does not have knowledge of, or control over, external router configurations). Autostate does not work on MSM or MSFC interfaces if the MSM or MSFC is not properly configured. For example, consider this MSM trunk configuration:

```
interface GigabitEthernet0/0/0.200
    encap isl 200
    .
    .
```

In the example, the GigabitEthernet0/0/0.200 interface is not autostated if any of these configuration errors are made:

- VLAN 200 is not configured on the switch.
- Trunking is not configured on the corresponding Gigabit Ethernet switch port.
- Trunking is configured but VLAN 200 is not an allowed VLAN on that trunk.

Autostate Exclude Mode

Autostate exclude mode allows you to specify the ports to exclude from autostate. In normal autostate mode, the Layer 3 interfaces remain up if at least one port in the VLAN remains up. If you have appliances, such as load balancers or firewall servers that are connected to the ports in the VLAN, you can configure these ports to be excluded from the autostate feature to make sure that the forwarding SVI does not go down if these ports become inactive.

Autostate exclude mode affects all VLANs to which the port belongs and is supported on Ethernet, Fast Ethernet, and Gigabit Ethernet ports only.

Note You cannot configure both autostate exclude mode and autostate track mode on the same port.

Autostate Track Mode

You can use autostate track mode to track key VLAN or port connections to the MSFC. When you configure the autostate track mode, the SVI stays up if any tracked connections remain up in the VLAN. Track mode requires that you define a global tracked VLAN group. The VLANs in this group will be tracked by MSFC autostate whether or not you define a member port to be tracked.

When you configure a VLAN and the ports to be tracked by autostate, the tracked SVIs remain down until at least one tracked Ethernet port in the VLAN moves to the Spanning Tree Protocol (STP) forwarding state. Conversely, tracked SVIs remain up if at least one tracked Ethernet port stays in the STP forwarding state.

Autostate track mode is supported on Ethernet, Fast Ethernet, and Gigabit Ethernet ports only.

Note You cannot configure both autostate exclude mode and autostate track mode on the same port.

Configuring Autostate Exclude Mode

To configure autostate exclude mode, perform one of these tasks in privileged mode:

Task	Command
Configure autostate exclude mode.	**set msfcautostate exclude** *mod/port*
Clear the autostate configuration.	**clear msfcautostate** {**all** | *mod/port*}

This example shows how to exclude a port from MSFC autostate:

```
Console> (enable) set msfcautostate exclude 3/1
Port 3/1 configured as excluded port
Console> (enable)
```

This example shows how to clear the autostate configuration:

```
Console> (enable) clear msfcautostate 3/1
MSFC autostate config cleared on excluded port 3/1
Console> (enable)
```

Configuring Autostate Track Mode

To configure autostate track mode, perform one of these tasks in privileged mode:

Task	Command
Configure autostate to track the specified VLANs.	**set msfcautostate track [disable | enable** *vlan_list*]
Configure autostate to track the specified ports.	**set msfcautostate track** *mod/port_list*
Clear the autostate track mode configuration.	**clear msfcautostate all |** *mod/port*

This example shows how to configure autostate to track VLANs 20, 21, 22, and 28:

```
Console> (enable) set msfcautostate track enable 20-22,28
Vlans 20-22,28 added to MSFC autostate track vlan group
Console> (enable)
```

This example shows how to configure autostate to track ports 1–5 on module 3:

```
Console> (enable) set msfcautostate track 3/1-5
Port 3/1-5 configured as tracked port
Console> (enable)
```

Displaying the Autostate Configuration

To display the current line protocol state determination for the MSM, perform this task in normal mode:

Task	Command
Display the current line protocol state determination for the MSM.	**show msmautostate** *mod*

This example shows how to display the current line protocol state determination for the MSM:

```
Console> show msmautostate
MSM Auto port state: enabled
Console>
```

To display the line protocol state determination for the MSFC, perform this task in privileged mode:

Task	Command
Display the line protocol state determination for the MSFC.	**show msfcautostate**

This example shows how to display the line protocol state determination for the MSFC:

```
Console> (enable) show msfcautostate
MSFC Auto port state: enabled
Excluded ports:
Tracked ports: 3/1-5
Tracked vlans: 20-22,28
Console> (enable)
```

To check which MSM interfaces are currently autostated, perform this task in enabled mode from the MSM prompt:

Task	Command
Check which MSM interfaces are currently autostated.	**show autostate entries**

This example shows how to check which MSM interfaces are currently autostated (shut down or brought up through autostate):

```
Router# show autostate entries
Port-channel1.5
Port-channel1.6
Port-channel1.4
Router#
```

Disabling Autostate

To disable autostate if you have an MSM installed, perform this task in privileged mode:

Task	Command
Disable autostate if you have an MSM installed.	**set msmautostate disable**

Autostate is enabled by default. This example shows how to disable autostate if you have an MSM installed:

```
Console> (enable) set msmautostate disable
MSM port auto state disabled.
Console> (enable)
```

To disable the line protocol state determination of the MSFC, perform this task in privileged mode:

Note If you toggle (enable to disable and/or disable to enable) the **msfcautostate** command, you might have to use the **shutdown** and **no shutdown** commands to disable and then restart the VLAN and WAN interfaces on the MSFC to bring them back up. Unless there is a valid reason, the MSFC autostate feature should not be disabled.

Task	Command
Disable the line protocol state determination of the MSFC.	**set msfcautostate disable**

This example shows how to disable the line protocol state determination of the MSFC:

```
Console> (enable) set msfcautostate disable

MSM port auto state disabled.
Console> (enable)
```

CHAPTER 13

Configuring CEF for PFC2 and PFC3A

This chapter describes how to configure Cisco Express Forwarding (CEF) for Policy Feature Card 2 (PFC2) and PFC3A on the Catalyst 6500 series switches.

CEF for PFC2 provides IP and Internetwork Packet Exchange (IPX) unicast Layer 3 switching and IP multicast Layer 3 switching for Supervisor Engine 2, PFC2, and Multilayer Switch Feature Card 2 (MSFC2).

CEF for PFC3A provides IP unicast Layer 3 switching and IP multicast Layer 3 switching for Supervisor Engine 720, PFC3A, and Multilayer Switch Feature Card 3 (MSFC3).

Note With Supervisor Engine 720 (MSFC3), IPX routing is done through the software.

Note For complete information on the syntax and usage information for the supervisor engine commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Note Supervisor Engine 1 with PFC1 and MSFC or MSFC2 provide Layer 3 switching with Multilayer Switching (MLS). See Chapter 14, "Configuring MLS," for more information.

Note To configure MSFC2 to support MLS on a Catalyst 5000 family switch, refer to the *Layer 3 Switching Software Configuration Guide* at this URL: http://www.cisco.com/en/US/products/hw/switches/ps5304/products_configuration_guide_book09186a00800d67ec.html.

Understanding How Layer 3 Switching Works

These sections describe Layer 3 switching with PFC2:

Layer 3 Switching Overview

Note With Supervisor Engine 720 (MSFC3), IPX routing is done through the software.

Layer 3 switching allows a switch, instead of a router, to forward the IP and IPX unicast traffic and the IP multicast traffic between the VLANs. Layer 3 switching is implemented in the hardware and provides wire-speed interVLAN forwarding on the switch, rather than on MSFC2/MSFC3. Layer 3 switching requires minimal support from MSFC2/MSFC3. MSFC2/MSFC3 routes any traffic that cannot be Layer 3 switched.

Note Layer 3 switching supports the routing protocols that are configured on MSFC2/MSFC3. Layer 3 switching does not replace the routing protocols that are configured on MSFC2/MSFC3. Layer 3 switching uses Protocol Independent Multicast (PIM) for multicast route determination.

Layer 3 switching on Catalyst 6500 series switches provides flow statistics that you can use to identify the traffic characteristics for administration, planning, and troubleshooting. Layer 3 switching uses NetFlow Data Export (NDE) to export the flow statistics. See Chapter 16, "Configuring NDE" for more information about NDE.

Note Traffic is Layer 3 switched after being processed by the VLAN access control list (VACL) feature and the quality of service (QoS) feature.

Understanding Layer 3-Switched Packet Rewrite

Note With Supervisor Engine 720 (MSFC3), IPX routing is done through the software.

When a packet is Layer 3 switched from a source in one VLAN to a destination in another VLAN, the switch performs a packet rewrite at the egress port that is based on information learned from MSFC2/MSFC3 so that the packets appear to have been routed by MSFC2/MSFC3.

Note Rather than just forwarding IP multicast packets, the PFC2/PFC3A replicates them as necessary on the appropriate VLANs.

The packet rewrite alters these five fields:

- Layer 2 (MAC) destination address
- Layer 2 (MAC) source address
- Layer 3 IP Time to Live (TTL) or IPX Transport Control
- Layer 3 checksum
- Layer 2 (MAC) checksum (also called the frame checksum or FCS)

Note Packets are rewritten with the encapsulation that is appropriate for the next-hop subnet.

If Source A and Destination B are on different VLANs and Source A sends a packet to MSFC2/MSFC3 to be routed to Destination B, the switch recognizes that the packet was sent to the Layer 2 (MAC) address of MSFC2/MSFC3.

To perform Layer 3 switching, the switch rewrites the Layer 2 frame header, changing the Layer 2 destination address to the Layer 2 address of Destination B and the Layer 2 source address to the Layer 2 address of MSFC2/MSFC3. The Layer 3 addresses remain the same.

In IP unicast and IP multicast traffic, the switch decrements the Layer 3 TTL value by 1 and recomputes the Layer 3 packet checksum. In IPX traffic, the switch increments the Layer 3 Transport Control value by 1 and recomputes the Layer 3 packet checksum. The switch recomputes the Layer 2 frame checksum and forwards (or for multicast packets, replicates as necessary) the rewritten packet to Destination B's VLAN.

These sections describe how the packets are rewritten:

- Understanding IP Unicast Rewrite, page 13-3
- Understanding IPX Unicast Rewrite, page 13-4
- Understanding IP Multicast Rewrite, page 13-4

Understanding IP Unicast Rewrite

Received IP unicast packets are (conceptually) formatted as follows:

Layer 2 Frame Header		**Layer 3 IP Header**				**Data**	**FCS**
Destination	Source	Destination	Source	TTL	Checksum		
MSFC2/MSFC3 MAC	*Source A MAC*	*Destination B IP*	*Source A IP*	*n*	*calculation1*		

After the switch rewrites an IP unicast packet, it is (conceptually) formatted as follows:

Layer 2 Frame Header		**Layer 3 IP Header**				**Data**	**FCS**
Destination	Source	Destination	Source	TTL	Checksum		
Destination B MAC	*MSFC2/MSFC3 MAC*	*Destination B IP*	*Source A IP*	*n-1*	*calculation2*		

Understanding IPX Unicast Rewrite

Received IPX packets are (conceptually) formatted as follows:

Layer 2 Frame Header		Layer 3 IPX Header			Data	FCS
Destination	Source	Checksum/ IPX Length/ Transport Control	Destination Net/ Node/ Socket	Source Net/ Node/ Socket		
MSFC2 MAC	*Source A MAC*	*n*	*Destination B IPX*	*Source A IPX*		

After the switch rewrites an IPX packet, it is (conceptually) formatted as follows:

Layer 2 Frame Header		Layer 3 IPX Header			Data	FCS
Destination	Source	Checksum/ IPX Length/ Transport Control	Destination Net/ Node/ Socket	Source Net/ Node/ Socket		
Destination B MAC	*MSFC2 MAC*	*n+1*	*Destination B IPX*	*Source A IPX*		

Understanding IP Multicast Rewrite

Received IP multicast packets are (conceptually) formatted as follows:

Layer 2 Frame Header		Layer 3 IP Header				Data	FCS
Destination	Source	Destination	Source	TTL	Checksum		
Group G1 MAC[1]	*Source A MAC*	*Group G1 IP*	*Source A IP*	*n*	*calculation1*		

1. In this example, Destination B is a member of Group G1.

After the switch rewrites an IP multicast packet, it is (conceptually) formatted as follows:

Frame Header		IP Header				Data	FCS
Destination	Source	Destination	Source	TTL	Checksum		
Group G1 MAC	*MSFC2/MSFC3 MAC*	*Group G1 IP*	*Source A IP*	*n–1*	*calculation2*		

Understanding CEF for PFC2/PFC3A

Note With Supervisor Engine 720 (MSFC3), IPX routing is done through the software.

These sections describe CEF for PFC2:

CEF for PFC2/PFC3A Overview

Supervisor Engine 2, PFC2, and MSFC2 provide Layer 3 switching with CEF for PFC2. CEF for PFC2 is permanently enabled on Supervisor Engine 2. Cisco IOS CEF is permanently enabled on MSFC2 in support of CEF for PFC2.

Supervisor Engine 720, PFC3A, and MSFC3 provide Layer 3 switching with CEF for PFC3A. CEF for PFC3A is permanently enabled on Supervisor Engine 720. Cisco IOS CEF is permanently enabled on MSFC3 in support of CEF for PFC3A.

CEF for PFC2/PFC3A works with CEF (for unicast traffic) and PIM (for multicast traffic) on MSFC2/MSFC3 to support IP, IP multicast, and IPX traffic. CEF and PIM on MSFC2/MSFC3 are enhanced to support CEF for PFC2/PFC3A. CEF for PFC2/PFC3A generates flow statistics for Layer 3-switched traffic that can be displayed at the CLI or used for NDE.

CEF for PFC2/PFC3A provides Layer 3 switching for all packets that match a complete forwarding information base (FIB) entry (see the "Understanding the FIB" section on page 13-6). CEF for PFC2/PFC3A sends all packets that match an incomplete FIB entry (one where the MAC address has not been resolved) to MSFC2/MSFC3 to be routed until MSFC2/MSFC3 resolves the MAC address.

Note CEF for PFC2/PFC3A sends bridge traffic that is addressed at Layer 2 to MSFC2/MSFC3 to be processed.

Note Access control lists (ACLs) and policy-based routing can cause CEF for PFC2/PFC3A to ignore the FIB when making a forwarding decision (see the "Understanding the Forwarding Decisions" section on page 13-6).

Understanding the Forwarding Decisions

CEF for PFC2/PFC3A provides Layer 3 switching that is based on the following:

- Entries in the ACL ternary content addressable memory (TCAM) for policy-based routing decisions
- Entries in the NetFlow table for TCP intercept and reflexive ACL forwarding decisions (see the "Understanding the NetFlow Statistics" section on page 13-11)
- Entries in the FIB and adjacency table for all other forwarding decisions

Enter the **show mls entry** command to display information about the entries that are used to make forwarding decisions. CEF for PFC2/PFC3A makes a forwarding decision for each packet and sends the rewrite information for each packet to the egress port, where the rewrite occurs when the packet is transmitted from the switch.

Understanding the FIB

The FIB resides in a separate TCAM. The adjacency table is stored separately in DRAM. The NetFlow table is stored separately in DRAM. The FIB, the adjacency table, and the NetFlow table do not compete with any other features for storage space.

The FIB is conceptually similar to a routing table. It maintains a mirror image of the forwarding information that is contained in the unicast and multicast routing tables on MSFC2/MSFC3. When routing or topology changes occur in the network, the unicast and multicast routing tables on MSFC2/MSFC3 are updated and those changes are reflected in the FIB. The FIB maintains next-hop address information that is based on the information in the routing tables on MSFC2/MSFC3. The FIB supports 256,000 entries, which includes 16,000 IP multicast entries (128,000 IP multicast entries on MSFC3). With reverse path forwarding (RPF) check enabled, the number of IP entries doubles (with Supervisor Engine 720, the number of IP entries remain the same).

FIB lookup uses the following criteria:

- Destination IP address for IP unicast
- Destination IPX network for IPX unicast
- Source and destination IP address for IP unicast with RPF check
- Source and destination IP address for IP multicast with RPF check

Note Because the FIB mirrors the unicast and multicast routing tables on MSFC2/MSFC3, any commands on MSFC2/MSFC3 that change the unicast or multicast routing tables affect the FIB. Forwarding entries cannot be cleared from the Supervisor Engine 2 or Supervisor Engine 720 command-line interface (CLI).

In switches with redundant supervisor engines and MSFC2s/MSFC3s, the designated MSFC2/MSFC3 supports the FIB on the active Supervisor Engine 2 or Supervisor Engine 720. The routing protocols on the nondesignated MSFC2/MSFC3 send information to the routing protocols on the designated MSFC2/MSFC3.

Enter the **show mls entry cef** command to display the following:

- Module number of the MSFC that is supporting the FIB
- FIB entry type (receive, connected, resolved, drop, wildcard, or default)
- Destination address (IP address or IPX network)
- Destination mask
- Next-hop address (IP address or IPX network)

- Next-hop mask
- Next-hop load-sharing weight

```
Console> (enable) show mls entry cef
Mod FIB-Type  Destination-IP  Destination-Mask NextHop-IP      Weight
--- --------- --------------- ---------------- --------------- ------
 15 receive   0.0.0.0         255.255.255.255
 15 receive   255.255.255.255 255.255.255.255
 15 receive   127.0.0.0       255.255.255.255
 15 receive   127.0.0.52      255.255.255.255
 15 receive   127.255.255.255 255.255.255.255
 15 receive   10.1.1.2        255.255.255.255
 15 receive   10.1.1.0        255.255.255.255
 15 receive   10.1.1.255      255.255.255.255
 15 receive   10.10.1.1       255.255.255.255
 15 receive   10.10.0.0       255.255.255.255
.
.
.
Console> (enable)
```

Enter the **show mls** command to display a Layer 3 switching summary:

```
Console> (enable) show mls
Total packets switched = 35254
Total bytes switched = 2256256
Total routes = 120569
Total number of Netflow entries = 120000

IP statistics flows aging time = 50 seconds
Long-duration flows aging time = 320 seconds
IP statistics flows fast aging time = 0 seconds, packet threshold = 0

IP Current flow mask is Full-Vlan flow
Netflow Data Export version: 7
Netflow Data Export disabled
Netflow Data Export port/host is not configured.
Total packets exported = 0
Destination Ifindex export is enabled
Source Ifindex export is enabled
Rate limiting is turned off, packets are bridged to router
Load balancing hash is based on source and destination IP addresses and universc
Per-prefix Stats for ALL FIB entries is Enabled
Console> (enable)
```

Understanding the Adjacency Table

For each FIB entry, CEF for PFC2/PFC3A stores Layer 2 information from the designated MSFC2/MSFC3 for adjacent nodes in the adjacency table. Adjacent nodes are nodes that are directly connected at Layer 2. To forward traffic, CEF for PFC2/PFC3A selects a route from a FIB entry, which points to an adjacency entry, and uses the Layer 2 header for the adjacent node in the adjacency table entry to rewrite the packet during Layer 3 switching. CEF for PFC2 supports 256,000 adjacency table entries. CEF for PFC3A supports 1,000,000 adjacency table entries. Only half of the adjacency table entries provide statistics.

Table 13-1 lists the adjacency types.

Table 13-1 Adjacency Types

Adjacency Type	Description
connect	Entry type that contains complete rewrite information
punt	Entry to send traffic to MSFC2/MSFC3
no r/w	Entry to send traffic to MSFC2/MSFC3 when rewrite information is incomplete
frc drp	Entry that is used to drop packets due to ARP throttling
drop, null, loopbk	Entries that are used to drop packets

Enter the **show mls entry cef adjacency** command to display the following:

- FIB information (see the "Understanding the FIB" section on page 13-6)
- Adjacency type (connect, drop, null, loopbk, frc drp, punt, no r/w)
- Next-hop MAC address
- Next-hop VLAN
- Next-hop encapsulation
- Number of packets that are transmitted to this adjacency from the associated FIB entry
- Number of bytes that are transmitted to this adjacency from the associated FIB entry

```
Console> (enable) show mls entry cef adjacency
Mod:                15
Destination-IP:     140.140.1.5       Destination-Mask:   255.255.255.255
FIB-Type:           resolved

AdjType  NextHop-IP      NextHop-Mac       Vlan Encp Tx-Packets   Tx-Octets
-------- --------------- ----------------- ---- ---- ------------ -------------
connect  140.140.1.5     00-00-d0-00-00-05  140 ARPA            0             0

Mod:                15
Destination-IP:     150.150.1.5       Destination-Mask:   255.255.255.255
FIB-Type:           resolved

AdjType  NextHop-IP      NextHop-Mac       Vlan Encp Tx-Packets   Tx-Octets
-------- --------------- ----------------- ---- ---- ------------ -------------
connect  150.150.1.5     00-00-e0-00-00-05  150 ARPA            0             0

Mod:                15
Destination-IP:     153.153.1.5       Destination-Mask:   255.255.255.255
FIB-Type:           resolved

AdjType  NextHop-IP      NextHop-Mac       Vlan Encp Tx-Packets   Tx-Octets
-------- --------------- ----------------- ---- ---- ------------ -------------
connect  153.153.1.5     00-00-e3-00-00-05  153 ARPA            0             0
.
.
.
Console> (enable)
```

Enter the **clear mls entry cef adjacency** command to clear the CEF adjacency information:

```
Console> (enable) clear mls entry cef adjacency
Adjacency statistics has been cleared.
Console> (enable)
```

Partially and Completely Switched Multicast Flows

Some flows might be partially Layer 3 switched instead of completely Layer 3 switched in these situations:

- MSFC2/MSFC3 is configured as a member of the IP multicast group (using the **ip igmp join-group** command) on the RPF interface of the multicast source.
- MSFC2/MSFC3 is the first-hop router to the source in PIM sparse mode (in this case, MSFC2/MSFC3 must send PIM-register messages to the rendezvous point).
- The multicast TTL threshold is configured on an egress interface for the flow.
- The multicast helper is configured on the RPF interface for the flow, and multicast to broadcast translation is required.
- Multicast tag switching is configured on an egress interface.
- Network Address Translation (NAT) is configured on an interface, and source address translation is required for the outgoing interface.

Note CEF for PFC2/PFC3A provides Layer 3 switching when the extended access list deny condition on the RPF interface specifies something other than the Layer 3 source, Layer 3 destination, or IP protocol (an example is the Layer 4 port numbers).

For partially switched flows, all multicast traffic belonging to the flow reaches MSFC2/MSFC3 and is software switched for any interface that is not Layer 3 switched.

Note All (*,G) flows are always partially Layer 3 switched.

PFC2/PFC3A prevents multicast traffic in flows that are completely Layer 3 switched from reaching MSFC2/MSFC3, reducing the load on MSFC2/MSFC3. The **show ip mroute** and **show mls ip multicast** commands identify completely Layer 3-switched flows with the text string “RPF-MFD.” Multicast Fast Drop (MFD) indicates that from the perspective of MSFC2/MSFC3, the multicast packet is dropped, because it is switched by the PFC2/PFC3A.

For all completely Layer 3-switched flows, PFC2/PFC3A periodically sends multicast packet and byte count statistics to MSFC2/MSFC3, because MSFC2/MSFC3 cannot record multicast statistics for completely switched flows, which it never sees. MSFC2/MSFC3 uses the statistics to update the corresponding multicast routing table entries and reset the appropriate expiration timers.

CEF for PFC2/PFC3A Examples

Figure 13-1 shows a simple IP CEF network topology. In this example, Host A is on the Sales VLAN (IP subnet 171.59.1.0), Host B is on the Marketing VLAN (IP subnet 171.59.3.0), and Host C is on the Engineering VLAN (IP subnet 171.59.2.0).

When Host A initiates an HTTP file transfer to Host C, PFC2/PFC3A uses the information in the FIB and adjacency table to forward packets from Host A to Host C.

Figure 13-1 IP CEF Example Topology

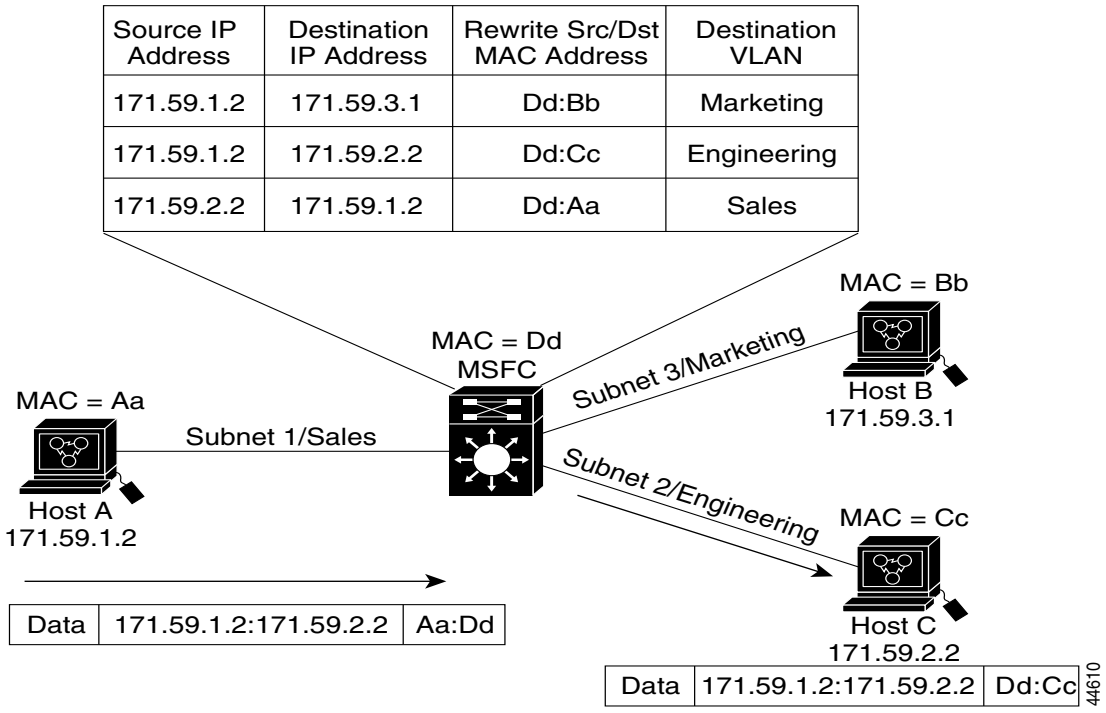
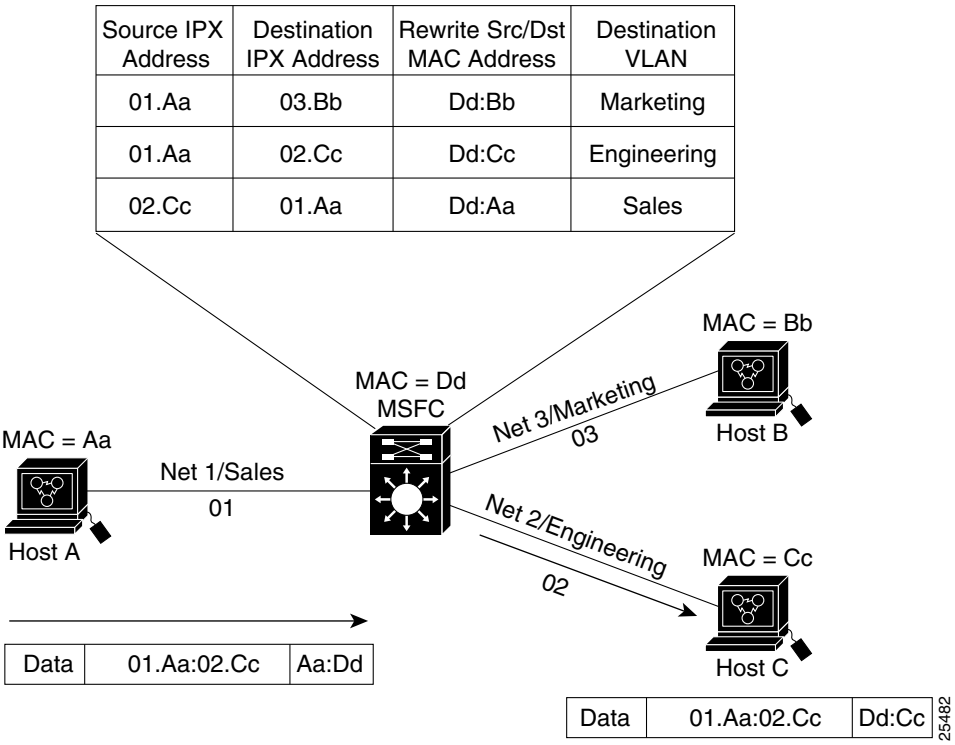


Figure 13-2 shows a simple IPX CEF network topology. In this example, Host A is on the Sales VLAN (IPX address 01.Aa), Host B is on the Marketing VLAN (IPX address 03.Bb), and Host C is on the Engineering VLAN (IPX address 02.Cc).

When Host A initiates a file transfer to Host C, PFC2 uses the information in the FIB and adjacency table to forward packets from Host A to Host C.

Figure 13-2 IPX CEF Example Topology



Understanding the NetFlow Statistics

Note With Supervisor Engine 720 (MSFC3), IPX routing is done through the software.

These sections describe NetFlow statistics:

- NetFlow Statistics Overview, page 13-11
- NetFlow Table Entry Aging, page 13-12
- Flow Masks, page 13-12

NetFlow Statistics Overview

CEF for PFC2/PFC3A generates flow statistics for Layer 3-switched traffic, which are stored in the NetFlow table. NetFlow statistics can be displayed with **show** commands and are also available to NetFlow Data Export (NDE).

Note A NetFlow table with more than 32,000 entries increases the probability that there will be insufficient room to store statistics. To reduce the number of entries in the NetFlow table, you can exclude specified IP protocols from the statistics or use the least granular flow mask (see the "Excluding the IP Protocol Entries from the NetFlow Table" section on page 13-31).

NetFlow statistics support unicast and multicast flows as follows:

- A unicast flow can be any of the following:
  - Destination only: All traffic to a particular IP destination
  - Destination-source: All traffic from a particular IP source to a particular IP destination
  - Full-flow: All traffic from a particular IP source to a particular IP destination that shares the same protocol and transport-layer information
- A multicast flow is all traffic with the same protocol and transport-layer information from a particular source to the members of a particular destination multicast group.

NetFlow Table Entry Aging

The state and identity of flows are maintained while packet traffic is active; when traffic for a flow ceases, the entry ages out. You can configure the aging time for the NetFlow table entries that are kept in the NetFlow table. If an entry is not used for the specified period of time, the entry ages out and statistics for that flow can be exported to a flow collector application.

Flow Masks

Flow masks determine how the NetFlow table entries are created. CEF for PFC2 supports only one flow mask (the most specific one) for all statistics. If NetFlow for PFC2 detects different flow masks from different MSFCs for which it is performing Layer 3 switching, it changes its flow mask to the most specific flow mask detected (this applies to the PFC2/MSFC2 only).

When the flow mask changes, the entire NetFlow table is purged. When CEF for PFC2/PFC3A exports cached entries, flow records are created based on the current flow mask. Depending on the current flow mask, some fields in the flow record might not have values. Unsupported fields are filled with a zero (0).

The statistics flow masks are as follows:

- destination-ip—The least-specific flow mask for IP
- destination-ipx—The only flow mask for IPX
- source-destination-ip—For IP
- source-destination-vlan—For IP multicast
- full flow—The most-specific flow mask
- full vlan—The same fields as in full flow plus the source VLAN

Enter the **show mls statistics entry** command to display the contents of the NetFlow table and the current flow mask. Use the keyword options to display information for specific traffic (refer to the *Catalyst 6500 Series Switch Command Reference* publication for more information).

Default CEF for PFC2/PFC3A Configuration

Table 13-2 shows the default CEF for PFC2/PFC3A configuration.

Table 13-2 Default CEF for PFC2/PFC3A Configuration

Feature	Default Value
CEF for PFC2 enable state	Enabled (cannot be disabled)
CEF enable state on MSFC2/MSFC3	Enabled (cannot be disabled)
Multicast services (IGMP snooping)	Enabled
Multicast services (GMRP)	Disabled
Multicast routing on MSFC2/MSFC3	Disabled globally
PIM routing on MSFC2/MSFC3	Disabled on all interfaces
IP MMLS Threshold	Unconfigured—no default value
IP MMLS	Enabled when multicast routing is enabled and IGMP snooping is enabled

CEF for PFC2/PFC3A Configuration Guidelines and Restrictions

Note With Supervisor Engine 720 (MSFC3), IPX routing is done through the software.

This section describes the guidelines and restrictions for configuring CEF for PFC2/PFC3A:

- PFC2 supports a maximum of 16 unique Hot Standby Router Protocol (HSRP) group numbers. You can use the same HSRP group numbers in different VLANs. If you configure more than 16 HSRP groups, this restriction prevents use of the VLAN number as the HSRP group number.

  Note Identically numbered HSRP groups use the same virtual MAC address, which might cause errors if you configure bridging on the MSFC.

- Because of the restriction to 16 unique HSRP group numbers, CEF for PFC2 cannot support the **standby use-bia** HSRP command.
- PFC3A supports 256 HSRP groups.
- CEF for PFC2 supports the following ingress and egress encapsulations:

  Note CEF for PFC3A supports Ethernet V2.0 (ARPA) only.

  – For IP unicast:
  Ethernet V2.0 (ARPA)
  802.3 with 802.2 with 1 byte control (SAP1)
  802.3 with 802.2 and SNAP

  – For IPX:
  Ethernet V2.0 (ARPA)
  802.3 (raw)
  802.2 with 1 byte control (SAP1)
  SNAP

Note When the ingress encapsulation for IPX traffic is SAP1, CEF for PFC2 provides Layer 3 switching only when the egress encapsulation is also SAP1. MSFC2 routes IPX SAP1 traffic that requires an encapsulation change.

  - For IP multicast—Ethernet V2.0 (ARPA)

CEF for PFC2/PFC3A does not provide Layer 3 switching for an IP multicast flow in the following cases:

- For IP multicast groups that fall into the range 224.0.0.* (where * is in the range 0–255), which is used by routing protocols. CEF for PFC2/PFC3A supports 225.0.0.* through 239.0.0.* and 224.128.0.* through 239.128.0.*.

  Note Groups in the 224.0.0.* range are reserved for routing control packets and must be flooded to all forwarding ports of the VLAN. These addresses map to the multicast MAC address range 01-00-5E-00-00-*xx,* where *xx* is in the range 0–0xFF.

- For PIM auto-RP multicast groups (IP multicast group addresses 224.0.1.39 and 224.0.1.40).

  Note In systems with redundant MSFC2s/MSFC3s, the PIM interface configuration must be the same on both the active and the redundant MSFC2/MSFC3.

- If the shortest-path tree (SPT) bit for the flow is cleared when running PIM sparse mode for the interface or group.
- For fragmented IP packets and packets with IP options. However, packets in the flow that are not fragmented or that do not specify IP options are multilayer switched.
- For source traffic that is received on tunnel interfaces (such as MBONE traffic).
- For any RPF interface with multicast tag switching enabled.

Configuring CEF for PFC2/PFC3A on the Switch

These sections describe how to configure CEF for PFC2/PFC3A:

Note For information on configuring routing on MSFC2/MSFC3, see Chapter 12, “Configuring InterVLAN Routing.”

Displaying the Layer 3-Switching Entries on the Supervisor Engine

CEF for PFC2/PFC3A is permanently enabled on Supervisor Engine 2 with PFC2 and MSFC2 and on Supervisor Engine 720 with PFC3A and MSFC3. No configuration is required.

To display all the Layer 3-switching entries on the supervisor engine, perform this task:

Task	Command
Display Layer 3-switching information.	**show mls entry** [**pbr-route**] [**cef**] | [**netflow-route**] [**qos**]

This example shows how to display the Layer 3-switching entries:

```
Console> (enable) show mls entry
Mod FIB-Type  Destination-IP  Destination-Mask NextHop-IP      Weight
--- --------- --------------- ---------------- --------------- ------
 15 receive   0.0.0.0         255.255.255.255
 15 receive   255.255.255.255 255.255.255.255
 15 receive   127.0.0.12      255.255.255.255
 16 receive   127.0.0.0       255.255.255.255
 16 receive   127.255.255.255 255.255.255.255
 15 resolved  127.0.0.11      255.255.255.255  127.0.0.11           1
 15 receive   21.2.0.4        255.255.255.255
 16 receive   21.0.0.0        255.255.255.255
 16 receive   21.255.255.255  255.255.255.255
 15 receive   44.0.0.1        255.255.255.255
 16 receive   44.0.0.0        255.255.255.255
 16 receive   44.255.255.255  255.255.255.255
 15 receive   42.0.0.1        255.255.255.255
 16 receive   42.0.0.0        255.255.255.255
 16 receive   42.255.255.255  255.255.255.255
 15 receive   43.0.0.99       255.255.255.255
 15 receive   43.0.0.0        255.255.255.255
 15 receive   43.255.255.255  255.255.255.255
 15 receive   192.20.20.20    255.255.255.255
 16 receive   21.2.0.5        255.255.255.255
 16 receive   42.0.0.20       255.255.255.255
 15 connected 43.0.0.0        255.0.0.0
 15 drop      224.0.0.0       240.0.0.0
 15 wildcard  0.0.0.0         0.0.0.0

Mod FIB-Type  Dest-IPX-net NextHop-IPX               Weight
--- --------- ------------ ------------------------ ------
 15 connected 21
 15 connected 44
 15 connected 42
 15 resolved  450          42.0050.3EA9.ABFD             1
 15 resolved  480          42.0050.3EA9.ABFD             1
 15 wildcard  0
```

```
Destination-IP  Source-IP       Prot  DstPrt SrcPrt Destination-Mac   Vlan EDst Stat-Pkts
Stat-Bytes  Uptime   Age      TcpDltSeq TcpDltAck
--------------- --------------- ----- ------ ------ ----------------- ---- ---- ----------
----------- -------- -------- --------- ---------
0.0.0.5         0.0.0.5         5     204    104    cc-cc-cc-cc-cc-cc 5    ARPA 0
0           01:03:18 01:00:51 cccccccc  cccccccc
0.0.0.2         0.0.0.2         2     201    101    cc-cc-cc-cc-cc-cc 2    ARPA 0
0           01:03:21 01:00:51 cccccccc  cccccccc
0.0.0.4         0.0.0.4         4     203    X      cc-cc-cc-cc-cc-cc 4    ARPA 0
0           01:03:19 01:00:51 cccccccc  cccccccc
0.0.0.1         0.0.0.1         ICMP  200    100    cc-cc-cc-cc-cc-cc 1    ARPA 0
0           01:03:25 01:00:52 cccccccc  cccccccc
0.0.0.3         0.0.0.3         3     202    102    cc-cc-cc-cc-cc-cc 3    ARPA 0
0           01:03:20 01:00:52 cccccccc  cccccccc
0.0.0.6         0.0.0.6         TCP   205    105    cc-cc-cc-cc-cc-cc 6    ARPA 0
0           01:03:18 01:00:52 cccccccc  cccccccc
Console> (enable)
```

Enter the **show mls entry cef** command to display only the FIB entries. Enter the **show mls entry netflow-route** command to display only the entries from the TCP intercept feature and reflexive access control lists (ACLs). Enter the **show mls entry pbr-route** command to display only the PBR entries. Enter the **show mls entry qos** command to display only the QoS entries.

Configuring CEF on MSFC2/MSFC3

CEF is permanently enabled on MSFC2/MSFC3. No configuration is required to support CEF for PFC2/PFC3A.

Note The **ip load-sharing per-packet**, **ip cef accounting per-prefix**, and **ip cef accounting non-recursive** Cisco IOS CEF commands on MSFC2/MSFC3 apply only to traffic that is switched by CEF on MSFC/MSFC3. The commands do not affect traffic that is switched by CEF for PFC2/PFC3A on the supervisor engine.

Specifying CEF Maximum Routes

Note This feature is only available with Supervisor Engine 720.

To specify the maximum number of routes that can be programmed in the FIB TCAM for a protocol, use the **set mls cef maximum-routes** {**ip** | **ip-multicast**} *routes* command. The syntax is as follows:

- ip—Specifies IP MLS.
- ip-multicast—Specifies IP multicasting MLS.
- routes—Specifies the number of routes that can be programmed in the FIB TCAM.

Follow these guidelines when specifying the maximum number of routes that can be programmed in the FIB TCAM:

- Routes that exceed the specified number of routes are not installed in the hardware. Packets that take those routes are switched by the MSFC. The routes argument is a unit of 1,000 entries. Setting the routes argument to 0 returns the system to a system-determined default value.
- When no protocols are set, an initial default value is assigned for each protocol. When at least one protocol is set, the default value for other unassigned protocols might change as the system tries to assign the remaining space to the unassigned protocols.

This command has the following characteristics:

- Changing the setting takes effect only after rebooting the active supervisor engine. The change does not take effect after a switchover.
- The setting on the standby supervisor engine is synchronized with the active supervisor engine. If the standby supervisor engine is inserted, both the bootup setting and new setting, if existing, on the active supervisor engine are synchronized with the standby supervisor engine. The standby supervisor engine uses the bootup setting to configure the FIB TCAM. The standby supervisor engine might need to be reset if its original bootup setting is different from the bootup setting of the active supervisor engine. An informational message (FIB_MAXROUTES_RESET) is printed on the active supervisor engine console if this situation occurs.
- To maximize the TCAM utilization, we recommend that you set the maximum routes for IP unicast as a multiple of 16,000 and set the maximum routes for IP multicast as a multiple of 8,000. The internal allocation scheme uses 16,000 as the allocation unit for unicast and 8,000 as the allocation unit for multicast. For example, if IP unicast is set to 1,000, 16,000 entries are reserved, but only 1,000 is allowed.
- When the maximum routes are exceeded or the allocated TCAM space for a protocol is full, a system message (FIB_ALLOC_TCAM_FULL) displays. Because of the internal software allocation scheme, the allocated TCAM space might be full before the maximum routes are exceeded.

Note The sum of the number of maximum routes for all protocols cannot exceed 256,000.

Note If the routes values for all protocols are set to 0, the bootup default is used. When you set the routes value for one protocol to a non-zero value, the default value for the other protocol changes to the remaining size.

Note If the maximum number of routes is not set for an MLS protocol, a system-determined default value is shown. The default value for a protocol might not be fixed, as the system tries to assign the remaining space to the unassigned protocols. If the maximum-routes configuration is changed after bootup, the **show mls cef maximum-routes** command displays two kinds of information: one for the current (bootup) configuration and the other for the new configuration that takes effect after reboot.

To specify the maximum number of routes that can be programmed in the FIB TCAM for a protocol, perform these tasks in privileged mode:

Task	Command
Specify the maximum number of routes that can be programmed in the FIB TCAM for a protocol.	**set mls cef maximum-routes** {**ip** | **ip-multicast**} *routes*
Display the maximum number of routes that are configured for each MLS protocol.	**show mls cef maximum-routes**

This example shows how to specify the maximum number of routes for IP unicast:

```
Console> (enable) set mls cef maximum-routes ip 220
Configuration change will take effect after next reboot.
Console> (enable) show mls cef maximum-routes
Current:
    IPv4           :192k (default)
    IPv4 multicast : 32k (default)
User configured:(effective after reboot)
    IPv4           :220k
    IPv4 multicast : 16k (adjusted default)
Console> (enable)
```

Configuring IP Multicast on MSFC2/MSFC3

These sections describe how to configure MSFC2/MSFC3 for IP multicast:

- Enabling IP Multicast Routing Globally, page 13-18
- Enabling IP PIM on an MSFC2/MSFC3 Interface, page 13-19
- Configuring the IP MMLS Global Threshold, page 13-19
- Enabling IP MMLS on MSFC2/MSFC3 Interfaces, page 13-20

Note This section describes how to enable IP multicast routing on MSFC2/MSFC3. For more detailed IP multicast configuration information, refer to the "IP Multicast" section of the *Cisco IOS IP and IP Routing Configuration Guide* at this URL: http://www.cisco.com/en/US/docs/ios/12_1/iproute/configuration/guide/ip_c.html

Enabling IP Multicast Routing Globally

You must enable IP multicast routing globally on MSFC2/MSFC3 before you can enable PIM on MSFC2/MSFC3 interfaces.

To enable IP multicast routing globally on MSFC2/MSFC3, perform this task in global configuration mode:

Task	Command
Enable IP multicast routing globally.	Router(config)# **ip multicast-routing**

This example shows how to enable IP multicast routing globally:

```
Router(config)# ip multicast-routing
Router(config)#
```

Enabling IP PIM on an MSFC2/MSFC3 Interface

You must enable PIM on MSFC2/MSFC3 interfaces before IP multicast will function on those interfaces.

To enable IP PIM on an MSFC2/MSFC3 interface, perform this task in interface configuration mode:

Task	Command
Enable IP PIM on an MSFC2/MSFC3 interface.	Router(config-if)# **ip pim** {**dense-mode** | **sparse-mode** | **sparse-dense-mode**}

This example shows how to enable PIM on an MSFC2/MSFC3 interface using the default mode (**sparse-dense-mode**):

```
Router(config-if)# ip pim
Router(config-if)#
```

This example shows how to enable PIM sparse mode on an MSFC2/MSFC3 interface:

```
Router(config-if)# ip pim sparse-mode
Router(config-if)#
```

Configuring the IP MMLS Global Threshold

You can configure a global multicast rate threshold, specified in packets per second, below which all multicast traffic is routed by MSFC2/MSFC3. This prevents creation of MLS entries for short-lived multicast flows, such as join requests.

Note This command does not affect flows that are already being routed. To apply the threshold to existing routes, clear the route and let it reestablish.

To configure the IP MMLS threshold, perform this task:

Task	Command
Configure the IP MMLS threshold.	Router(config)# [**no**] **mls ip multicast threshold** *ppsec*

This example shows how to configure the IP MMLS threshold to 10 packets per second:

```
Router(config)# mls ip multicast threshold 10
Router(config)#
```

Use the **no** keyword to deconfigure the threshold.

Enabling IP MMLS on MSFC2/MSFC3 Interfaces

IP MMLS is enabled by default on the MSFC2/MSFC3 interface when you enable IP PIM on the interface. Perform this task only if you disabled IP MMLS on the interface and you want to reenable it.

Note You must enable IP PIM on all participating MSFC2/MSFC3 interfaces before IP MMLS will function. For information on configuring IP PIM on MSFC2/MSFC3 interfaces, see the "Enabling IP PIM on an MSFC2/MSFC3 Interface" section on page 13-19.

To enable IP MMLS on an MSFC2/MSFC3 interface, perform this task:

Task	Command
Enable IP MMLS on an MSFC2/MSFC3 interface.	Router(config-if)# [**no**] **mls ip multicast**

This example shows how to enable IP MMLS on an MSFC2/MSFC3 interface:

```
Router(config-if)# mls ip multicast
Router(config-if)#
```

Use the **no** keyword to disable IP MMLS on an MSFC2/MSFC3 interface.

Displaying IP Multicast Information

These sections describe how to display IP multicast information:

Displaying IP Multicast Information on MSFC2/MSFC3

These sections describe displaying IP multicast information on MSFC2/MSFC3:

Displaying IP MMLS Interface Information

The **show ip pim interface count** command displays the IP MMLS enable state on MSFC2/MSFC3 IP PIM interfaces and the number of packets that are received and sent on the interface. The output lists the PIM interfaces that are fast switched and process switched, and the packet counts for these interfaces. An "H" is displayed on interfaces where IP MMLS is enabled.

The **show ip interface** command displays the IP MMLS enable state on an MSFC2/MSFC3 interface.

To display IP MMLS information for an IP PIM MSFC2/MSFC3 interface, perform one of these tasks:

Task	Command
Display IP MMLS interface information.	Router# **show ip pim interface** [*type number*] **count**
Display the IP MMLS interface enable state.	Router# **show ip interface**

This example shows how to display information about the IP MMLS interfaces:

```
Router# show ip pim interface count
States: FS - Fast Switched, H - Hardware Switched

Address          Interface          FS  Mpackets In/Out

192.168.10.2       Vlan10             * H 40886/0

192.168.11.2       Vlan11             * H 0/40554

192.168.12.2       Vlan12             * H 0/40554

192.168.23.2       Vlan23             *   0/0

192.168.24.2       Vlan24             *   0/0

Router#
```

Displaying the IP Multicast Routing Table

The **show ip mroute** command displays the IP multicast routing table on MSFC2/MSFC3.

To display the IP multicast routing table, perform this task:

Task	Command
Display the IP multicast routing table.	Router# **show ip mroute** [*group*[*source*]] | [**summary**] | [**count**] | [**active** *kbps*]

This example shows how to display the IP multicast routing table:

```
Router# show ip mroute 239.252.1.1
IP Multicast Routing Table
Flags:D - Dense, S - Sparse, C - Connected, L - Local, P - Pruned
       R - RP-bit set, F - Register flag, T - SPT-bit set, J - Join SPT
       M - MSDP created entry, X - Proxy Join Timer Running
       A - Advertised via MSDP
Outgoing interface flags:H - Hardware switched
Timers:Uptime/Expires
Interface state:Interface, Next-Hop or VCD, State/Mode

(*, 239.252.1.1), 04:04:59/00:02:59, RP 80.0.0.2, flags:SJ
  Incoming interface:Vlan800, RPF nbr 80.0.0.2
  Outgoing interface list:
    Vlan10, Forward/Dense, 01:29:57/00:00:00, H

(22.0.0.10, 239.252.1.1), 00:00:19/00:02:41, flags:JT
  Incoming interface:Vlan800, RPF nbr 80.0.0.2, RPF-MFD
  Outgoing interface list:
    Vlan10, Forward/Dense, 00:00:19/00:00:00, H
```

Displaying IP Multicast Details

The **show mls ip multicast** command displays detailed information about IP MMLS.

To display detailed MMLS information on MSFC2/MSFC3, perform one of these tasks:

Task	Command
Display IP MMLS group information.	Router# **show mls ip multicast group** *group-address* [**interface** *type number* | **statistics**]
Display IP MMLS details for all interfaces.	Router# **show mls ip multicast interface** *type number* [**statistics** | **summary**]
Display a summary of IP MMLS information.	Router# **show mls ip multicast summary**
Display IP MMLS statistics.	Router# **show mls ip multicast statistics**
Display IP MMLS source information.	Router# **show mls ip multicast source** *ip-address* [**interface** *type number* | **statistics**]

This example shows how to display IP MMLS statistics on MSFC2/MSFC3:

```
Router# show mls ip multicast statistics
MLS Multicast configuration and state:
    Router Mac:0050.0f2d.9bfd, Router IP:1.12.123.234
    MLS multicast operating  state:ACTIVE
    Maximum number of allowed outstanding messages:1
    Maximum size reached from feQ:1
    Feature Notification sent:5
    Feature Notification Ack received:4
    Unsolicited Feature Notification received:0
    MSM sent:33
    MSM ACK received:33
    Delete notifications received:1
    Flow Statistics messages received:248
```

```
MLS Multicast statistics:
    Flow install Ack:9
    Flow install Nack:0
    Flow update Ack:2
    Flow update Nack:0
    Flow delete Ack:0
    Complete flow install Ack:10
    Complete flow install Nack:0
    Complete flow delete Ack:1
    Input VLAN delete Ack:4
    Output VLAN delete Ack:0
    Group delete sent:0
    Group delete Ack:0
    Global delete sent:7
    Global delete Ack:7

    L2 entry not found error:0
    Generic error :3
    LTL entry not found error:0
    MET entry not found error:0
    L3 entry exists error :0
    Hash collision error :0
    L3 entry not found error:0
    Complete flow exists error :0
```

This example shows how to display information on a specific IP MMLS entry on MSFC2/MSFC3:

```
Router# show mls ip multicast 224.1.1.1
Multicast hardware switched flows:
(1.1.13.1, 224.1.1.1) Incoming interface: Vlan13, Packets switched: 61590
Hardware switched outgoing interfaces: Vlan20 Vlan9
RFD-MFD installed: Vlan13

(1.1.9.3, 224.1.1.1) Incoming interface: Vlan9, Packets switched: 0
Hardware switched outgoing interfaces: Vlan20
RFD-MFD installed: Vlan9

(1.1.12.1, 224.1.1.1) Incoming interface: Vlan12, Packets switched: 62010
Hardware switched outgoing interfaces: Vlan20 Vlan9
RFD-MFD installed: Vlan12

(1.1.12.3, 224.1.1.1) Incoming interface: Vlan12, Packets switched: 61980
Hardware switched outgoing interfaces: Vlan20 Vlan9
RFD-MFD installed: Vlan12

(1.1.11.1, 224.1.1.1) Incoming interface: Vlan11, Packets switched: 62430
Hardware switched outgoing interfaces: Vlan20 Vlan9
RFD-MFD installed: Vlan11

(1.1.11.3, 224.1.1.1) Incoming interface: Vlan11, Packets switched: 62430
Hardware switched outgoing interfaces: Vlan20 Vlan9
RFD-MFD installed: Vlan11

Total hardware switched installed: 6
Router#
```

This example shows how to display a summary of IP MMLS information on MSFC2/MSFC3:

```
Router# show mls ip multicast summary
7 MMLS entries using 560 bytes of memory
Number of partial hardware-switched flows:2
Number of complete hardware-switched flows:5
Router#
```

Using the Debug Commands

Table 13-3 describes the IP MMLS-related debug troubleshooting commands.

Table 13-3 IP MMLS Debug Commands

Command	Description
[**no**] **debug mls ip multicast group** *group_id group_mask*	Configures filtering that applies to all other multicast debugging commands.
[**no**] **debug mls ip multicast events**	Displays the IP MMLS events.
[**no**] **debug mls ip multicast errors**	Turns on the debug messages for multicast MLS-related errors.
[**no**] **debug mls ip multicast messages**	Displays the IP MMLS messages from/to the hardware switching engine.
[**no**] **debug mls ip multicast all**	Turns on all the IP MMLS messages.
[**no**] **debug mdss error**	Turns on the Multicast Distributed Switching Services (MDSS) error messages.
[**no**] **debug mdss events**	Turns on the MDSS-related events.
[**no**] **debug mdss all**	Turns on all the MDSS messages.

Using the Debug Commands on the SCP

Table 13-4 describes the Serial Control Protocol (SCP)-related debug commands to troubleshoot the SCP that runs over the Ethernet out-of-band channel (EOBC).

Table 13-4 SCP Debug Commands

Command	Description
[**no**] **debug scp async**	Displays the trace for asynchronous data in and out of the SCP system.
[**no**] **debug scp data**	Shows the packet data trace.
[**no**] **debug scp errors**	Displays the errors and warnings in the SCP.
[**no**] **debug scp packets**	Displays the packet data in and out of the SCP system.
[**no**] **debug scp timeouts**	Reports timeouts.
[**no**] **debug scp all**	Turns on all SCP debugging messages.

Displaying the IP Multicast Information on the Supervisor Engine

These sections describe how to display the IP multicast information:

- Displaying the IP Multicast Statistics, page 13-25
- Clearing the IP Multicast Statistics, page 13-26
- Displaying the IP Multicast Entries, page 13-26

Displaying the IP Multicast Statistics

The **show mls multicast statistics** command displays the IP multicast statistics.

To display the IP multicast statistics, perform this task:

Task	Command
Display the IP multicast statistics.	**show mls multicast statistics** [*ip_addr*]

This example shows how to display the IP multicast statistics for MSFC2/MSFC3:

```
Console (enable) show mls multicast statistics
Router IP          Router Name        Router MAC
-----------------------------------------------------------
1.1.9.254          ?                  00-50-0f-06-3c-a0

Transmit:
  Delete Notifications:                    23
  Acknowledgements:                        92
  Flow Statistics:                         56

Receive:
  Open Connection Requests:                1
  Keep Alive Messages:                     72
  Shortcut Messages:                       19
    Shortcut Install TLV:                  8
    Selective Delete TLV:                  4
    Group Delete TLV:                      0
    Update TLV:                            3
    Input VLAN Delete TLV:                 0
    Output VLAN Delete TLV:                0
    Global Delete TLV:                     0
    MFD Install TLV:                       7
    MFD Delete TLV:                        0
Router IP          Router Name        Router MAC
-----------------------------------------------------------
1.1.5.252          ?                  00-10-29-8d-88-01

Transmit:
  Delete Notifications:                    22
  Acknowledgements:                        75
  Flow Statistics:                         22

Receive:
  Open Connection Requests:                1
  Keep Alive Messages:                     68
  Shortcut Messages:                       6
    Shortcut Install TLV:                  4
    Selective Delete TLV:                  2
    Group Delete TLV:                      0
    Update TLV:                            0
    Input VLAN Delete TLV:                 0
    Output VLAN Delete TLV:                0
    Global Delete TLV:                     0
    MFD Install TLV:                       4
    MFD Delete TLV:                        0
Console (enable)
```

Clearing the IP Multicast Statistics

The **clear mls multicast statistics** command clears the IP multicast statistics.

To clear the IP multicast statistics, perform this task in privileged mode:

Task	Command
Clear the IP multicast statistics.	**clear mls multicast statistics**

This example shows how to clear the IP multicast statistics:

```
Console> (enable) clear mls multicast statistics
All statistics for the MLS routers in include list are cleared.
Console> (enable)
```

Displaying the IP Multicast Entries

The **show mls multicast entry** command displays a variety of information about the multicast flows that are being handled by PFC2/PFC3A. You can display entries that are based on any combination of the participating MSFC2/MSFC3, the VLAN, the multicast group address, or the multicast traffic source.

To display information about the IP multicast entries, perform this task in privileged mode:

Task	Command
Display information about the IP multicast entries.	**show mls multicast entry** [[[*mod*] [**vlan** *vlan_id*] [**group** *ip_addr*] [**source** *ip_addr*]] | [**all**]]

This example shows how to display all the IP multicast entries:

```
Console> (enable) show mls multicast entry all
Router IP       Dest IP         Source IP       Pkts       Bytes       InVlan  OutVlans
--------------- --------------- --------------- ---------- ----------- ------- --------

1.1.5.252       224.1.1.1       1.1.11.1        15870      2761380     20
1.1.9.254       224.1.1.1       1.1.12.3        473220     82340280    12
1.1.5.252       224.1.1.1       1.1.12.3        15759      2742066     20
1.1.9.254       224.1.1.1       1.1.11.1        473670     82418580    11
1.1.5.252       224.1.1.1       1.1.11.3        15810      2750940     20
1.1.9.254       224.1.1.1       1.1.12.1        473220     82340280    12
1.1.5.252       224.1.1.1       1.1.13.1        15840      2756160     20
1.1.9.254       224.1.1.1       1.1.13.1        472770     82261980    13
1.1.5.252       224.1.1.1       1.1.12.1        15840      2756160     20
1.1.9.254       224.1.1.1       1.1.11.3        473667     82418058    11
Total Entries: 10
Console> (enable)
```

This example shows how to display the IP multicast entries for a specific MSFC2/MSFC3:

```
Console> (enable) show mls multicast entry 15
Router IP       Dest IP         Source IP       Pkts       Bytes       InVlan  OutVlans
--------------- --------------- --------------- ---------- ----------- ------- --------

1.1.5.252       224.1.1.1       1.1.11.1        15870      2761380     20
1.1.5.252       224.1.1.1       1.1.12.3        15759      2742066     20
1.1.5.252       224.1.1.1       1.1.11.3        15810      2750940     20
1.1.5.252       224.1.1.1       1.1.13.1        15840      2756160     20
1.1.5.252       224.1.1.1       1.1.12.1        15840      2756160     20
Total Entries: 5
Console> (enable)
```

This example shows how to display the IP multicast entries for a specific multicast group address:

```
Console> (enable) show mls multicast entry group 226.0.1.3 short
Router IP      Dest IP     Source IP    InVlan Pkts   Bytes     OutVlans
-------------- ----------- ------------ ------ ------ --------- ---------
171.69.2.1     226.0.1.3   172.2.3.8    20     171    23512     10,201,22,45
171.69.2.1     226.0.1.3   172.3.4.9    12     25     3120      8,20
Total Entries: 2
Console> (enable)
```

This example shows how to display the IP multicast entries for a specific MSFC2/MSFC3 and a specific multicast source address:

```
Console> (enable) show mls multicast entry 15 source 1.1.11.1 short
Router IP       Dest IP         Source IP       Pkts       Bytes
 InVlan  OutVlans
--------------- --------------- --------------- ---------- --------------------
  ------ ----------
172.20.49.159   224.1.1.6       1.1.40.4        368        57776
  40     23,25
172.20.49.159   224.1.1.71      1.1.22.2        99         65142
  22     30,37
172.20.49.159   224.1.1.8       1.1.22.2        396        235620
  22     13,19
Console> (enable)
```

Configuring the NetFlow Statistics on the Switch

Note With Supervisor Engine 720 (MSFC3), IPX routing is done through the software.

These sections describe how to configure the NetFlow statistics:

Specifying NetFlow Table Entry Creation on a Per-Interface Basis

Note This feature requires PFC3B, PFC3BXL or later.

With software release 8.4(1) and later releases, you can create the NetFlow table entries on a per-interface basis. This feature uses the same mechanism as the bridged-flow statistics to create flows. The NetFlow entries are created for both VLANs on which the bridged-flow statistics are enabled and on the VLANs on which NetFlow entry creation is enabled (see the "Enabling and Disabling Bridged-Flow Statistics on VLANs" section on page 16-12).

For example, if you enable the Layer 3 per-interface entry creation on VLAN 100 and 200 and at the same time you want to enable the bridged-flow statistics on VLAN 150 and 250, the NetFlow entry and the bridged-flow statistics are enabled on all four VLANs. To specify only the bridged-flow statistics for VLAN 150 and 250, you must disable the per-interface entry feature.

In addition, the bridged-flow statistics are automatically enabled when you enable the NetFlow entry creation on a per-interface basis for VLANs. The CLI allows you to disable NetFlow per interface if you do not want this overlap in the Netflow table entry creation.

The status of this feature is displayed as part of the **show mls** command. The VLANs that have entry creation enabled are displayed as part of the VLANs that have the bridged-flow statistics feature enabled.

To enable or disable the NetFlow per-interface table entries, perform this task in privileged mode:

Task	Command
Enable the NetFlow per-interface table entries.	**set mls netflow-per-interface** [**enable** | **disable**]

This example shows how to enable the NetFlow per-interface table entries:

```
Console> (enable) set mls netflow-per-interface enable
Console> (enable)
```

You can specify the VLANs for which the NetFlow entries can be enabled or disabled. To control the flow creation on a VLAN basis, perform this task in privileged mode:

Task	Command
Enable the per-VLAN NetFlow table entries.	**set mls netflow-entry-create** [**enable** | **disable**] *vlan-list*

This example shows how to specify the VLANs that are used to create the NetFlow table entries:

```
Console> (enable) set mls netflow-entry-create enable 150, 250
Console> (enable)
```

Specifying the NetFlow Table Entry Aging-Time Value

The entry aging time for each protocol (IP and IPX) applies to all the protocol-specific NetFlow table entries. Any entry that has not been used for *agingtime* seconds is aged out. The default is 16 seconds.

For normal aging time, you can specify the aging time in the range of 1–1092 seconds in 8-second increments. Any aging-time value that is not a multiple of 8 seconds is adjusted to the closest multiple of 8 seconds. For example, a value of 65 is adjusted to 64 and a value of 127 is adjusted to 128.

To specify the entry aging time for both IP and IPX, perform this task in privileged mode:

Task	Command
Specify the aging time for the NetFlow table entries.	**set mls agingtime** [*agingtime*]

This example shows how to specify the entry aging time:

```
Console> (enable) set mls agingtime 16
Multilayer switching agingtime IP and IPX set to 16
Console> (enable)
```

To specify the IP entry aging time, perform this task in privileged mode:

Task	Command
Specify the IP entry aging time for the NetFlow table.	**set mls agingtime ip** [*agingtime*]

This example shows how to specify the IP entry aging time:

```
Console> (enable) set mls agingtime ip 16
Multilayer switching aging time IP set to 16
Console> (enable)
```

To specify the IPX entry aging time, perform this task in privileged mode:

Task	Command
Specify the IPX entry aging time for the NetFlow table.	**set mls agingtime ipx** [*agingtime*]

This example shows how to specify the IPX entry aging time:

```
Console> (enable) set mls agingtime ipx 16
Multilayer switching aging time IPX set to 16
Console> (enable)
```

Specifying the NetFlow Table IP Entry Fast Aging Time and Packet Threshold Values

Note The IPX entries do not use fast aging.

To increase the utilization of the NetFlow table, enable IP entry fast aging time. The IP entry fast aging time applies to the NetFlow table entries that have no more than *pkt_threshold* packets that are routed within *fastagingtime* seconds after they are created. A typical NetFlow table entry that is removed is the entry for flows to and from a Domain Name Server (DNS) or TFTP server; the entry might never be used again after it is created. Detecting and aging out these entries saves space in the NetFlow table for other data traffic.

The default *fastagingtime* value is 0 (no fast aging). For Supervisor Engine 1 and Supervisor Engine 2, you can configure the *fastagingtime* value from 8–128 seconds in increments of 8 seconds. For Supervisor Engine 720, you can configure the *fastagingtime* value from 0–128 seconds in increments of 1 second. Any *fastagingtime* value that is not configured exactly as the indicated values is adjusted to the closest one. For Supervisor Engine 1 and Supervisor Engine 2, you can configure the *pkt_threshold* value to 0, 1, 3, 7, 15, 31, 63, or 127 packets. For Supervisor Engine 720, you can configure the *pkt_threshold* value from 1–127 packets in increments of 1 packet.

If you need to enable IP entry fast aging time, initially set the value to 128 seconds. If the NetFlow table remains full, decrease the setting. If the NetFlow table continues to remain full, decrease the normal IP entry aging time.

To specify the IP entry fast aging time and packet threshold, perform this task in privileged mode:

Task	Command
Specify the IP entry fast aging time and packet threshold for a NetFlow table entry.	**set mls agingtime fast** [*fastagingtime*] [*pkt_threshold*]

This example shows how to set the IP entry fast aging time to 8 seconds with a packet threshold of 15 packets:

```
Console> (enable) set mls agingtime fast 8 15
Multilayer switching fast aging time set to 8 seconds for entries with no more than 15
packets switched.
Console> (enable)
```

You can force an active flow to age out by entering the **set mls agingtime long-duration** {*longagingtime*} command. You can specify the aging time of the active flow in the range of 64–1920 seconds in increments of 64. The default *longagingtime* is 320.

This example shows how to set the aging time for active flows:

```
Console> (enable) set mls agingtime long-duration 128
Multilayer switching agingtime set to 128 seconds for long duration flows
Console> (enable)
```

Setting the Minimum Statistics Flow Mask

You can set the minimum granularity of the flow mask for the NetFlow table. The actual flow mask will be at least of the granularity that is specified by this command. For information on how the different flow masks work, see the "Flow Masks" section on page 13-12.

Note Entering the **set mls flow** command purges all the existing entries in the NetFlow table.

To set the minimum NetFlow statistics flow mask, perform this task in privileged mode:

Task	Command
Set the minimum statistics flow mask.	**set mls flow** {**destination** | **destination-source** | **null** | **full**}

This example shows how to set the minimum statistics flow mask to destination-source-ip:

```
Console> (enable) set mls flow destination-source
Configured IP flow mask is set to destination-source flow.
Console> (enable)
```

Excluding the IP Protocol Entries from the NetFlow Table

You can configure the NetFlow table to exclude specified IP protocols.

To exclude the IP protocols from the NetFlow table, perform this task in privileged mode:

Task	Command
Exclude the IP protocols from the NetFlow table.	**set mls exclude protocol** {**tcp** | **upd** | **both**} *port*

The *port* parameter can be a port number or a keyword: **dns**, **ftp**, **smtp**, **telnet**, **x** (X-Windows), or **www**.

This example shows how to exclude the Telnet traffic from the NetFlow table:

```
Console> (enable) set mls exclude protocol tcp telnet
NetFlow table will not create entries for TCP packets with protocol port 23.
Note: MLS exclusion only works in full flow mode.
Console> (enable)
```

Displaying the NetFlow Statistics

Note To display the forwarding decision entries, enter the **show mls entry cef** command (see the "Displaying the Layer 3-Switching Entries on the Supervisor Engine" section on page 13-15).

To display a summary of the NetFlow table entries and statistics, perform this task in privileged mode:

Task	Command
Display all the NetFlow table entries and statistics.	**show mls**

This example shows how to display all the NetFlow table entries (the display is from a Supervisor Engine 2):

```
Console> (enable) show mls
show mls
=======
Total packets switched = 2
Total bytes switched = 112
Total routes = 48
IP statistics flows aging time = 16 seconds
IP statistics flows fast aging time = 0 seconds, packet threshold = 0
IP Current flow mask is Full flow
Netflow Data Export version:7
Netflow Data Export disabled
Netflow Data Export port/host is not configured.
Total packets exported = 0

IPX statistics flows aging time = 16 seconds
IPX flow mask is Destination flow
IPX max hop is 15

Module 15:Physical MAC-Address 00-50-3e-a9-ab-fc
Vlan Virtual MAC-Address(es)
---- -----------------------
  42 00-00-0c-07-ac-00
Console>
```

This example shows how to display all the NetFlow table entries (the display is from a Supervisor Engine 720):

```
Console> (enable) show mls
Total packets switched = 35254
Total bytes switched = 2256256
Total routes = 120569
Total number of Netflow entries = 120000

IP statistics flows aging time = 50 seconds
Long-duration flows aging time = 320 seconds
IP statistics flows fast aging time = 0 seconds, packet threshold = 0

IP Current flow mask is Full-Vlan flow
Netflow Data Export version: 7
Netflow Data Export disabled
Netflow Data Export port/host is not configured.
Total packets exported = 0
Destination Ifindex export is enabled
Source Ifindex export is enabled
Rate limiting is turned off, packets are bridged to router
Load balancing hash is based on source and destination IP addresses and universc
Per-prefix Stats for ALL FIB entries is Enabled
Console> (enable)
```

The **show mls statistics entry** command can display all statistics or the statistics for the specific NetFlow table entries. Specify the destination address, source address, and for IP, the protocol, and source and destination ports to see the statistics for a specific NetFlow table entry.

A value of zero (0) for *src_port* or *dst_port* is treated as a wildcard, and all the NetFlow statistics are displayed (unspecified options are treated as wildcards). If the protocol that is specified is not TCP or UDP, set the *src_port* and *dst_prt* to 0 or no NetFlow statistics will display.

To display the statistics for the NetFlow table entries, perform this task in privileged mode:

Task	Command
Display the statistics for the NetFlow table entries. If you do not specify a NetFlow table entry, all the NetFlow statistics are shown.	**show mls statistics entry** [**ip** | **ipx** | **uptime**] [**destination** *ip_addr_spec*] [**source** *ip_addr_spec*] [**flow** *protocol src_port dst_port*]

This example shows how to display the NetFlow statistics for a particular NetFlow table entry:

```
Console> show mls statistics entry ip destination 172.20.22.14
                                  Last    Used
Destination IP  Source IP       Prot DstPrt SrcPrt Stat-Pkts Stat-Bytes
--------------- --------------- ---- ------ ------ --------- -----------
MSFC 127.0.0.12:
172.20.22.14    172.20.25.10    6    50648  80     3152       347854
Console>
```

The **show mls statistics entry ip top-talkers** command can display the statistics for the netflows with the maximum amount of network usage. The NetFlow entries are pulled out of the NetFlow table based on the number of packets that each flow has. The results are displayed in descending order with the top talkers being the entries with the largest packet count. You can get the statistics for the network (the top 32 talkers are displayed) or for a specified number of flows such as the top 1 or 2 talkers.

To display the NetFlow top talkers for the NetFlow table entries, perform this task in privileged mode:

Task	Command
Display the NetFlow talkers with the maximum amount of network usage.	**show mls statistics entry ip top-talkers**

This example shows how to display the NetFlow top talkers for a network:

```
Console> show mls statistics entry ip top-talkers
Last    Used
Destination IP   Source IP       Prot  DstPrt SrcPrt Vlan  Stat-Pkts  Stat-Bytes
---------------- --------------- ----- ------ ------ ----- ---------  -----------
 12.0.0.5        11.0.0.6        255   N/A    N/A    N/A   387110     17807060
 12.0.0.5        11.0.0.7        255   N/A    N/A    N/A   387109     17807014
 12.0.0.5        11.0.0.4        TCP   8      7      N/A   20         920
 127.0.0.20      127.0.0.19      UDP   67     68     N/A   18         828
 12.0.0.5        11.0.0.2        TCP   6      5      N/A   15         690
 12.0.0.5        11.0.0.5        TCP   8      7      N/A   15         690
 12.0.0.5        11.0.0.3        TCP   6      5      N/A   12         552
Console>
```

This example shows how to display the statistics for a specified number of NetFlows with the maximum network usage:

```
Console> show mls statistics entry ip top-talkers 2
Last    Used
Destination IP   Source IP       Prot  DstPrt SrcPrt Vlan  Stat-Pkts  Stat-Bytes
---------------- --------------- ----- ------ ------ ----- ---------  -----------
 12.0.0.5        11.0.0.6        255   N/A    N/A    N/A   387110     17807060
 12.0.0.5        11.0.0.7        255   N/A    N/A    N/A   387109     17807014
Console>
```

Clearing the NetFlow IP and IPX Statistics

These sections describe how to clear the NetFlow statistics:

Note The **clear mls** commands affect only the statistics. None of the **clear mls** commands affect the forwarding entries or the NetFlow table entries that correspond to the forwarding entries.

Clearing All the NetFlow Statistics

To clear all the NetFlow IP and IPX statistics, perform this task in privileged mode:

Task	Command
Clear all the NetFlow statistics.	**clear mls statistics entry all**

This example shows how to clear all the NetFlow statistics:

```
Console> (enable) clear mls statistics entry all
All MLS IP and IPX entries cleared.
Console> (enable)
```

Clearing the NetFlow IP Statistics

The **clear mls statistics entry ip** command clears the NetFlow IP statistics. Use the **all** keyword to clear all the NetFlow IP statistics. The **destination** and **source** keywords specify the source and destination IP addresses. The destination and source *ip_addr_spec* can be a full IP address or a subnet address in the format *ip_subnet_addr*, *ip_addr/subnet_mask*, or *ip_addr/subnet_mask_bits*.

The **flow** keyword specifies the following additional flow information:

- Protocol family (*protocol*)—Specify **tcp**, **udp**, **icmp**, or a decimal number for other protocol families. A value of zero (0) for *protocol* is treated as a wildcard (unspecified options are treated as wildcards).
- TCP or UDP source and destination port numbers (*src_port* and *dst_port*)—If the protocol that you specify is TCP or UDP, specify the source and destination TCP or UDP port numbers. A value of zero (0) for *src_port* or *dst_port* is treated as a wildcard (unspecified options are treated as wildcards). For other protocols, set the *src_port* and *dst_port* to 0, or no entries will clear.

To clear the statistics for a NetFlow table IP entry, perform this task in privileged mode:

Task	Command
Clear the statistics for a NetFlow table IP entry.	**clear mls statistics entry ip** [**destination** *ip_addr_spec*] [**source** *ip_addr_spec*] [**flow** *protocol src_port dst_port*] [**all**]

This example shows how to clear the statistics for NetFlow table entries with destination IP address 172.20.26.22:

```
Console> (enable) clear mls statistics entry ip destination 172.20.26.22
MLS IP entry cleared
Console> (enable)
```

This example shows how to clear the statistics for the NetFlow table entries with destination IP address 172.20.22.113, TCP source port 1652, and TCP destination port 23:

```
Console> (enable) clear mls statistics entry ip destination 172.20.26.22 source
172.20.22.113 flow tcp 1652 23
MLS IP entry cleared
Console> (enable)
```

Clearing the NetFlow IPX Statistics

The **clear mls statistics entry ipx** command clears the NetFlow IPX statistics. Use the **all** keyword to clear all the NetFlow IPX statistics. The **destination** and **source** keywords specify the source and destination IPX addresses.

To clear the statistics for a NetFlow table IPX entry, perform this task in privileged mode:

Task	Command
Clear the statistics for a NetFlow table IPX entry.	**clear mls statistics entry ipx** [**destination** *ipx_addr_spec*] [**source** *ipx_addr_spec*] [**all**]

This example shows how to clear the statistics for the IPX MLS entries with destination IPX address 1.0002.00e0.fefc.6000:

```
Console> (enable) clear mls statistics entry ipx destination 1.0002.00e0.fefc.6000
MLS IPX entry cleared.
Console> (enable)
```

Clearing the NetFlow Statistics Totals

The **clear mls statistics** command clears the following NetFlow statistics:

- Total packets that are switched (IP and IPX)
- Total packets that are exported (for NDE)

To clear the NetFlow statistic totals, perform this task in privileged mode:

Task	Command
Clear the NetFlow statistics totals.	**clear mls statistics**

This example shows how to clear the NetFlow statistics totals:

```
Console> (enable) clear mls statistics
All mls statistics cleared.
Console> (enable)
```

Displaying the NetFlow Statistics Debug Information

The **show mls debug** command displays the NetFlow statistics debug information that you can send to your technical support representative for analysis if necessary.

To display the NetFlow statistics debug information, perform this task:

Task	Command
Display the NetFlow statistics debug information that you can send to your technical support representative.	**show mls debug**

Note The **show tech-support** command displays supervisor engine system information. Use application-specific commands to get more information about particular applications.

Configuring the MLS IP-Directed Broadcasts on the Switch

The IP-directed broadcasts are used primarily for ticker-type (stock quote) devices; however, when the feature is enabled on router interfaces, it provides a means to enable malicious denial-of-service attacks.

An IP-directed broadcast is a datagram that is sent to the broadcast address of a subnet to which the sending machine is not directly connected. The directed broadcast is routed through the network as a unicast packet until it arrives at the target subnet, where it is converted into a link layer broadcast. Due to the IP addressing architecture, only the last router in the chain, the one that is connected directly to the target subnet, can conclusively identify an IP-directed broadcast.

Before supervisor engine software release 7.2(2), the IP-directed broadcast traffic was handled by enabling the IP-directed broadcasts using the **ip directed-broadcast** command on the MSFC. The MSFC handled the traffic at the process level, which caused high CPU utilization.

With software release 7.2(2) and later releases, you can configure MSFC2 to handle the IP-directed broadcasts in the hardware using PFC2.

Note Cisco IOS Release 12.1(11b)E is required on MSFC2.

This example shows how to enable the IP-directed broadcasts:

```
Router(config-if)# mls ip directed-broadcast ?
  exclude-router  exclude router from recepient list for directed broadcast
  include-router  include router in recepient list for directed broadcast
```

The **exclude-router** option forwards the IP-directed broadcast packet in the hardware to all the hosts in the VLAN except the router.

The **include-router** option forwards the IP-directed broadcast packet in the hardware to all the hosts in the VLAN including the router. With this option, the router does not forward the IP-directed broadcast packet again.

The **no** form of the command is as follows:

```
Router(config-if)# no mls ip directed-broadcast [exclude-router | include-router]
```

The **no** form returns the interface configuration to the default mode. In the default mode, the IP-directed broadcast packets are not hardware forwarded. They are handled at the process level by MSFC2. The MSFC2 decision to forward or not forward the packet depends on the **ip directed-broadcast** command configuration.

There is no interaction between the **ip directed-broadcast** command and the **mls ip directed-broadcast** command. The **ip directed-broadcast** command involves software forwarding and the **mls ip directed-broadcast** command involves hardware forwarding.

CHAPTER 14

Configuring MLS

This chapter describes how to configure Multilayer Switching (MLS) for the Catalyst 6500 series switches. MLS provides IP and Internetwork Packet Exchange (IPX) unicast Layer 3 switching and IP multicast Layer 3 switching with Supervisor Engine 1, the Policy Feature Card (PFC), and the Multilayer Switch Feature Card (MSFC) or MSFC2.

Note For complete information on the syntax and usage information for the supervisor engine commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Note Supervisor Engine 720 with PFC3A/PFC3B/PFC3BXL and MSFC3 and Supervisor Engine 32 with PFC3B/PFC3BXL and MSFC2A provide Layer 3 switching with Cisco Express Forwarding for PFC3 (CEF for PFC3). See Chapter 13, "Configuring CEF for PFC2 and PFC3A," for more information.

Note Supervisor Engine 2, PFC2, and MSFC2 provide Layer 3 switching with Cisco Express Forwarding for PFC2 (CEF for PFC2). See Chapter 13, "Configuring CEF for PFC2 and PFC3A," for more information.

Understanding How Layer 3 Switching Works

Layer 3 switching allows the switch, instead of a router, to forward IP and IPX unicast traffic and IP multicast traffic between VLANs. Layer 3 switching is implemented in the hardware and provides wire-speed interVLAN forwarding on the switch, rather than on the MSFC. Layer 3 switching requires minimal support from the MSFC. The MSFC routes any traffic that cannot be Layer 3 switched.

Note Layer 3 switching supports the routing protocols that are configured on the MSFC. Layer 3 switching does not replace the routing protocols that are configured on the MSFC. Layer 3 switching uses IP Protocol Independent Multicast (IP PIM) for multicast route determination.

Layer 3 switching on Catalyst 6500 series switches provides traffic statistics that you can use to identify traffic characteristics for administration, planning, and troubleshooting. Layer 3 switching uses NetFlow Data Export (NDE) to export flow statistics (for more information about NDE, see Chapter 16, "Configuring NDE").

These sections describe Layer 3 switching and MLS on the Catalyst 6500 series switches:

Understanding Layer 3-Switched Packet Rewrite

When a packet is Layer 3 switched from a source in one VLAN to a destination in another VLAN, the switch performs a packet rewrite at the egress port based on information learned from the MSFC so that the packets appear to have been routed by the MSFC.

Note Rather than just forwarding multicast packets, the switch replicates them as necessary on the appropriate VLANs.

The packet rewrite alters these five fields:

- Layer 2 (MAC) destination address
- Layer 2 (MAC) source address
- Layer 3 IP Time to Live (TTL) or IPX Transport Control
- Layer 3 checksum
- Layer 2 (MAC) checksum (also called the frame checksum or FCS)

If Source A and Destination B are on different VLANs and Source A sends a packet to the MSFC to be routed to Destination B, the switch recognizes that the packet was sent to the Layer 2 (MAC) address of the MSFC.

To perform Layer 3 switching, the switch rewrites the Layer 2 frame header, changing the Layer 2 destination address to the Layer 2 address of Destination B and the Layer 2 source address to the Layer 2 address of the MSFC. The Layer 3 addresses remain the same.

In IP unicast and IP multicast traffic, the switch decrements the Layer 3 Time to Live (TTL) value by 1 and recomputes the Layer 3 packet checksum. In IPX traffic, the switch increments the Layer 3 Transport Control value by 1 and recomputes the Layer 3 packet checksum. The switch recomputes the Layer 2 frame checksum and forwards (or for multicast packets, replicates as necessary) the rewritten packet to Destination B's VLAN.

These sections describe how the packets are rewritten:

Understanding IP Unicast Rewrite

Received IP unicast packets are (conceptually) formatted as follows:

Layer 2 Frame Header		Layer 3 IP Header				Data	FCS
Destination	Source	Destination	Source	TTL	Checksum		
MSFC MAC	*Source A MAC*	*Destination B IP*	*Source A IP*	*n*	*calculation1*		

After the switch rewrites an IP unicast packet, it is (conceptually) formatted as follows:

Layer 2 Frame Header		Layer 3 IP Header				Data	FCS
Destination	Source	Destination	Source	TTL	Checksum		
Destination B MAC	*MSFC MAC*	*Destination B IP*	*Source A IP*	*n-1*	*calculation2*		

Understanding IPX Unicast Rewrite

Received IPX packets are (conceptually) formatted as follows:

Layer 2 Frame Header		Layer 3 IPX Header			Data	FCS
Destination	Source	Checksum/ IPX Length/ Transport Control	Destination Net/ Node/ Socket	Source Net/ Node/ Socket		
MSFC MAC	*Source A MAC*	*n*	*Destination B IPX*	*Source A IPX*		

After the switch rewrites an IPX packet, it is (conceptually) formatted as follows:

Layer 2 Frame Header		Layer 3 IPX Header			Data	FCS
Destination	Source	Checksum/ IPX Length/ Transport Control	Destination Net/ Node/ Socket	Source Net/ Node/ Socket		
Destination B MAC	*MSFC MAC*	*n+1*	*Destination B IPX*	*Source A IPX*		

Understanding IP Multicast Rewrite

Received IP multicast packets are (conceptually) formatted as follows:

Layer 2 Frame Header		Layer 3 IP Header				Data	FCS
Destination	Source	Destination	Source	TTL	Checksum		
Group G1 MAC[1]	*Source A MAC*	*Group G1 IP*	*Source A IP*	*n*	*calculation1*		

1. In this example, Destination B is a member of Group G1.

After the switch rewrites an IP multicast packet, it is (conceptually) formatted as follows:

Layer 2 Frame Header		Layer 3 IP Header				Data	FCS
Destination	Source	Destination	Source	TTL	Checksum		
Group G1 MAC	*MSFC MAC*	*Group G1 IP*	*Source A IP*	*n–1*	*calculation2*		

Understanding MLS

Note Supervisor Engine 1, PFC, and MSFC or MSFC2 can only do MLS internally with the MSFC or MSFC2 in the same chassis; an external MLS-RP cannot be used in place of the internal MLS-RP.

Supervisor Engine 1, PFC, and MSFC or MSFC2 provide Layer 3 switching with MLS. Layer 3 switching with MLS identifies flows on the switch after the first packet has been routed by the MSFC and transfers the process of forwarding the remaining traffic in the flow to the switch, which reduces the load on the MSFC.

These sections describe MLS:

Understanding MLS Flows

Layer 3 protocols, such as IP and IPX, are connectionless—they deliver every packet independently of every other packet. However, actual network traffic consists of many end-to-end conversations, or flows, between users or applications.

MLS supports unicast and multicast flows:

- A unicast flow can be any of the following:
  - All traffic to a particular destination
  - All traffic from a particular source to a particular destination
  - All traffic from a particular source to a particular destination that shares the same protocol and transport-layer information
- A multicast flow is all traffic with the same protocol and transport-layer information from a particular source to the members of a particular destination multicast group.

For example, communication from a client to a server and from the server to the client are separate flows. The Telnet traffic that is transferred from a particular source to a particular destination comprises a separate flow from File Transfer Protocol (FTP) packets between the same source and destination.

Note The PFC uses the Layer 2 multicast forwarding table to identify the ports to which Layer 2 multicast traffic should be forwarded (if any). The multicast forwarding table entries are populated by whichever multicast constraint feature is enabled on the switch (IGMP snooping or Generic Attribute Registration Protocol [GARP] Multicast Registration Protocol [GMRP]). These entries map the destination multicast MAC address to the outgoing switch ports for a given VLAN.

Understanding the MLS Cache

These sections describe the MLS cache:

MLS Cache

The PFC maintains a Layer 3 switching table called the MLS cache for the Layer 3-switched flows. The cache also includes the entries for the traffic statistics that are updated simultaneously with the switching of packets. After the PFC creates an MLS cache entry, the packets that are identified as belonging to an existing flow can be Layer 3 switched based on the cached information. The MLS cache maintains flow information for all the active flows.

Unicast Traffic

For unicast traffic, the PFC creates an MLS cache entry for the initial routed packet of each unicast flow. Upon receipt of a routed packet that does not match any unicast flow currently in the MLS cache, the PFC creates a new MLS entry.

Multicast Traffic

For multicast traffic, the PFC populates the MLS cache using information that is learned from the MSFC. Whenever the MSFC receives traffic for a new multicast flow, it updates its multicast routing table and forwards the new information to the PFC. In addition, if an entry in the multicast routing table ages out, the MSFC deletes the entry and forwards the updated information to the PFC.

For each multicast flow cache entry, the PFC maintains a list of outgoing interfaces for the destination IP multicast group. The PFC uses this list to identify the VLANs on which traffic to a given multicast flow should be replicated.

These Cisco IOS commands affect the multicast MLS cache entries on the switch:

- Using the **clear ip mroute** command to clear the multicast routing table on the MSFC clears all multicast MLS cache entries on the PFC.
- Using the **no ip multicast-routing** command to disable IP multicast routing on the MSFC purges all multicast MLS cache entries on the PFC.

MLS Cache Aging

The state and identity of flows are maintained while the packet traffic is active; when the traffic for a flow ceases, the entry ages out. You can configure the aging time for the MLS entries that are kept in the MLS cache. If an entry is not used for the specified period of time, the entry ages out and the statistics for that flow can be exported to a flow collector application.

MLS Cache Size

The maximum MLS cache size is 128,000 entries. The MLS cache is shared by all MLS processes on the switch (IP MLS, IP MMLS, and IPX MLS). An MLS cache that is larger than 32,000 entries increases the probability that a flow will not be Layer 3 switched but will be forwarded to the MSFC.

Understanding Flow Masks

The PFC uses flow masks to determine how the MLS entries are created.

These sections describe the flow mask modes:

- Flow Mask Modes—Prior to Software Release 8.5(1), page 14-6
- Flow Mask Modes—Software Release 8.5(1) and Later Releases, page 14-7
- Flow Mask Modes and show mls entry Command Outputs, page 14-9

Flow Mask Modes—Prior to Software Release 8.5(1)

The PFC supports only one flow mask (the most specific one) for all MSFCs that are Layer 3 switched by that PFC. If the PFC detects different flow masks from the different MSFCs for which it is performing Layer 3 switching, it changes its flow mask to the most specific flow mask detected.

When the PFC flow mask changes, the entire MLS cache is purged. When the PFC exports the cached entries, the flow records are created based on the current flow mask. Depending on the current flow mask, some fields in the flow record might not have values. The unsupported fields are filled with a zero (0).

The MLS flow masks are as follows:

- destination-ip—The least-specific flow mask. The PFC maintains one MLS entry for each Layer 3 destination address. All flows to a given Layer 3 destination address use this MLS entry.
- destination-ipx—The only flow mask mode for IPX MLS is destination mode. The PFC maintains one IPX MLS entry for each destination IPX address (network and node). All flows to a given destination IPX address use this IPX MLS entry.
- source-destination-ip—The PFC maintains one MLS entry for each source and destination IP address pair. All flows between a given source and destination use this MLS entry regardless of the IP protocol ports.
- source-destination-vlan—For IP MMLS. The PFC maintains one MMLS cache entry for each {source IP, destination group IP, source VLAN}. The multicast source-destination-vlan flow mask differs from the IP unicast MLS source-destination-ip flow mask in that, for IP MMLS, the source VLAN is included as part of the entry. The source VLAN is the multicast reverse path forwarding (RPF) interface for the multicast flow.
- full flow—The most-specific flow mask. The PFC creates and maintains a separate MLS cache entry for each IP flow. A full flow entry includes the source IP address, destination IP address, protocol, and protocol ports.

Flow Mask Modes—Software Release 8.5(1) and Later Releases

With software release 8.5(1) and later releases, the multiple flow mask feature is supported on Supervisor Engine 720. This feature results in some changes to the NetFlow Data Export (NDE) functionality.

A new value for the flow mask is **null**. Enter the **set mls flow null** command to set the flow mask to null (null is the new default flow mask). When the flow mask is set to **null** and no feature is driving a more specific flow mask, all the flows will match to the same null flow. The counters for the null flow are incremented each time a flow hits the null flow. When the flow masks are null and no other feature is driving a flow mask, when you enter the **show mls statistics entry** command, the command output displays the null flow as follows (in the example, flows are not being exported because NDE is disabled):

```
Console> (enable) show mls statistics entry
Flowmask set to Null. Please set the flowmask to see the flows
                                    Last    Used
Destination IP   Source IP       Prot  DstPrt SrcPrt Vlan  Stat-Pkts  Stat-Bytes

--------------- --------------- ----- ------ ------ ----- ---------  -----------
 -               -               -     -      -      N/A  728915    33530090

Console> (enable)
```

To enable flow creation, specify the flow mask by entering the **set mls flow** {**destination** | **destination-source** | **null** | **full**} command and specify a keyword other than the **null** keyword. If NDE is enabled when the **null** flow mask is configured, NDE will not export any flows. An example is as follows:

```
Console> (enable) set mls nde enable
Netflow export enabled
Console> (enable) 2005 Sep 18 18:04:43 %MLS-5-FLOWMASK_NULL:IP Flowmask set to Null:Flows
will not be exported
```

Conversely, if NDE is enabled and you set the flow mask to **null**, the following message displays:

```
Console> (enable) set mls flow null
2000 Sep 18 18:04:02 %MLS-5-FLOWMASK_NULL:IP Flowmask set to Null:Flows will not be
exported
Console> (enable)
```

When you upgrade from software release 8.4(x) to software release 8.5(1) and later releases, note the following:

- In binary configuration mode:
  - In software release 8.4(x), if you set the flow mask to "xxx" by entering the **set mls flow xxx** command (where xxx is *any* of the keyword options), after the upgrade, the flow mask is still **xxx**.
  - In software release 8.4(x), if you did not set the flow mask, after the upgrade, the flowmask would be **destination**.
- In text configuration mode:
  - In software release 8.4(x), if you set the flow mask to **destination** (the default) by entering the **set mls flow destination** command, after the upgrade, the flow mask would be **null** (the new default).
  - In software release 8.4(x), if you did not set the flow mask, after the upgrade, the flow mask would be **null**.
  - In software release 8.4(x), if you set the flow mask to any keyword option other than **destination**, after the upgrade, the flow mask would be the same flow mask that you configured.

Because multiple flow masks can now coexist on the switch, the **show mls statistics entry** command displays only the relevant fields per flow. Depending on the flow mask that is used to create a particular flow, the relevant fields are zeroed out. NDE is used by the flow collector software. NDE assumes that all flows are created with the same flow mask. Due to this restriction, NDE cannot be enabled with certain features requiring conflicting flow masks. One specific case is hardware-accelerated NAT. NDE and hardware-accelerated NAT are mutually exclusive.

Software release 8.5(1) introduces hardware acceleration for some MSFC features. When upgrading from software release 8.4(x) to software release 8.5(1), there are no issues with MSFC features that were already configured and running. In addition to NAT, such features as reflexive ACLs and Context Based Access Control (CBAC) can work in the hardware if there is no flow mask conflict. A feature will work in the hardware unless the feature needs a flow mask that is in conflict with another feature such as an NDE or QoS microflow policer.

Hardware acceleration is also introduced in software release 8.5(1) for WCCP and TCP intercept. These MSFC features can coexist with NDE if there is no flow mask conflict. The ACL manager attempts to merge the flow mask requirements of different features. The basic idea is to allocate a new flow mask only for a strict flow mask requirement that is incompatible with already allocated flow masks. NDE does not have a strict flow mask requirement, so the flow mask for NDE can be moved up.

To use the hardware acceleration functionality for NAT, if a flow mask has been configured for NDE (enter the **show mls** command to display flow masks), perform these steps:

Step 1 Enter the **set mls flow null** command.

Step 2 The MSFC needs to request a flow mask. This is accomplished by reconfiguring the specific MSFC feature.

NDE will fail if any of the following events occur:

- Hardware-accelerated NAT is enabled.
- Two or more features with conflicting flow masks have been configured on the switch.

Conversely, once NDE is successfully configured, NAT cannot be configured to work in the hardware and two different features with conflicting flow mask requirements cannot be configured on the switch.

Software release 8.5(1) introduces the **show mls flowmask** command that displays the flow masks used by the various features on the switch.

These examples show the output with various configurations when no features are configured on the MSFC:

```
Console> show mls flowmask
Netflow Data Export is enabled
NDE Flowmask is configured to use at least Null flowmask
Console>

Console> show mls flowmask
Netflow Data Export is enabled and is using Full flowmask
NDE Flowmask is configured to use at least Full flowmask
Console>

Console> show mls flowmask
Netflow Data Export is disabled
NDE Flowmask is configured to use at least Full flowmask
Console>
```

This example shows the output when NAT is configured on the MSFC:

```
Console> show mls flowmask
The MSFC features are using NotVlanFullFlow and VlanFullFlowOnly flow mask on vlan(s)
10-11,50-51,90-91.
Netflow Data Export is disabled
NDE Flowmask is configured to at least the Null flowmask
Console>
```

These examples show the output with various configurations when the reflexive ACL feature is configured on the MSFC:

```
Console> show mls flowmask
The MSFC features are using VlanFullFlowOnly flow mask on vlan(s) 13.
Netflow Data Export is disabled
NDE Flowmask is configured to use at least Null flowmask
Console>

Console> show mls flowmask
The MSFC features are using VlanFullFlowOnly flow mask on vlan(s) 13.
Netflow Data Export is enabled and is using Full-Vlan flowmask
NDE Flowmask is configured to use at least Full-Vlan flowmask
Console>
```

Flow Mask Modes and show mls entry Command Outputs

With the destination-ip flow mask, the source IP, protocol, and source and destination port fields show the details of the last packet that was Layer 3 switched using the MLS cache entry.

This example shows how the **show mls entry** command output appears in destination-ip mode:

```
Console> (enable) show mls entry ip short
Destination-IP  Source-IP       Prot  DstPrt SrcPrt Destination-Mac   Vlan
--------------- --------------- ----- ------ ------ ----------------- ----
 ESrc EDst SPort DPort Stat-Pkts Stat-Byte    Uptime   Age
 ---- ---- ----- ----- --------- ------------ -------- --------
171.69.200.234  -               -     -      -      00-60-70-6c-fc-22 4
 ARPA SNAP 5/8   11/1  3152      347854       09:01:19 09:08:20
171.69.1.133    -               -     -      -      00-60-70-6c-fc-23 2
 SNAP ARPA 5/8   1/1   2345      123456       09:03:32 09:08:12

Total Entries: 2
* indicates TCP flow has ended
Console> (enable)
```

Note The **short** keyword exists for some **show** commands and displays the output by wrapping the text after 80 characters. The default is **long** (no text wrap).

With the source-destination-ip flow mask, the protocol, source port, and destination port fields display the details of the last packet that was Layer 3 switched using the MLS cache entry.

This example shows how the **show mls entry** command output appears in source-destination-ip mode:

```
Console> (enable) show mls entry ip short
Destination-IP  Source-IP       Prot  DstPrt SrcPrt Destination-Mac   Vlan
--------------- --------------- ----- ------ ------ ----------------- ----
 ESrc EDst SPort DPort Stat-Pkts Stat-Byte    Uptime   Age
 ---- ---- ----- ----- --------- ------------ -------- --------
171.69.200.234  171.69.192.41   -     -      -      00-60-70-6c-fc-22 4
 ARPA SNAP 5/8   11/1  3152      347854       09:01:19 09:08:20
171.69.1.133    171.69.192.42   -     -      -      00-60-70-6c-fc-23 2
 SNAP ARPA 5/8   1/1   2345      123456       09:03:32 09:08:12

Total Entries: 2
* indicates TCP flow has ended
Console> (enable)
```

With the full-flow flow mask, because a separate MLS entry is created for every ip flow, the details are shown for each flow.

This example shows how the **show mls entry** command output appears in full flow mode:

```
Console> (enable) show mls entry ip short
Destination-IP  Source-IP       Prot  DstPrt SrcPrt Destination-Mac   Vlan
--------------- --------------- ----- ------ ------ ----------------- ----
 ESrc EDst SPort DPort Stat-Pkts Stat-Byte    Uptime   Age
 ---- ---- ----- ----- --------- ------------ -------- --------
171.69.200.234  171.69.192.41   TCP*  6000   59181  00-60-70-6c-fc-22 4
 ARPA SNAP 5/8   11/1  3152      347854       09:01:19 09:08:20
171.69.1.133    171.69.192.42   UDP   2049   41636  00-60-70-6c-fc-23 2
 SNAP ARPA 5/8   1/1   2345      123456       09:03:32 09:08:12

Total Entries: 2
* indicates TCP flow has ended
Console> (enable)
```

Partially and Completely Switched Multicast Flows

Some flows might be partially Layer 3 switched instead of completely Layer 3 switched in these situations:

- The MSFC is configured as a member of the IP multicast group (using the **ip igmp join-group** command) on the RPF interface of the multicast source.
- The MSFC is the first-hop router to the source in PIM sparse mode (in this case, the MSFC must send PIM-register messages to the rendezvous point).
- The multicast TTL threshold is configured on an egress interface for the flow.
- The extended access list deny condition on the RPF interface specifies anything other than the Layer 3 source, Layer 3 destination, or IP protocol (an example is a Layer 4 port number).
- The multicast helper is configured on the RPF interface for the flow, and multicast to broadcast translation is required.
- Multicast tag switching is configured on an egress interface.
- Network address translation (NAT) is configured on an interface, and source address translation is required for the outgoing interface.

For partially switched flows, all multicast traffic belonging to the flow reaches the MSFC and is software switched for any interface that is not Layer 3 switched.

The PFC prevents multicast traffic in flows that are completely Layer 3 switched from reaching the MSFC, reducing the load on the MSFC. The **show ip mroute** and **show mls ip multicast** commands identify completely Layer 3-switched flows with the text string "RPF-MFD." Multicast Fast Drop (MFD) indicates that from the perspective of the MSFC, the multicast packet is dropped because it is switched by the PFC.

For all completely Layer 3-switched flows, the PFC periodically sends the multicast packet and byte count statistics to the MSFC, because the MSFC cannot record the multicast statistics for completely switched flows, which it never sees. The MSFC uses the statistics to update the corresponding multicast routing table entries and to reset the appropriate expiration timers.

MLS Examples

Figure 14-1 shows a simple IP MLS network topology. In this example, Host A is on the Sales VLAN (IP subnet 171.59.1.0), Host B is on the Marketing VLAN (IP subnet 171.59.3.0), and Host C is on the Engineering VLAN (IP subnet 171.59.2.0).

When Host A initiates an HTTP file transfer to Host C, an MLS entry for this flow is created (this entry is the second item in the MLS cache shown in Figure 14-1). The PFC stores the MAC addresses of the MSFC and Host C in the MLS entry when the MSFC forwards the first packet from Host A through the switch to Host C. The PFC uses this information to rewrite the subsequent packets from Host A to Host C.

Figure 14-1 IP MLS Example Topology

Source IP Address	Destination IP Address	Application	Rewrite Src/Dst MAC Address	Destination VLAN
171.59.1.2	171.59.3.1	FTP	Dd:Bb	Marketing
171.59.1.2	171.59.2.2	HTTP	Dd:Cc	Engineering
171.59.2.2	171.59.1.2	HTTP	Dd:Aa	Sales

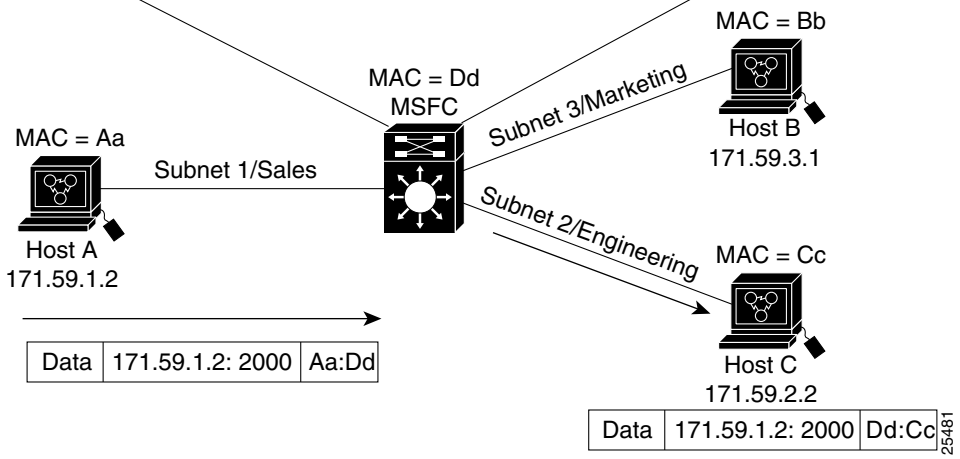
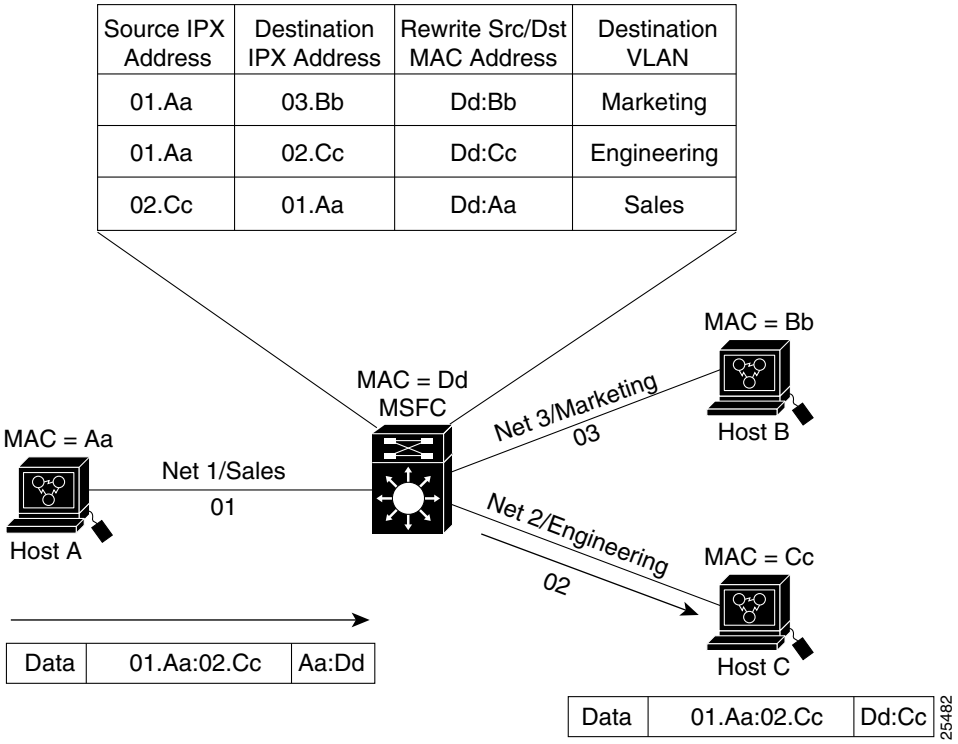


Figure 14-2 shows a simple IPX MLS network topology. In this example, Host A is on the Sales VLAN (IPX address 01.Aa), Host B is on the Marketing VLAN (IPX address 03.Bb), and Host C is on the Engineering VLAN (IPX address 02.Cc).

When Host A initiates a file transfer to Host B, an IPX MLS entry for this flow is created (this entry is the first item in the table shown in Figure 14-1). The PFC stores the MAC addresses of the MSFC and Host B in the IPX MLS entry when the MSFC forwards the first packet from Host A through the switch to Host B. The PFC uses this information to rewrite the subsequent packets from Host A to Host B.

Similarly, a separate IPX MLS entry is created in the MLS cache for the traffic from Host A to Host C, and for the traffic from Host C to Host A. The destination VLAN is stored as part of each IPX MLS entry so that the correct VLAN identifier is used when encapsulating traffic on the trunk links.

Figure 14-2 IPX MLS Example Topology



Default MLS Configuration

Table 14-1 shows the default IP MLS configuration.

Table 14-1 Default IP MLS Configuration

Feature	Default Value
IP MLS enable state	Enabled
IP MLS aging time	16 seconds
IP MLS fast aging time	0 seconds (no fast aging)
IP MLS fast aging-time packet threshold	0 packets

Table 14-2 shows the default IP MMLS switch configuration.

Table 14-2 Default IP MMLS Supervisor Engine Configuration

Feature	Default Value
Multicast services (IGMP snooping or GMRP)	Disabled
IP MMLS	Enabled

Table 14-3 shows the default IP MMLS MSFC configuration.

Table 14-3 Default IP MMLS MSFC Configuration

Feature	Default Value
Multicast routing	Disabled globally
IP PIM routing	Disabled on all interfaces
IP MMLS Threshold	Unconfigured—no default value
IP MMLS	Enabled when multicast routing is enabled and IP PIM is enabled on the interface

Table 14-4 shows the default IPX MLS configuration.

Table 14-4 Default IPX MLS Configuration

Feature	Default Value
IPX MLS enable state	Enabled
IPX MLS aging time	256 seconds

Configuration Guidelines and Restrictions

These sections describe the configuration guidelines and restrictions for IP MLS, IP MMLS, and IPX MLS:

IP MLS

These sections describe the IP MLS configuration guidelines:

Maximum Transmission Unit Size

The default maximum transmission unit (MTU) for IP MLS is 1500. To change the MTU on an IP MLS-enabled interface, enter the **ip mtu** *mtu* command.

Restrictions on Using IP Routing Commands with IP MLS Enabled

Enabling certain IP processes on an interface will affect IP MLS on the interface. Table 14-5 shows the affected commands and the resulting behavior.

Table 14-5 IP Routing Command Restrictions

Command	Behavior
clear ip route	Clears all MLS cache entries for all switches performing Layer 3 switching for this MSFC.
ip routing	The **no** form purges all MLS cache entries and disables IP MLS on this MSFC.
ip security (all forms of this command)	Disables IP MLS on the interface.
ip tcp compression-connections	Disables IP MLS on the interface.
ip tcp header-compression	Disables IP MLS on the interface.

IP MMLS

These sections describe the IP MMLS configuration guidelines:

- IP MMLS Supervisor Engine Guidelines and Restrictions, page 14-14
- IP MMLS MSFC Configuration Restrictions, page 14-15
- Unsupported IP MMLS Features, page 14-15

IP MMLS Supervisor Engine Guidelines and Restrictions

This section describes the guidelines and restrictions for configuring Supervisor Engine 1 for IP MMLS:

- Only the ARPA rewrites are supported for the IP multicast packets.
- The Subnetwork Address Protocol (SNAP) rewrites are not supported.
- You must enable one of the multicast services (IGMP snooping or GMRP) on the switch in order to use IP MMLS.
- The IP multicast flows are not multilayer switched if there is no entry in the Layer 2 multicast forwarding table (for example, if no Layer 2 multicast services are enabled or the forwarding table is full). Enter the **show multicast group** command to check for a Layer 2 entry for a particular IP multicast destination.
- If a Layer 2 entry is cleared, the corresponding Layer 3 flow information is purged.
- When using two MSFCs that have one or more interfaces in the same VLAN, the switch uses two reserved VLANs (VLANs 1012 and 1013) internally to forward the multicast flows properly.
- The MSFC will not act as an external router for a Catalyst 5000 family switch that has Layer 3-switching hardware.

IP MMLS MSFC Configuration Restrictions

IP MMLS does not perform multilayer switching for an IP multicast flow in these situations:

- For the IP multicast groups that fall into these ranges (where * is in the range 0–255):

  224.0.0.* through 239.0.0.*
  224.128.0.* through 239.128.0.*

  Note The groups in the 224.0.0.* range are reserved for routing the control packets and must be flooded to all the forwarding ports of the VLAN. These addresses map to the multicast MAC address range 01-00-5E-00-00-*xx* where *xx* is in the range 0–0xFF.

- For the IP PIM auto-RP multicast groups (IP multicast group addresses 224.0.1.39 and 224.0.1.40).

  Note In the systems with redundant MSFCs, the IP PIM interface configuration must be the same on both the active and redundant MSFCs.

- For the flows that are forwarded on the multicast-shared tree (that is, {*,G,*} forwarding) when the interface or group is running IP PIM sparse mode.
- If the shortest-path tree (SPT) bit for the flow is cleared when running IP PIM sparse mode for the interface or group.
- For the fragmented IP packets and packets with IP options. However, the packets in the flow that are not fragmented or that do not specify the IP options are multilayer switched.
- For the source traffic that is received on the tunnel interfaces (such as MBONE traffic).
- For any RPF interface with multicast tag switching enabled.

Unsupported IP MMLS Features

If you enable IP MMLS, the IP accounting for the interface will not reflect accurate values.

IPX MLS

These sections describe the configuration guidelines that apply when configuring IPX MLS:

- IPX MLS Interaction with Other Features, page 14-15
- IPX MLS and Maximum Transmission Unit Size, page 14-16

IPX MLS Interaction with Other Features

Other Cisco IOS software features affect IPX MLS as follows:

- IPX accounting—IPX accounting cannot be enabled on an IPX MLS-enabled interface.
- IPX EIGRP—To support MLS on EIGRP interfaces, you must set the Transport Control (TC) maximum to a value that is greater than the default (16). Enter the **ipx maximum-hop** *tc_value* global configuration command on the MSFC with the *tc_value* greater than 16.

IPX MLS and Maximum Transmission Unit Size

In IPX, the two end points of communication negotiate the maximum transmission unit (MTU) to be used. The MTU size is limited by the media type.

Configuring MLS

These sections describe how to configure MLS:

Configuring Unicast MLS on the MSFC

These sections describe how to configure MLS on the MSFC:

For information on configuring routing on the MSFC, see Chapter 12, "Configuring InterVLAN Routing." For information on configuring unicast Layer 3 switching on Supervisor Engine 1, see the "Configuring MLS on Supervisor Engine 1" section on page 14-19.

Note The MSFC can be specified as the MLS route processor (MLS-RP) for Catalyst 5000 family switches using MLS. Refer to the *Layer 3 Switching Configuration Guide—Catalyst 5000 Family, 2926G Series, 2926 Series Switches,* for MLS configuration procedures.

Disabling and Enabling Unicast MLS on an MSFC Interface

Unicast MLS for IP and IPX is enabled globally by default, but can be disabled and enabled on a specified interface.

To disable unicast IP or IPX MLS on a specific MSFC interface, perform one of these tasks:

Task	Command
Specify an MSFC interface.	Router(config)# **interface** *vlan-id*
Disable IP MLS on an MSFC interface.	Router(config-if)# **no mls ip**
Disable IPX MLS on an MSFC interface.	Router(config-if)# **no mls ipx**

This example shows how to disable IP MLS on an MSFC interface:

```
Router(config)# interface vlan 100
Router(config-if)# no mls ip
Router(config-if)#
```

This example shows how to disable IPX MLS on an MSFC interface:

```
Router(config)# interface vlan 100
Router(config-if)# no mls ipx
Router(config-if)#
```

Note Unicast MLS is enabled by default; you only need to enable (or reenable) it if you have previously disabled it.

To enable unicast IP or IPX MLS on a specific MSFC interface, perform this task:

	Task	Command
Step 1	Specify an MSFC interface.	Router(config)# **interface** *vlan-id*
Step 2	Enable IP or IPX MLS on an MSFC interface.	Router(config-if)# **mls ip** or Router(config-if)# **mls ipx**

This example shows how to enable IP MLS on an MSFC interface:

```
Router(config)# interface vlan 100
Router(config-if)# mls ip
Router(config-if)#
```

This example shows how to enable IPX MLS on an MSFC interface:

```
Router(config)# interface vlan 100
Router(config-if)# mls ipx
Router(config-if)#
```

Displaying MLS Information on the MSFC

The **show mls status** command displays the MLS details. To display the MLS information on the MSFC, perform this task:

Task	Command
Display the MLS status.	**show mls status**

This example shows how to display the MLS status on the MSFC:

```
Router# show mls status
MLS global configuration status:
global mls ip:                      enabled
global mls ipx:                     enabled
global mls ip multicast:            disabled
current ip flowmask for unicast:    destination only
current ipx flowmask for unicast:   destination only
Router#
```

Using Debug Commands on the MSFC

Table 14-6 describes the MLS-related debug commands that you can use to troubleshoot the MLS problems on the MSFC.

Table 14-6 MLS Debug Commands

Command	Description
[**no**] **debug l3-mgr events**	Displays the Layer 3 manager-related events.
[**no**] **debug l3-mgr packets**	Displays the Layer 3 manager packets.
[**no**] **debug l3-mgr global**	Displays the bug trace of IP global purge events.
[**no**] **debug l3-mgr all**	Turns on all the Layer 3 manager debugging messages.

Table 14-7 describes the MLS-related debug commands that you can use to troubleshoot the MLS problems when using the MSFC as an external router for a Catalyst 5000 family switch.

Table 14-7 MLS Debug Commands—External Router Function

Command	Description
[**no**] **debug mls ip**	Turns on the IP-related events for MLS including route purging and changes of access lists and flow masks.
[**no**] **debug mls ipx**	Turns on the IPX-related events for MLS including route purging and changes of access lists and flow masks.
[**no**] **debug mls rp**	Turns on the route processor-related events.
[**no**] **debug mls locator**	Identifies which switch is switching a particular flow by using MLS explorer packets.
[**no**] **debug mls all**	Turns on all the MLS debugging events.

Using Debug Commands on the SCP

Table 14-8 describes the Serial Control Protocol (SCP)-related debug commands to troubleshoot the SCP that runs over the Ethernet out-of-band channel (EOBC).

Table 14-8 SCP Debug Commands

Command	Description
[**no**] **debug scp async**	Displays the trace for asynchronous data in and out of the SCP system.
[**no**] **debug scp data**	Displays the packet data trace.
[**no**] **debug scp errors**	Displays the errors and warnings in the SCP.
[**no**] **debug scp packets**	Displays the packet data in and out of the SCP system.
[**no**] **debug scp timeouts**	Reports the timeouts.
[**no**] **debug scp all**	Turns on all the SCP debugging messages.

Configuring MLS on Supervisor Engine 1

MLS is enabled by default on Catalyst 6500 series switches. You only need to configure Supervisor Engine 1 in these circumstances:

- You want to change the MLS aging time
- You want to enable NDE

These sections describe how to configure MLS on Supervisor Engine 1:

For information on configuring the VLANs on the switch, see Chapter 11, "Configuring VLANs." For information on configuring MLS on the MSFC, see the "Configuring Unicast MLS on the MSFC" section on page 14-16.

Note When you disable IP or IPX MLS on the MSFC, IP or IPX MLS is automatically disabled on Supervisor Engine 1. All the existing protocol-specific MLS cache entries are purged. To disable MLS on the MSFC, see the "Disabling and Enabling Unicast MLS on an MSFC Interface" section on page 14-16.

Note If NDE is enabled and you disable MLS, you will lose the statistics for existing cache entries—they are not exported.

Specifying the MLS Aging-Time Value

The MLS aging time for each protocol (IP and IPX) applies to all protocol-specific MLS cache entries. Any MLS entry that has not been used for *agingtime* seconds is aged out. The default is 256 seconds.

You can configure the aging time in the range of 8 to 2032 seconds in 8-second increments. Any aging-time value that is not a multiple of 8 seconds is adjusted to the closest multiple of 8 seconds. For example, a value of 65 is adjusted to 64 and a value of 127 is adjusted to 128.

Note We recommend that you keep the size of the MLS cache below 32,000 entries. If the number of MLS entries exceeds 32,000, some flows are sent to the MSFC. To keep the size of the MLS cache down, for IP, enable IP MLS fast aging as described in the "Specifying IP MLS Long-Duration Aging Time, Fast Aging Time, and Packet Threshold Values" section on page 14-20.

To specify the MLS aging time for both IP and IPX, perform this task in privileged mode:

Task	Command
Specify the MLS aging time for MLS cache entries.	**set mls agingtime** [*agingtime*]

This example shows how to specify the MLS aging time:

```
Console> (enable) set mls agingtime 512
Multilayer switching agingtime IP and IPX set to 512
Console> (enable)
```

To specify the IP MLS aging time, perform this task in privileged mode:

Task	Command
Specify the IP MLS aging time for an MLS cache entry.	**set mls agingtime ip** [*agingtime*]

This example shows how to specify the IP MLS aging time:

```
Console> (enable) set mls agingtime ip 512
Multilayer switching aging time IP set to 512
Console> (enable)
```

To specify the IPX MLS aging time, perform this task in privileged mode:

Task	Command
Specify the IPX MLS aging time for an MLS cache entry.	**set mls agingtime ipx** [*agingtime*]

This example shows how to specify the IPX MLS aging time:

```
Console> (enable) set mls agingtime ipx 512
Multilayer switching aging time IPX set to 512
Console> (enable)
```

Specifying IP MLS Long-Duration Aging Time, Fast Aging Time, and Packet Threshold Values

Note IPX MLS does not use fast aging. IPX MLS only operates in destination-source and destination flow modes; therefore, the number of IPX MLS entries in the MLS table is low relative to the number of IP MLS entries in full-flow mode.

To keep the MLS cache size below 32,000 entries, enable IP MLS fast aging time. The IP MLS fast aging time applies to the MLS entries that have no more than *pkt_threshold* packets that are switched within *fastagingtime* seconds after they are created. A typical cache entry that is removed is the entry for flows to and from a Domain Name Server (DNS) or TFTP server; the entry might never be used again after it is created. Detecting and aging out these entries saves space in the MLS cache for the other data traffic.

The default *fastagingtime* value is 0 (no fast aging). You can configure the *fastagingtime* value to 32, 64, 96, or 128 seconds. Any *fastagingtime* value that is not configured exactly as the indicated values is adjusted to the closest one. You can configure the *pkt_threshold* value to 0, 1, 3, 7, 15, 31, or 63 packets.

If you need to enable IP MLS fast aging time, initially set the value to 128 seconds. If the size of the MLS cache continues to grow over 32,000 entries, decrease the setting until the cache size stays below 32,000. If the cache continues to grow over 32,000 entries, decrease the normal IP MLS aging time.

The typical values for *fastagingtime* and *pkt_threshold* are 32 seconds and 0 packets (no packets are switched within 32 seconds after the entry is created).

To specify the IP MLS fast aging time and packet threshold, perform this task in privileged mode:

Task	Command
Specify the IP MLS fast aging time and packet threshold for an MLS cache entry.	**set mls agingtime fast** [*fastagingtime*] [*pkt_threshold*]

This example shows how to set the IP MLS fast aging time to 32 seconds with a packet threshold of 0 packets:

```
Console> (enable) set mls agingtime fast 32 0
Multilayer switching fast aging time set to 32 seconds for entries with no more than 0
packets switched.
Console> (enable)
```

To specify that an active flow gets aged out, perform this task in privileged mode:

Task	Command
Specify that an active flow gets aged out.	**set mls agingtime long-duration** *agingtime*

This example shows how to force an active flow to age out. You can specify the aging time of the active flow in the range of 64 to 1920 seconds in increments of 64.

```
Console> (enable) set mls agingtime long-duration 128
Multilayer switching agingtime set to 128 seconds for long duration flows
Console> (enable)
```

Setting the Minimum IP MLS Flow Mask

You can set the minimum granularity of the flow mask for the MLS cache on the PFC. The actual flow mask that is used will be at least of the granularity that is specified by this command. For information on how the different flow masks work, see the "Understanding Flow Masks" section on page 14-6.

For example, if you do not configure the access lists on any MSFC, then the IP MLS flow mask on the PFC is destination-ip by default. However, you can force the PFC to use the source-destination-ip flow mask by setting the minimum IP MLS flow mask using the **set mls flow destination-source** command.

Caution The **set mls flow destination-source** command purges all the existing shortcuts in the MLS cache and affects the number of the active shortcuts on the PFC. Exercise care when using this command.

To set the minimum IP MLS flow mask, perform this task in privileged mode:

Task	Command
Set the minimum IP MLS flow mask.	**set mls flow** {**destination** | **destination-source** | **null** | **full**}

This example shows how to set the minimum IP MLS flow mask to destination-source-ip:

```
Console> (enable) set mls flow destination-source
Configured IP flow mask is set to destination-source flow.
Console> (enable)
```

Displaying CAM Entries on the Supervisor Engine

The **show cam** command displays the content-addressable memory (CAM) entries that are associated with a specific MAC address. If the MAC address belongs to an MSFC, an "R" is appended to the MAC address.

If you specify a VLAN number, only those CAM entries that correspond to that VLAN number are displayed. If a VLAN is not specified, the entries for all the VLANs are displayed.

To display the CAM entries, perform this task:

Task	Command
Display the CAM entries by MAC address.	**show cam msfc** [*vlan*]

This example shows how to display the CAM entries:

```
Console> show cam msfc
VLAN  Destination MAC      Destination-Ports or VCs        Xtag  Status
----  ------------------   ------------------------------  ----  ------
194   00-e0-f9-d1-2c-00R   7/1                             2     H
193   00-00-0c-07-ac-c1R   7/1                             2     H
193   00-00-0c-07-ac-5dR   7/1                             2     H
202   00-00-0c-07-ac-caR   7/1                             2     H
204   00-e0-f9-d1-2c-00R   7/1                             2     H
195   00-e0-f9-d1-2c-00R   7/1                             2     H
192   00-00-0c-07-ac-c0R   7/1                             2     H
192   00-e0-f9-d1-2c-00R   7/1                             2     H
204   00-00-0c-07-ac-ccR   7/1                             2     H
202   00-e0-f9-d1-2c-00R   7/1                             2     H
194   00-00-0c-07-ac-5eR   7/1                             2     H
196   00-e0-f9-d1-2c-00R   7/1                             2     H
194   00-00-0c-07-ac-c2R   7/1                             2     H
193   00-e0-f9-d1-2c-00R   7/1                             2     H
Total Matching CAM Entries Displayed = 14
Console>
```

This example shows how to display the CAM entries for a specified VLAN:

```
Console> show cam msfc 192
VLAN  Destination MAC      Destination-Ports or VCs        Xtag  Status
----  ------------------   ------------------------------  ----  ------
192   00-00-0c-07-ac-c0R   7/1                             2     H
192   00-e0-f9-d1-2c-00R   7/1                             2     H
Console>
```

Displaying MLS Information

The **show mls** command displays protocol-specific MLS information and MSFC-specific information.

To display protocol-specific MLS information and MSFC-specific information, perform this task:

Task	Command
Display general IP or IPX MLS information and MSFC-specific information for all MSFCs.	**show mls** {**ip** | **ipx**} [*mod*[1]]

1. The **mod** keyword specifies the module number of the MSFC; either 15 (if the MSFC is installed on Supervisor Engine 1 in slot 1) or 16 (if the MSFC is installed on Supervisor Engine 1 in slot 2).

This example shows how to display IP MLS information and MSFC-specific information:

```
Console> (enable) show mls ip
Total Active MLS entries = 0
Total packets switched = 0
IP Multilayer switching enabled
IP Multilayer switching aging time = 256 seconds
IP Multilayer switching fast aging time = 0 seconds, packet threshold = 0
IP Flow mask: Full Flow
Configured flow mask is Destination flow
Active IP MLS entries = 0
Netflow Data Export version: 8
Netflow Data Export disabled
Netflow Data Export port/host is not configured
Total packets exported = 0

MSFC ID         Module XTAG MAC               Vlans
--------------- ------ ---- ------------------ --------------------
52.0.03         15     1    01-10-29-8a-0c-00 1,10,123,434,121
                                              222,666,959

Console> (enable)
```

This example shows how to display IPX MLS information:

```
Console> (enable) show mls ipx
IPX Multilayer switching aging time = 256 seconds
IPX flow mask is Destination flow
IPX max hop is 15
Active IPX MLS entries = 356

IPX MSFC ID     Module XTAG MAC               Vlans
--------------- ------ ---- ----------------- ----------------
22.1.0.56       15     1    00-10-07-38-29-18 2,3,4,5,6,
                                              7,8,9,10,11,
                                              12,13,14,15,16,
                                              17,18,19,20,66,
                                              77
                          00-d0-d3-9c-e3-f4 25
                          00-10-07-38-29-18 26,111
                          00-d0-d3-9c-e3-f4 112

22.1.0.58       16     2    00-10-07-38-22-22 2,3,4,5,6,
                                              7,8,9,10,11,
                                              12,13,14,15,16,
                                              17,18,19,20
                          00-d0-d3-33-17-8c 25
                          00-10-07-38-22-22 26,66,77,88,99,
                                              111
                          00-d0-d3-33-17-8c 112

Console> (enable)
```

Displaying IP MLS Cache Entries

These sections describe how to display the MLS cache entries on Supervisor Engine 1:

- Displaying All MLS Entries, page 14-25
- Displaying MLS Entries for a Specific IP Destination Address, page 14-25
- Displaying IPX MLS Entries for a Specific IPX Destination Address, page 14-26
- Displaying MLS Entries for a Specific IP Source Address, page 14-26
- Displaying MLS Entries for a Specific IP Flow, page 14-27
- Displaying IPX MLS Entries for a Specific MSFC, page 14-27
- Displaying MLS Entries for Bridged-Flow Statistics, page 14-28

Note For a description of how the flow mask mode affects the screen displays when showing the MLS entries, see the "Flow Mask Modes and show mls entry Command Outputs" section on page 14-9.

Displaying All MLS Entries

To display all the MLS entries (IP and IPX), perform this task in privileged mode:

Task	Command
Display all the MLS entries.	**show mls entry** [**short** | **long**]

This example shows how to display all the MLS entries (IP and IPX):

```
Console> (enable) show mls entry short
Destination-IP  Source-IP       Prot  DstPrt SrcPrt Destination-Mac  Vlan
--------------- --------------- ----- ------ ------ ----------------- ----
 ESrc EDst SPort DPort Stat-Pkts  Stat-Bytes   Created  LastUsed
 ---- ---- ----- ----- ---------- ------------ -------- --------
171.69.200.234  171.69.192.41   TCP*  6000   59181  00-60-70-6c-fc-22 4
 ARPA SNAP 5/8   11/1  3152       347854       09:01:19 09:08:20
171.69.1.133    171.69.192.42   UDP   2049   41636  00-60-70-6c-fc-23 2
 SNAP ARPA 5/8   1/1   2345       1234567      09:03:32 09:08:12
171.69.1.133    171.69.192.42   UDP   2049   41636  00-60-70-6c-fc-23 2
 SNAP ARPA 5/8   1/1   2345       1234567      09:03:32 09:08:12
171.69.1.133    171.69.192.42   UDP   2049   41636  00-60-70-6c-fc-23 2
 SNAP ARPA 5/8   1/1   2345       1234567      09:03:32 09:08:12
171.69.1.133    171.69.192.42   UDP   2049   41636  00-60-70-6c-fc-23 2
 SNAP ARPA 5/8   1/1   2345       1234567      09:03:32 09:08:12

Total IP entries: 5
* indicates TCP flow has ended.

Destination-IPX           Source-IPX-net Destination-Mac   Vlan Port
 Stat-Pkts Stat-Bytes
------------------------- -------------- ----------------- ---- -----
 --------- -----------
BABE.0000.0000.0001       -              00-a0-c9-0a-89-1d 211  13/37
 30230     1510775
201.00A0.2451.7423        -              00-a0-24-51-74-23 201  14/33
 30256     31795084
501.0000.3100.0501        -              31-00-05-01-00-00 501  9/37
 12121     323232
401.0000.0000.0401        -              00-00-04-01-00-00 401  3/1
 4633      38676

Total IPX entries: 4
Console>
```

Displaying MLS Entries for a Specific IP Destination Address

To display the MLS entries for a specific destination IP address, perform this task in privileged mode:

Task	Command
Display the MLS entries for the specified destination IP address.	**show mls entry ip destination** [*ip_addr*]

This example shows how to display the MLS entries for a specific destination IP address:

```
Console> (enable) show mls entry ip destination 172.20.22.14/24
Destination-IP  Source-IP       Prot  DstPrt SrcPrt Destination-Mac   Vlan
 EDst ESrc DPort  SPort  Stat-Pkts  Stat-Bytes  Uptime   Age
--------------- --------------- ----- ------ ------ ----------------- ----
 ---- ---- ------ ------ ---------- ----------- -------- --------
MSFC 172.20.25.1 (Module 15):
172.20.22.14    -               -     -      -      00-60-70-6c-fc-22 4
 ARPA ARPA 5/39   5/40   115        5290        00:12:20 00:00:04
MSFC 172.20.27.1 (Module 16):

Total entries:1
Console> (enable)
```

Displaying IPX MLS Entries for a Specific IPX Destination Address

To display the IPX MLS entries for a specific destination IPX address, perform this task in privileged mode:

Task	Command
Display the IPX MLS entries for a specific destination IPX address (net_address.node_address).	**show mls entry ipx destination** *ipx_addr*

This example shows how to display the IPX MLS entries for a specific destination IPX address:

```
Console> (enable) show mls entry ipx destination 3E.0010.298a.0c00
Destination IPX           Source IPX net Destination Mac   Vlan Port
------------------------- -------------- ----------------- ---- -----
MSFC 22.1.0.56 (Module 15):
3E.0010.298a.0c00                    13 00-00-00-00-00-09 26   4/7

Console> (enable)
```

Displaying MLS Entries for a Specific IP Source Address

To display the MLS entries for a specific source IP address, perform this task in privileged mode:

Task	Command
Display the MLS entries for the specified source IP address.	**show mls entry ip source** [*ip_addr*]

This example shows how to display the MLS entries for a specific source IP address:

```
Console> (enable) show mls entry ip source 10.0.2.15
Destination-IP  Source-IP       Prot  DstPrt SrcPrt Destination-Mac   Vlan
 EDst ESrc DPort  SPort  Stat-Pkts  Stat-Bytes  Uptime   Age
--------------- --------------- ----- ------ ------ ----------------- ----
 ---- ---- ------ ------ ---------- ----------- -------- --------
MSFC 172.20.25.1 (Module 15):
172.20.22.14    10.0.2.15       TCP   Telnet 37819  00-e0-4f-15-49-ff 51
 ARPA ARPA 5/39   5/40   115        5290        00:12:20 00:00:04
MSFC 172.20.27.1 (Module 16):
Total entries:1
Console> (enable)
```

Displaying MLS Entries for a Specific IP Flow

The **show mls entry ip flow** command displays the MLS entries for a specific IP flow. The *protocol* argument can be **tcp**, **udp**, **icmp**, or a decimal number for other protocol families. The *src_port* and *dst_port* arguments specify the protocol ports if the protocol is TCP or User Datagram Protocol (UDP). A value of zero (0) for *src_port, dst_port,* or *protocol* is treated as a wildcard and all entries are displayed (unspecified options are treated as wildcards). If the protocol selected is not TCP or UDP, set the *src_port* and *dst_prt* to 0 or no flows will display.

To display the MLS entries for a specific IP flow (when the flow mask mode is full flow), perform this task in privileged mode:

Task	Command
Display the MLS entries for a specific IP flow (when the flow mask mode is full flow).	**show mls entry ip flow** [*protocol src_port dst_port*]

This example shows how to display the MLS entries for a specific IP flow:

```
Console> (enable) show mls entry ip flow tcp 23 37819
Destination IP  Source IP       Port DstPrt SrcPrt Destination Mac   Vlan Port
--------------- --------------- ---- ------ ------ ----------------- ---- -----
MSFC 51.0.0.3:
10.0.2.15       51.0.0.2        TCP  37819  Telnet 08-00-20-7a-07-75 10   3/1
Console> (enable)
```

Displaying IPX MLS Entries for a Specific MSFC

To display the IPX MLS entries for a specific MSFC, perform this task in privileged mode:

Task	Command
Display the IPX MLS entries for a specific MSFC.	**show mls entry ipx** *mod*[1]

1. The **mod** keyword specifies the module number of the MSFC; either 15 (if the MSFC is installed on Supervisor Engine 1 in slot 1) or 16 (if the MSFC is installed on Supervisor Engine 1 in slot 2).

This example shows how to display the IPX MLS entries for a specific MSFC:

```
Console> (enable) show mls entry ipx 15
Destination-IPX           Destination-Mac   Vlan EDst ESrc  Port  Stat-Pkts
 Stat-Bytes  Uptime   Age
------------------------- ----------------- ---- ---- ---- ----- ----------
 ----------- -------- --------
MSFC 22.1.0.56 (Module 15):
11.0000.0000.2B10         00-00-00-00-2b-10 11   ARPA ARPA  -     7869
 361974      00:15:52 00:00:00
11.0000.0000.A810         00-00-00-00-a8-10 11   ARPA ARPA  -     3934
 180964      00:15:52 00:00:00
11.0000.0000.3210         00-00-00-00-32-10 11   ARPA ARPA  -     7871
 362066      00:15:52 00:00:00
11.0000.0000.B110         00-00-00-00-b1-10 11   ARPA ARPA  -     3935
 181010      00:15:52 00:00:00
11.0000.0000.1910         00-00-00-00-19-10 11   ARPA ARPA  -     7873
 362158      00:15:52 00:00:00
11.0000.0000.9A10         00-00-00-00-9a-10 11   ARPA ARPA  -     3936
 181056      00:15:52 00:00:00
```

```
11.0000.0000.0010         00-00-00-00-00-10 11   ARPA ARPA  3/11  7875
 362250      00:15:52 00:00:00
11.0000.0000.8310         00-00-00-00-83-10 11   ARPA ARPA  -     3937
 181102      00:15:52 00:00:00
10.0000.0000.0109         00-00-00-00-01-09 10   ARPA ARPA  3/10  96364
 4432744     00:15:52 00:00:00
11.0000.0000.4F10         00-00-00-00-4f-10 11   ARPA ARPA  -     7877
 362342      00:15:53 00:00:00
11.0000.0000.CC10         00-00-00-00-cc-10 11   ARPA ARPA  -     3938
 181148      00:15:53 00:00:00
11.0000.0000.5610         00-00-00-00-56-10 11   ARPA ARPA  -     7879
 362434      00:15:53 00:00:00
11.0000.0000.D510         00-00-00-00-d5-10 11   ARPA ARPA  -     3939
 181194      00:15:53 00:00:00
11.0000.0000.7D10         00-00-00-00-7d-10 11   ARPA ARPA  -     3940
 181240      00:15:53 00:00:00
11.0000.0000.FE10         00-00-00-00-fe-10 11   ARPA ARPA  -     3941
 181286      00:15:53 00:00:00
11.0000.0000.6410         00-00-00-00-64-10 11   ARPA ARPA  -     7883
 362618      00:15:53 00:00:00
11.0000.0000.E710         00-00-00-00-e7-10 11   ARPA ARPA  -     3941
 181286      00:15:53 00:00:00
11.0000.0000.6010         00-00-00-00-60-10 11   ARPA ARPA  -     7885
 362710      00:15:53 00:00:00
11.0000.0000.E310         00-00-00-00-e3-10 11   ARPA ARPA  -     3942
 181332      00:15:53 00:00:00
11.0000.0000.7910         00-00-00-00-79-10 11   ARPA ARPA  -     3943
 181378      00:15:54 00:00:00

Console> (enable)
```

Displaying MLS Entries for Bridged-Flow Statistics

To display the MLS entries for the bridged-flow statistics, perform this task in privileged mode:

Task	Command
Display the MLS entries for the bridged-flow statistics.	**show mls entry**

This example shows how to display the MLS entries for the bridged-flow statistics:

```
Console> (enable) show mls entry
     Destination-IP  Source-IP       Prot  DstPrt SrcPrt Destination-Mac   Vlan EDst ESrc
DPort     SPort     Stat-Pkts  Stat-Bytes  Uptime   Age
     --------------- ---------- ---------- ----- ------ ------ ----------------- ---- ---- ----
--------- --------- ---------- ---------- -------- --------
     224.0.0.5       21.2.0.22       -     0      0      00-00-00-00-00-00 0    ARPA ARPA
-         5/11      20         1280        00:03:14 00:00:04
     224.0.0.13      1.1.1.2         -     0      0      00-00-00-00-00-00 0    ARPA ARPA
-         5/11      7          210         00:03:02 00:00:02
     255.255.255.255 -               -     0      0      ff-ff-ff-ff-ff-ff 21   ARPA ARPA
-         5/11      28         2996        00:03:10 00:00:02
     10.6.62.195     -               -     0      0      00-00-00-00-00-02 20   ARPA ARPA
-         5/5       291494     13408724    00:03:16 00:00:00

     Destination-IPX           Destination-Mac   Vlan EDst ESrc  Port  Stat-Pkts
Stat-Bytes  Uptime   Age
     ------------------------- ----------------- ---- ---- ---- ----- -----------
----------- -------- --------
Total entries displayed:2
```

Clearing MLS Cache Entries

The **clear mls entry** command removes specific MLS cache entries. The **all** keyword clears all the MLS entries. The **destination** and **source** keywords specify the source and destination IP addresses. The destination and source *ip_addr_spec* can be a full IP address or a subnet address in the format *ip_subnet_addr*, *ip_addr/subnet_mask*, or *ip_addr/subnet_mask_bits*.

The **flow** keyword specifies the following additional flow information:

- Protocol family (*protocol*)—Specify **tcp**, **udp**, **icmp**, or a decimal number for other protocol families. A value of zero (0) for *protocol* is treated as a wildcard, and the entries for all protocols are cleared (the unspecified options are treated as wildcards).
- TCP or UDP source and destination port numbers (*src_port* and *dst_port*)—If the protocol you specify is TCP or UDP, specify the source and destination TCP or UDP port numbers. A value of zero (0) for *src_port* or *dst_port* is treated as a wildcard, and the entries for all source or destination ports are cleared (the unspecified options are treated as wildcards). For other protocols, set the *src_port* and *dst_port* to 0, or no entries will clear.

To clear an MLS entry, perform this task in privileged mode:

Task	Command
Clear an MLS entry.	**clear mls entry ip** [**destination** *ip_addr_spec*] [**source** *ip_addr_spec*] [**flow** *protocol src_port dst_port*] [**all**]

This example shows how to clear the MLS entries with destination IP address 172.20.26.22:

```
Console> (enable) clear mls entry ip destination 172.20.26.22
MLS IP entry cleared
Console> (enable)
```

This example shows how to clear the MLS entries with destination IP address 172.20.22.113, TCP source port 1652, and TCP destination port 23:

```
Console> (enable) clear mls entry destination 172.20.26.22 source 172.20.22.113 flow tcp
1652 23
MLS IP entry cleared
Console> (enable)
```

Clearing IPX MLS Cache Entries

The **clear mls entry ipx** command removes specific IPX MLS cache entries. The **destination** and **source** keywords specify the source and destination IPX addresses. The **all** keyword clears all the MLS entries.

Displaying IP MLS Statistics

These sections describe how to display the IP MLS statistics:

- Displaying IP MLS Statistics by Protocol, page 14-30
- Displaying Statistics for MLS Cache Entries, page 14-30

Displaying IP MLS Statistics by Protocol

The **show mls statistics protocol** command displays the IP MLS statistics by protocol (such as Telnet, FTP, and WWW). The **protocol** keyword functions only if the flow mask mode is full flow. Enter the **show mls** command to see the current flow mask.

To display the IP MLS statistics by protocol, perform this task in privileged mode:

Task	Command
Display the IP MLS statistics by protocol (only if IP MLS is in full flow mode).	**show mls statistics protocol**

This example shows how to display the IP MLS statistics by protocol:

```
Console> (enable) show mls statistics protocol
Protocol  TotalFlows  TotalPackets    Total Bytes
-------   ----------  --------------  ------------
Telnet    900         630             4298
FTP       688         2190            3105
WWW       389         42679           623686
SMTP      802         4966            92873
X         142         2487            36870
DNS       1580        52              1046
Others    82          1               73
Total     6583        53005           801951
Console> (enable)
```

Displaying Statistics for MLS Cache Entries

The **show mls statistics entry** command displays the IP MLS statistics for the MLS cache entries. Specify the destination IP address, source IP address, protocol, and source and destination ports to see the specific MLS cache entries.

A value of zero (0) for *src_port* or *dst_port* is treated as a wildcard, and all the statistics are displayed (the unspecified options are treated as wildcards). If the protocol specified is not TCP or UDP, set the *src_port* and *dst_prt* to 0 or no statistics will display.

To display the statistics for the MLS cache entries, perform this task in privileged mode:

Task	Command
Display the statistics for the MLS cache entries. If you do not specify an MLS cache entry, all the statistics are shown.	**show mls statistics entry ip** [**destination** *ip_addr_spec*] [**source** *ip_addr_spec*] [**flow** *protocol src_port dst_port*]

This example shows how to display the statistics for a particular MLS cache entry:

```
Console> show mls statistics entry ip destination 172.20.22.14
                                    Last    Used
Destination IP  Source IP       Prot DstPrt SrcPrt Stat-Pkts Stat-Bytes
--------------- --------------- ---- ------ ------ --------- -----------
MSFC 127.0.0.12:
172.20.22.14    172.20.25.10    6    50648  80     3152       347854
Console>
```

Clearing MLS Statistics

The **clear mls statistics** command clears the following statistics:

- Total packets that are switched (IP and IPX)
- Total packets that are exported (for NDE)

To clear the IP MLS statistics, perform this task in privileged mode:

Task	Command
Clear the IP MLS statistics.	**clear mls statistics**

This example shows how to clear the IP MLS statistics:

```
Console> (enable) clear mls statistics
All mls statistics cleared.
Console> (enable)
```

Displaying MLS Debug Information

The **show mls debug** command displays MLS debug information that you can send to your technical support representative for analysis if necessary.

To display the MLS debug information, perform this task:

Task	Command
Display the MLS debug information that you can send to your technical support representative.	**show mls debug**

Note The **show tech-support** command displays supervisor engine system information. Use the application-specific commands to get more information about particular applications.

Configuring IP MMLS

These sections describe how to configure IP MMLS:

- Configuring IP MMLS on the MSFC, page 14-32
- Displaying Global IP MMLS Information on the Supervisor Engine, page 14-37

Configuring IP MMLS on the MSFC

These sections describe how to configure the MSFC for IP MMLS:

Note For information on configuring routing on the MSFC, see Chapter 12, "Configuring InterVLAN Routing."

Note You can specify the MSFC as the MLS route processor (MLS-RP) for Catalyst 5000 family switches using MLS. Refer to the *Layer 3 Switching Configuration Guide—Catalyst 5000 Family, 2926G Series, 2926 Series Switches* for the Catalyst 5000 family switch MLS configuration procedures.

Note This section describes how to enable IP multicast routing on the MSFC. For more detailed IP multicast configuration information, refer to the "IP Multicast" section of the *Cisco IOS IP and IP Routing Configuration Guide* at this URL:
http://www.cisco.com/en/US/docs/ios/12_1/iproute/configuration/guide/ip_c.html

Enabling IP Multicast Routing Globally

You must enable IP multicast routing globally on the MSFC before you can enable IP MMLS on the MSFC interfaces.

To enable IP multicast routing globally on the MSFC, perform this task in global configuration mode:

Task	Command
Enable IP multicast routing globally.	Router(config)# **ip multicast-routing**

This example shows how to enable IP multicast routing globally:

```
Router(config)# ip multicast-routing
Router(config)#
```

Enabling IP PIM on MSFC Interfaces

You must enable IP PIM on the MSFC interfaces before IP MMLS will function on those interfaces.

To enable IP PIM on an interface, perform this task:

Task	Command
Enable IP PIM on an MSFC interface.	Router(config-if)# **ip pim** {**dense-mode** | **sparse-mode** | **sparse-dense-mode**}

This example shows how to enable IP PIM on an interface using the default mode (**sparse-dense-mode**):

```
Router(config-if)# ip pim
Router(config-if)#
```

This example shows how to enable IP PIM sparse mode on an interface:

```
Router(config-if)# ip pim sparse-mode
Router(config-if)#
```

Configuring the IP MMLS Global Threshold

You can configure a global multicast rate threshold, which is specified in packets per second, below which all (S,G) multicast traffic is routed by the MSFC. Entering the command prevents the creation of MLS entries for short-lived multicast flows (such as join requests).

Note This command does not affect the flows that are already being routed. To apply the threshold to existing routes, clear the route and let it reestablish.

To configure the IP MMLS threshold, perform this task:

Task	Command
Configure the IP MMLS threshold.	Router(config)# [**no**] **mls ip multicast threshold** *ppsec*

This example shows how to configure the IP MMLS threshold to 10 packets per second:

```
Router(config)# mls ip multicast threshold 10
Router(config)#
```

Use the **no** keyword to deconfigure the threshold.

Enabling IP MMLS on MSFC Interfaces

IP MMLS is enabled by default on the MSFC interface when you enable IP PIM on the interface. Perform this task only if you disabled IP MMLS on the interface and you want to reenable it.

Note You must enable IP PIM on all the participating MSFC interfaces before IP MMLS will function. For information on configuring IP PIM on the MSFC interfaces, see the "Enabling IP PIM on MSFC Interfaces" section on page 14-33.

To enable IP MMLS on an MSFC interface, perform this task:

Task	Command
Enable IP MMLS on an MSFC interface.	Router(config-if)# [**no**] **mls ip multicast**

This example shows how to enable IP MMLS on an MSFC interface:

```
Router(config-if)# mls ip multicast
Router(config-if)#
```

Use the **no** keyword to disable IP MMLS on an MSFC interface.

Displaying IP MMLS Interface Information

The **show ip pim interface count** command displays the IP MMLS enable state on the MSFC IP PIM interfaces and the number of packets that are received and sent on the interface.

The **show ip interface** command displays the IP MMLS enable state on an MSFC interface.

To display the IP MMLS information for an IP PIM MSFC interface, perform one of these tasks:

Task	Command
Display IP MMLS interface information.	Router# **show ip pim interface** [*type number*] **count**
Display the IP MMLS interface enable state.	Router# **show ip interface**

Displaying the IP Multicast Routing Table

The **show ip mroute** command displays the IP multicast routing table on the MSFC.

To display the IP multicast routing table, perform this task:

Task	Command
Display the IP multicast routing table.	Router# **show ip mroute** [*group*[*source*]] | [**summary**] | [**count**] | [**active** *kbps*]

This example shows how to display the IP multicast routing table for 239.252.1.1:

```
Router# show ip mroute 239.252.1.1
IP Multicast Routing Table
Flags:D - Dense, S - Sparse, C - Connected, L - Local, P - Pruned
       R - RP-bit set, F - Register flag, T - SPT-bit set, J - Join SPT
       M - MSDP created entry, X - Proxy Join Timer Running
       A - Advertised via MSDP
Outgoing interface flags:H - Hardware switched
Timers:Uptime/Expires
```

```
Interface state:Interface, Next-Hop or VCD, State/Mode

(*, 239.252.1.1), 04:04:59/00:02:59, RP 80.0.0.2, flags:SJ
  Incoming interface:Vlan800, RPF nbr 80.0.0.2
  Outgoing interface list:
    Vlan10, Forward/Dense, 01:29:57/00:00:00, H

(22.0.0.10, 239.252.1.1), 00:00:19/00:02:41, flags:JT
  Incoming interface:Vlan800, RPF nbr 80.0.0.2, RPF-MFD
  Outgoing interface list:
    Vlan10, Forward/Dense, 00:00:19/00:00:00, H
```

Monitoring IP MMLS on the MSFC

The **show mls ip multicast** command displays detailed information about IP MMLS.

To display the detailed IP MMLS information on the MSFC, perform one of these tasks:

Task	Command
Display IP MMLS group information.	Router# **show mls ip multicast group** *group-address* [**interface** *type number* | **statistics**]
Display IP MMLS details for all interfaces.	Router# **show mls ip multicast interface** *type number* [**statistics** | **summary**]
Display a summary of IP MMLS information.	Router# **show mls ip multicast summary**
Display IP MMLS statistics.	Router# **show mls ip multicast statistics**
Display IP MMLS source information.	Router# **show mls ip multicast source** *ip-address* [**interface** *type number* | **statistics**]

This example shows how to display the IP MMLS statistics on the MSFC:

```
Router# show mls ip multicast statistics
MLS Multicast configuration and state:
    Router Mac:0050.0f2d.9bfd, Router IP:1.12.123.234
    MLS multicast operating  state:ACTIVE
    Maximum number of allowed outstanding messages:1
    Maximum size reached from feQ:1
    Feature Notification sent:5
    Feature Notification Ack received:4
    Unsolicited Feature Notification received:0
    MSM sent:33
    MSM ACK received:33
    Delete notifications received:1
    Flow Statistics messages received:248

MLS Multicast statistics:
    Flow install Ack:9
    Flow install Nack:0
    Flow update Ack:2
    Flow update Nack:0
    Flow delete Ack:0
    Complete flow install Ack:10
    Complete flow install Nack:0
    Complete flow delete Ack:1
    Input VLAN delete Ack:4
    Output VLAN delete Ack:0
    Group delete sent:0
```

```
    Group delete Ack:0
    Global delete sent:7
    Global delete Ack:7

    L2 entry not found error:0
    Generic error :3
    LTL entry not found error:0
    MET entry not found error:0
    L3 entry exists error :0
    Hash collision error :0
    L3 entry not found error:0
    Complete flow exists error :0
```

This example shows how to display information on a specific IP MMLS entry on the MSFC:

```
Router# show mls ip multicast 224.1.1.1
Multicast hardware switched flows:
(1.1.13.1, 224.1.1.1) Incoming interface: Vlan13, Packets switched: 61590
Hardware switched outgoing interfaces: Vlan20 Vlan9
RFD-MFD installed: Vlan13

(1.1.9.3, 224.1.1.1) Incoming interface: Vlan9, Packets switched: 0
Hardware switched outgoing interfaces: Vlan20
RFD-MFD installed: Vlan9

(1.1.12.1, 224.1.1.1) Incoming interface: Vlan12, Packets switched: 62010
Hardware switched outgoing interfaces: Vlan20 Vlan9
RFD-MFD installed: Vlan12

(1.1.12.3, 224.1.1.1) Incoming interface: Vlan12, Packets switched: 61980
Hardware switched outgoing interfaces: Vlan20 Vlan9
RFD-MFD installed: Vlan12

(1.1.11.1, 224.1.1.1) Incoming interface: Vlan11, Packets switched: 62430
Hardware switched outgoing interfaces: Vlan20 Vlan9
RFD-MFD installed: Vlan11

(1.1.11.3, 224.1.1.1) Incoming interface: Vlan11, Packets switched: 62430
Hardware switched outgoing interfaces: Vlan20 Vlan9
RFD-MFD installed: Vlan11

Total hardware switched installed: 6
Router#
```

This example shows how to display a summary of the IP MMLS information on the MSFC:

```
Router# show mls ip multicast summary
7 MMLS entries using 560 bytes of memory
Number of partial hardware-switched flows:2
Number of complete hardware-switched flows:5
Router#
```

Using Debug Commands on the IP MMLS MSFC

Table 14-9 describes the IP MMLS-related debug troubleshooting commands.

Table 14-9 IP MMLS Debug Commands

Command	Description
[no] debug mls ip multicast group *group_id group_mask*	Configures filtering that applies to all the other multicast debugging commands.
[no] debug mls ip multicast events	Displays the IP MMLS events.
[no] debug mls ip multicast errors	Turns on the debug messages for the multicast MLS-related errors.
[no] debug mls ip multicast messages	Displays IP MMLS messages from/to the hardware switching engine.
[no] debug mls ip multicast all	Turns on all the IP MMLS messages.
[no] debug mdss error	Turns on the MDSS[1] error messages.
[no] debug mdss events	Turns on the MDSS-related events.
[no] debug mdss all	Turns on all the MDSS messages.

1. MDSS = Multicast Distributed Switching Services

Using Debug Commands on the SCP

Table 14-10 describes the Serial Control Protocol (SCP)-related debug commands to troubleshoot the SCP that runs over the Ethernet out-of-band channel (EOBC).

Table 14-10 SCP Debug Commands

Command	Description
[no] debug scp async	Displays the trace for asynchronous data in and out of the SCP system.
[no] debug scp data	Displays the packet data trace.
[no] debug scp errors	Displays the errors and warnings in the SCP.
[no] debug scp packets	Displays the packet data in and out of the SCP system.
[no] debug scp timeouts	Reports the timeouts.
[no] debug scp all	Turns on all the SCP debugging messages.

Displaying Global IP MMLS Information on the Supervisor Engine

These sections describe how to configure IP MMLS on Supervisor Engine 1:

- Displaying IP MMLS Configuration Information, page 14-38
- Displaying IP MMLS Statistics, page 14-38
- Clearing IP MMLS Statistics, page 14-39
- Displaying IP MMLS Entries, page 14-39

Note IP MMLS is permanently enabled on Supervisor Engine 1 and cannot be disabled.

Note To configure IP MMLS on the MSFC, see the "Configuring IP MMLS on the MSFC" section on page 14-32.

Displaying IP MMLS Configuration Information

The **show mls multicast** command displays the global IP MMLS configuration information and the state of the participating MSFCs.

To display the global IP MMLS configuration information, perform this task:

Task	Command
Display the global IP MMLS configuration information.	**show mls multicast**

This example shows how to display the global IP MMLS configuration information:

```
Console> (enable) show mls multicast
Admin Status: Enabled
Operational Status: Active
Configured flow mask is {Destination-source-vlan flow}
Active Entries = 10
Router include list :
1.1.9.254 (Active)
1.1.5.252 (Active)
Console> (enable)
```

Displaying IP MMLS Statistics

The **show mls multicast statistics** command displays the IP MMLS statistics for the multicast MSFCs.

To display the IP MMLS statistics for the multicast MSFCs, perform this task:

Task	Command
Display the IP multicast MSFC statistics.	**show mls multicast statistics** [*ip_addr*]

This example shows how to display the IP MMLS statistics for the multicast MSFCs:

```
Console (enable) show mls multicast statistics
Router IP          Router Name        Router MAC
-------------------------------------------------------
1.1.9.254          ?                  00-50-0f-06-3c-a0

Transmit:
  Delete Notifications:                    23
  Acknowledgements:                        92
  Flow Statistics:                         56

Receive:
  Open Connection Requests:                1
  Keep Alive Messages:                     72
  Shortcut Messages:                       19
    Shortcut Install TLV:                  8
    Selective Delete TLV:                  4
    Group Delete TLV:                      0
    Update TLV:                            3
    Input VLAN Delete TLV:                 0
    Output VLAN Delete TLV:                0
    Global Delete TLV:                     0
    MFD Install TLV:                       7
    MFD Delete TLV:                        0
```

```
Router IP          Router Name        Router MAC
-------------------------------------------------------
1.1.5.252          ?                  00-10-29-8d-88-01

Transmit:
  Delete Notifications:                    22
  Acknowledgements:                        75
  Flow Statistics:                         22

Receive:
  Open Connection Requests:                1
  Keep Alive Messages:                     68
  Shortcut Messages:                       6
    Shortcut Install TLV:                  4
    Selective Delete TLV:                  2
    Group Delete TLV:                      0
    Update TLV:                            0
    Input VLAN Delete TLV:                 0
    Output VLAN Delete TLV:                0
    Global Delete TLV:                     0
    MFD Install TLV:                       4
    MFD Delete TLV:                        0
Console (enable)
```

Clearing IP MMLS Statistics

The **clear mls multicast statistics** command clears the IP MMLS statistics for all the participating MSFCs.

To clear the IP MMLS statistics, perform this task in privileged mode:

Task	Command
Clear the IP MMLS statistics.	**clear mls multicast statistics**

This example shows how to clear the IP MMLS statistics:

```
Console> (enable) clear mls multicast statistics
All statistics for the MLS routers in include list are cleared.
Console> (enable)
```

Displaying IP MMLS Entries

The **show mls multicast entry** command displays information about the multicast flows that are handled by the PFC. You can display the entries that are based on any combination of the participating MSFC, the VLAN, the multicast group address, or the multicast traffic source.

To display information about the IP MMLS entries, perform this task in privileged mode:

Task	Command
Display information about the IP MMLS entries.	**show mls multicast entry** [[[*mod*] [**vlan** *vlan_id*] [**group** *ip_addr*] [**source** *ip_addr*]] | [**all**]]

This example shows how to display all the IP MMLS entries:

```
Console> (enable) show mls multicast entry all
Router IP       Dest IP         Source IP       Pkts       Bytes       InVlan  OutVlans
--------------- --------------- --------------- ---------- ----------- ------- --------

1.1.5.252       224.1.1.1       1.1.11.1        15870      2761380     20
1.1.9.254       224.1.1.1       1.1.12.3        473220     82340280    12
1.1.5.252       224.1.1.1       1.1.12.3        15759      2742066     20
1.1.9.254       224.1.1.1       1.1.11.1        473670     82418580    11
1.1.5.252       224.1.1.1       1.1.11.3        15810      2750940     20
1.1.9.254       224.1.1.1       1.1.12.1        473220     82340280    12
1.1.5.252       224.1.1.1       1.1.13.1        15840      2756160     20
1.1.9.254       224.1.1.1       1.1.13.1        472770     82261980    13
1.1.5.252       224.1.1.1       1.1.12.1        15840      2756160     20
1.1.9.254       224.1.1.1       1.1.11.3        473667     82418058    11
Total Entries: 10
Console> (enable)
```

This example shows how to display the IP MMLS entries for a specific MSFC:

```
Console> (enable) show mls multicast entry 15
Router IP       Dest IP         Source IP       Pkts       Bytes       InVlan  OutVlans
--------------- --------------- --------------- ---------- ----------- ------- --------

1.1.5.252       224.1.1.1       1.1.11.1        15870      2761380     20
1.1.5.252       224.1.1.1       1.1.12.3        15759      2742066     20
1.1.5.252       224.1.1.1       1.1.11.3        15810      2750940     20
1.1.5.252       224.1.1.1       1.1.13.1        15840      2756160     20
1.1.5.252       224.1.1.1       1.1.12.1        15840      2756160     20
Total Entries: 5
Console> (enable)
```

This example shows how to display the IP MMLS entries for a specific multicast group address:

```
Console> (enable) show mls multicast entry group 226.0.1.3 short
Router IP      Dest IP     Source IP    InVlan Pkts   Bytes     OutVlans
-------------- ----------- ------------ ------ ------ --------- ---------
171.69.2.1     226.0.1.3   172.2.3.8    20     171    23512     10,201,22,45
171.69.2.1     226.0.1.3   172.3.4.9    12     25     3120      8,20
Total Entries: 2
Console> (enable)
```

This example shows how to display the IP MMLS entries for a specific MSFC and a specific multicast source address:

```
Console> (enable) show mls multicast entry 15 1.1.5.252 source 1.1.11.1 short
Router IP       Dest IP         Source IP       Pkts       Bytes
 InVlan  OutVlans
--------------- --------------- --------------- ---------- --------------------
  ------ ----------
172.20.49.159   224.1.1.6       1.1.40.4        368        57776
  40     23,25
172.20.49.159   224.1.1.71      1.1.22.2        99         65142
  22     30,37
172.20.49.159   224.1.1.8       1.1.22.2        396        235620
  22     13,19
Console> (enable)
```

CHAPTER 15

Configuring Access Control

This chapter describes how to configure the access control lists (ACLs) on the Catalyst 6500 series switches. The configuration of the ACLs depends on the type of hardware that you install on your supervisor engine. See the "Hardware Requirements" section on page 15-2 for more information.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

Note For detailed information on configuring policy-based ACLs (PBACLs), see the "Configuring Policy-Based ACLs" section on page 44-21.

This chapter consists of these sections:

Note Except where specifically differentiated, the information and procedures in this chapter apply to Supervisor Engine 32 with Policy Feature Card 3B/3BXL (PFC3B/PFC3BXL), Supervisor Engine 720 with PFC3A/PFC3B/PFC3BXL, Supervisor Engine 2 with PFC2, and Supervisor Engine 1 with PFC.

Understanding How ACLs Work

Traditionally, switches operated at Layer 2 only; switches switched traffic within a VLAN and routers routed traffic between the VLANs. Catalyst 6500 series switches with the Multilayer Switch Feature Card (MSFC) can accelerate packet routing between VLANs by using Layer 3 switching (Multilayer Switching [MLS]). The switch first bridges the packet, the packet is then routed internally without going to the router, and then the packet is bridged again to send it to its destination. During this process, the switch can access control *all* packets that it switches *including* the packets that are bridged within a VLAN.

Cisco IOS ACLs provide access control for the routed traffic between the VLANs, and the VLAN ACLs (VACLs) provide access control for *all* packets.

The standard and extended Cisco IOS ACLs are used to classify the packets. The classified packets can be subject to a number of features such as access control (security), encryption, policy-based routing, and so on. The standard and extended Cisco IOS ACLs are configured only on the router interfaces and applied on the routed packets.

The VACLs can provide access control that is based on the Layer 3 addresses for the IP and IPX protocols. The unsupported protocols are access controlled through the MAC addresses. A VACL is applied to all packets (bridged and routed) and can be configured on any VLAN interface. Once a VACL is configured on a VLAN, all packets (routed or bridged) entering the VLAN are checked against the VACL. The packets can either enter the VLAN through a switch port or through a router port after being routed.

Note With Supervisor Engine 720 (PFC3A/PFC3B/PFC3BXL) and Supervisor Engine 32 (PFC3B/PFC3BXL), the IPX routing is done through the software and the IPX Cisco IOS ACLs and IPX VACLs are not supported. You can match the IPX packets using the MAC VACLs. You can enter the **ipx-arpa** keyword to match the IPX ARPA frames. Use 0xffff EtherType to match on IPX non-ARPA frames and frames with an EtherType of 0xffff. For information on configuring the MAC VACLs, see the "Creating a Non-IP Version 4/Non-IPX VACL (MAC VACL) and Adding ACEs" section on page 15-52.

Hardware Requirements

The hardware that is required to configure the ACLs on Catalyst 6500 series switches is as follows:

- Cisco IOS ACLs:
  - Supervisor Engine 1 and Policy Feature Card (PFC) and MSFC or MSFC2
  - Supervisor Engine 2 and PFC2 and MSFC2
  - Supervisor Engine 720 and PFC3A/PFC3B/PFC3BXL and MSFC3
  - Supervisor Engine 32 and PFC3B/PFC3BXL and MSFC2A

- VACLs and QoS ACLs:
  - Supervisor Engine 1 and PFC
  - Supervisor Engine 2 and PFC2
  - Supervisor Engine 720 and PFC3A/PFC3B/PFC3BXL
  - Supervisor Engine 32 and PFC3B/PFC3BXL

Note The quality of service (QoS) feature set that is supported on your switch is determined by the switching engine daughter card that is installed on the supervisor engine. See Chapter 52, "Configuring QoS" for more information.

Supported ACLs

These sections describe the ACLs that are supported by the Catalyst 6500 series switches:

- QoS ACLs, page 15-3
- Cisco IOS ACLs, page 15-3
- VACLs, page 15-4

QoS ACLs

You can configure the QoS ACLs on the switch; see Chapter 52, "Configuring QoS."

Cisco IOS ACLs

Cisco IOS ACLs are configured on the MSFC VLAN interfaces. An ACL provides access control and consists of an ordered set of access control entries (ACEs). Many other features also use ACLs for specifying flows. For example, Web Cache Redirect (through the Web Cache Coordination Protocol [WCCP]) uses the ACLs to specify the HTTP flows that can be redirected to a Web cache engine.

Most Cisco IOS features are applied on the interfaces for specific directions (inbound versus outbound). However, some features use the ACLs globally. For such features, the ACLs are applied on all interfaces for a given direction. As an example, TCP intercept uses a global ACL that is applied on all outbound interfaces.

One Cisco IOS ACL can be used with multiple features for a given interface, and one feature can use multiple ACLs. When a single ACL is used by multiple features, Cisco IOS software examines it multiple times.

Cisco IOS software examines the ACLs that are associated with the features that are configured on a given interface and a direction. As the packets enter the router on a given interface, Cisco IOS software examines the ACLs that are associated with all the inbound features that are configured on that interface for the following:

- Inbound ACLs (standard, extended, and/or reflexive)
- Encryption ACLs (not supported on the MSFC)
- Policy routing ACLs
- Network Address Translation (NAT) for outside-to-inside translation

After the packets are routed and before they are forwarded out to the next hop, Cisco IOS software examines all ACLs that are associated with the outbound features configured on the egress interface for the following:

- Outbound ACLs (standard, extended, and/or reflexive)
- Encryption ACLs (not supported on the MSFC)
- NAT ACLs (for inside-to-outside translation)
- WCCP ACL
- TCP intercept ACL

VACLs

The following sections describe the VACLs:

VACL Overview

The VACLs can access control *all* traffic. You can configure the VACLs on the switch to apply to all packets that are routed into or out of a VLAN or are bridged within a VLAN. The VACLs are strictly for security packet filtering and redirecting traffic to specific physical switch ports. Unlike the Cisco IOS ACLs, the VACLs are not defined by direction (input or output).

You can configure the VACLs on the Layer 3 addresses for IP and IPX. All other protocols are access controlled through the MAC addresses and EtherType using the MAC VACLs.

Caution The IP traffic and IPX traffic are not access controlled by the MAC VACLs. All other traffic types (AppleTalk, DECnet, and so on) are classified as MAC traffic; the MAC VACLs are used to access control this traffic.

Note With Supervisor Engine 720 (PFC3A/PFC3B/PFC3BXL) and Supervisor Engine 32 (PFC3B/PFC3BXL), the IPX routing is done through the software and the IPX Cisco IOS ACLs and IPX VACLs are not supported. You can match the IPX packets using the MAC VACLs. You can enter the **ipx-arpa** keyword to match the IPX ARPA frames. Use 0xffff EtherType to match on the IPX non-ARPA frames and the frames with an EtherType of 0xffff. For information on configuring the MAC VACLs, see the "Creating a Non-IP Version 4/Non-IPX VACL (MAC VACL) and Adding ACEs" section on page 15-52.

You can enforce the VACLs only on the packets going through the Catalyst 6500 series switch; you cannot enforce the VACLs on the traffic between the hosts on a hub or another switch that is connected to the Catalyst 6500 series switch.

ACEs Supported in VACLs

A VACL contains an ordered list of access control entries (ACEs). Each VACL can contain ACEs of only one type. Each ACE contains a number of fields that are matched against the contents of a packet. Each field can have an associated bit mask to indicate which bits are relevant. An action is associated with each ACE that describes what the system should do with the packet when a match occurs. The action is feature dependent. Catalyst 6500 series switches support three types of ACEs in the hardware:

- IP ACEs
- IPX ACEs
- Ethernet ACEs

Table 15-1 lists the parameters that are associated with each ACE type.

Table 15-1 ACE Types and Parameters

ACE Type	TCP or UDP[1]	ICMP[1]	Other IP[1]	IPX	Ethernet[2]
Layer 4 parameters	Source port				
	Source port operator				
	Destination port				
	Destination port operator	ICMP code[1]			
	N/A	ICMP type	N/A		
Layer 3 parameters	IP ToS byte	IP ToS byte	IP ToS byte		
	IP source address	IP source address	IP source address	IPX source network	
	IP destination address	IP destination address	IP destination address	IPX destination network	
				IPX destination node	
	TCP or UDP	ICMP	Other protocol	IPX packet type	
Layer 2 parameters					EtherType
					Ethernet source address
					Ethernet destination address

1. IP ACEs.
2. For Ethernet packets that are not IP version 4 or IPX.

Handling Fragmented and Unfragmented Traffic

TCP/UDP or any Layer 4 protocol traffic, when fragmented, loses the Layer 4 information (Layer 4 source/destination ports). This situation makes it difficult to enforce security that is based on the application. However, you can identify the fragments and distinguish them from the rest of the TCP/UDP traffic.

The Layer 4 parameters of the ACEs can filter the unfragmented and fragmented traffic with fragments that have offset 0. The IP fragments that have an offset other than 0 miss the Layer 4 port information and cannot be filtered. The following examples show how the ACEs handle the packet fragmentation.

This example shows that if the traffic from 1.1.1.1, port 68 is fragmented, only the first fragment goes to port 4/3, and the rest of the traffic from port 68 does not hit this entry.

```
redirect 4/3 tcp host 1.1.1.1 eq 68 host 255.255.255.255
```

This example shows that the traffic coming from 1.1.1.1, port 68 and going to 2.2.2.2, port 34 is permitted. If the packets are fragmented, the first fragment hits this entry and is permitted; the fragments that have an offset other than 0 are also permitted as a default result for the fragments.

```
permit tcp host 1.1.1.1 eq 68 host 2.2.2.2 eq 34
```

This example shows that the fragment that has offset 0 of the traffic from 1.1.1.1, port 68 going to 2.2.2.2, port 34 is denied. The fragments that have an offset other than 0 are permitted as a default.

```
deny tcp host 1.1.1.1 eq 68 host 2.2.2.2 eq 34
```

In the releases prior to software release 6.1(1), the fragment filtering was completely transparent; you would type an ACE such as **permit tcp .... port eq** *port_number* and the software would implicitly install the following ACE at the top of the ACL: permit tcp any any fragments.

Software release 6.1(1) and later releases, have a **fragment** option. If you do not specify the **fragment** keyword, the behavior is the same as in the previous releases. If you specify the **fragment** keyword, the system does not automatically install a global permit statement for the fragments. This keyword allows you to control how the fragments are handled.

In this example, 10.1.1.2 is configured to serve the HTTP connections. If you do not use a fragment ACE, all the fragments for the TCP traffic are permitted as the **permit tcp any any fragments** ACE is added automatically at the top of the ACL as follows:

```
permit tcp any any fragments
```

1. **permit tcp any host 10.1.1.2 eq www**
2. **deny ip any host 10.1.1.2**
3. **permit ip any any**

In the above example, if you change the entry 1 as follows:

1. deny tcp any host 10.1.1.2 eq www

A **permit tcp any any fragments** ACE is not added at the top of the ACL. If the entry is a **deny** statement, the next access-list entry is processed.

Note The **deny** statements are handled differently for the noninitial fragments versus the nonfragmented or initial fragments.

When you specify the **fragment** keyword, the system does not install the global permit TCP or UDP fragments statement. When you specify the **fragment** keyword for at least one ACE, the software implicitly installs the ACEs to permit the flows to a specific IP address (or subnet) that you specify.

In this ACL example, the **deny tcp any host 10.1.1.2 fragment** entry stops the fragmented traffic going to all TCP ports on host 10.1.1.2. Later in the ACL, the **permit udp any host 10.1.1.2 eq 69** entry allows the clients to connect to the TFTP server 10.1.1.2. The system automatically installs a **permit for all fragments of udp traffic to host 10.1.1.2** ACE; otherwise, the fragments would be denied by the entry **deny ip any host 10.1.1.2**.

1. **deny tcp any host 10.1.1.2 fragment**
2. **permit tcp any host 10.1.1.2 eq www**
3. **permit udp any host 10.1.1.2 eq 69**
4. **permit udp any gt 1023 10.1.1.2 gt 1023**
5. **deny ip any host 10.1.1.2**
6. **permit ip any any**

If you explicitly want to stop the fragmented UDP traffic to host 10.1.1.2, enter **deny udp any host 10.1.1.2 fragment** before entry number 3 as shown in this example:

[...]

3. **deny udp any host 10.1.1.2 fragment**
4. **permit udp any host 10.1.1.2 eq 69**
5. **permit udp any gt 1023 10.1.1.2 gt 1023**

[...]

Applying Cisco IOS ACLs and VACLs on VLANs

This section describes how to apply the Cisco IOS ACLs and VACLs to the VLAN for the bridged, routed, and multicast packets.

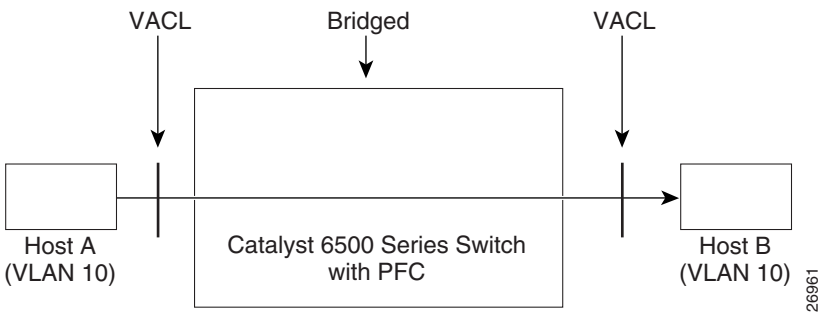
These sections show how the ACLs and the VACLs are applied:

- Bridged Packets, page 15-7
- Routed Packets, page 15-8
- Multicast Packets, page 15-8

Bridged Packets

Figure 15-1 shows how an ACL is applied on the bridged packets. For the bridged packets, only the Layer 2 ACLs are applied to the input VLAN.

Figure 15-1 Applying ACLs on Bridged Packets

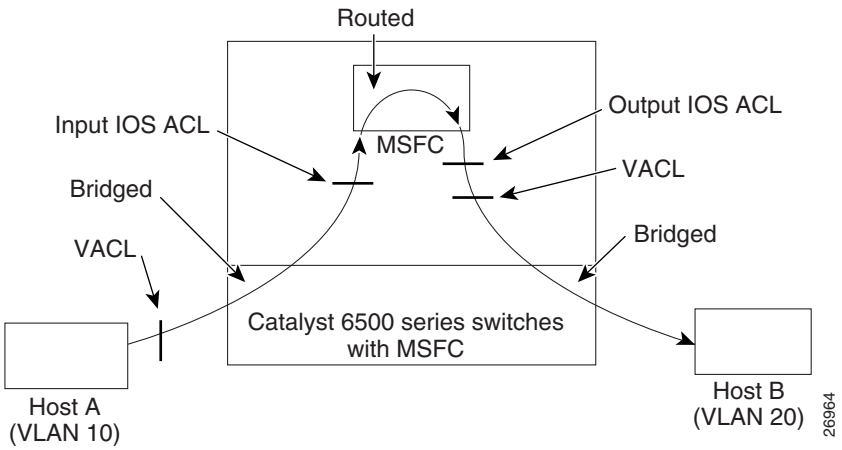


Routed Packets

Figure 15-2 shows how the ACLs are applied on the routed/Layer 3-switched packets. For the routed/Layer 3-switched packets, the ACLs are applied in the following order:

1. VACL for input VLAN
2. Input Cisco IOS ACL
3. Output Cisco IOS ACL
4. VACL for output VLAN

Figure 15-2 Applying ACLs on Routed Packets



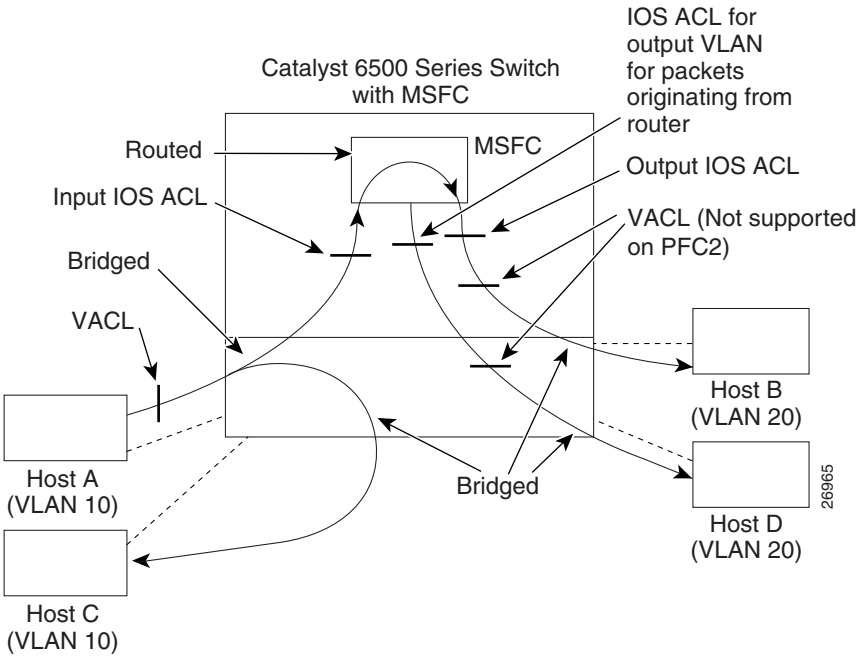
Multicast Packets

Figure 15-3 shows how the ACLs are applied on the packets that need multicast expansion. For the packets that need multicast expansion, the ACLs are applied in the following order:

1. Packets that need multicast expansion:
   a. VACL for input VLAN
   b. Input Cisco IOS ACL

2. Packets after multicast expansion:
   a. Output Cisco IOS ACL
   b. VACL for output VLAN
3. Packets originating from the router:
   a. VACL for output VLAN

Figure 15-3 Applying ACLs on Multicast Packets



Using Cisco IOS ACLs in your Network

Note Configuring Cisco IOS ACLs on the Catalyst 6500 series switch routed-VLAN interfaces is the same as configuring the ACLs on the other Cisco routers. To configure the Cisco IOS ACLs, see the "Unsupported Features" section on page 15-44 and the "VACL Configuration Guidelines" section on page 15-45. In addition, refer to the Cisco IOS configuration guides and command reference publication. To configure the ACLs for IP, refer to the "Configuring IP Services" chapter in the *Network Protocols Configuration Guide*, Part 1.

When a feature is configured on the router to process traffic (such as NAT), the Cisco IOS ACL that is associated with the feature determines the specific traffic that is bridged to the router instead of being switched in Layer 3. The router then applies the feature and routes the packet normally. Some exceptions to this process are described in the "Hardware and Software Handling of Cisco IOS ACLs with PFC" section on page 15-10.

Note In the systems with redundant MSFCs, the ACL configurations for Cisco IOS ACLs and VACLs must be the same on both MSFCs.

Caution For PFC: By default, the MSFC sends Internet Control Message Protocol (ICMP) unreachables when a packet is denied by an access group. These access-group denied packets are not dropped in the hardware but are bridged to the MSFC so that the MSFC can generate the ICMP-unreachable message. To drop the access-group denied packets in the hardware, you must disable the ICMP unreachables using the **no ip unreachables** interface configuration command. The **ip unreachables** command is enabled by default.

For PFC2 and PFC3A/PFC3B/PFC3BXL: If the IP unreachables or IP redirect is enabled on an interface, the deny is performed in the hardware although a small number of packets are sent to the MSFC2/MSFC3 to generate the appropriate ICMP-unreachable messages.

These sections describe the hardware and software handling of the ACLs with PFC, PFC2, and PFC3A/PFC3B/PFC3BXL:

- Hardware and Software Handling of Cisco IOS ACLs with PFC, page 15-10
- Hardware and Software Handling of Cisco IOS ACLs with PFC2 and PFC3A/PFC3B/PFC3BXL, page 15-13

Hardware and Software Handling of Cisco IOS ACLs with PFC

This section describes how Cisco IOS ACLs with the PFC are handled by the hardware and the software.

Note For information on Cisco IOS ACLs with PFC2 and PFC3A/PFC3B/PFC3BXL, see the "Hardware and Software Handling of Cisco IOS ACLs with PFC2 and PFC3A/PFC3B/PFC3BXL" section on page 15-13.

ACL feature processing requires forwarding of some flows by the software. The forwarding rate for the software-forwarded flows is substantially less than for the hardware-forwarded flows. The flows that require logging, as specified by the ACL, are handled in the software without impacting the non-log flow forwarding in the hardware.

Note When you enter the **show ip access-list** command, the match count that is displayed does not account for the packets that are access controlled in the hardware.

Note IPX Cisco IOS ACLs with the source host node number specified cannot be enforced on the switch in the hardware; the MSFC has to process the ACL in the software. This process *significantly* degrades system performance.

These sections describe how the different types of ACLs and traffic flows are handled by the hardware and the software:

Security Cisco IOS ACLs

The IP and IPX security Cisco IOS ACLs with PFC are as follows:

- The flows that match a “deny” statement in a security ACL are dropped by the hardware if “ip unreachables” is disabled. The flows matching a “permit” statement are switched in the hardware.
- Permit and deny actions of the standard and extended ACLs (input and output) for security access control are handled in the hardware.
- IP accounting for an ACL access violation on a given interface is supported by forwarding all denied packets for that interface to the software without impacting other flows.
- Dynamic (lock and key) ACL flows are supported in the hardware; however, idle timeout is not supported.
- IPX standard input and output ACLs are supported in the hardware when the ACL parameters are IPX source network, destination network, and/or destination node. If the ACL contains any other parameters, it is handled in the software.
- IPX extended input and output ACLs are supported in the hardware when the ACL parameters are IPX source network, destination network, destination node, and/or protocol type.
- ACL flows requiring logging are handled in the software without impacting non-log flow forwarding in the hardware.

Reflexive ACLs

Up to 512 simultaneous reflexive sessions are supported in the hardware. When the reflexive ACLs are applied, the flow mask is changed to VLAN-full flow.

TCP Intercept

TCP intercept implements the software to protect the TCP servers from the TCP SYN-flooding attacks, which are denial-of-service attacks. TCP intercept helps prevent the SYN-flooding attacks by intercepting and validating the TCP connection requests. In intercept mode, the TCP intercept software intercepts the TCP synchronization (SYN) packets from the clients to the servers that match an extended access list. The software establishes a connection with the client on behalf of the destination server, and if successful, establishes the connection with the server on behalf of the client and binds the two

half-connections together transparently. This process ensures that the connection attempts from the unreachable hosts never reach the server. The software continues to intercept and forward the packets throughout the duration of the connection.

Policy Routing

The policy routing-required flows are handled in the software without impacting the non-policy routed flow forwarding in the hardware. When a route map contains multiple "match" clauses, all conditions that are imposed by these match clauses must be met before a packet is policy routed. However, for the route maps that contain both "match ip address" and "match length," all traffic matching the ACL in the "match ip address" clause is forwarded to the software regardless of the match length criteria. For the route maps that contain only the match length clauses, all packets that are received on the interface are forwarded to the software.

When you enable hardware policy routing using the **mls ip pbr** global command, all policy routing occurs in the hardware.

Caution If you use the **mls ip pbr** command to enable policy routing, policy routing is applied in the hardware for all interfaces regardless of which interface was configured for the policy routing.

WCCP

The HTTP requests that are subject to Web Cache Coordination Protocol (WCCP) redirection are handled in the software; the HTTP replies from the server and the Cache Engine are handled in the hardware.

NAT

The NAT-required flows are handled in the software without impacting non-NAT flow forwarding in the hardware.

Unicast RPF Check

The unicast RPF feature is supported in the hardware on the PFC. For ACL-based RPF checks, the traffic that is denied by the unicast RPF ACL is forwarded to the MSFC for RPF validation.

Caution With ACL-based unicast RPF, the packets that are denied by the ACL are sent to the CPU for RPF validation. In the event of DoS attacks, these packets will most likely match the deny ACE and be forwarded to the CPU. Under heavy traffic conditions, this process could cause high CPU utilization.

Note The drop-suppress statistics for the ACL-based RPF check is not supported.

Bridge-Groups

Cisco IOS bridge-group ACLs are handled in the software.

Hardware and Software Handling of Cisco IOS ACLs with PFC2 and PFC3A/PFC3B/PFC3BXL

This section describes how Cisco IOS ACLs are handled by the hardware and the software in the switches that are configured with the PFC2 and PFC3A/PFC3B/PFC3BXL.

ACL feature processing requires forwarding some flows to the software. The forwarding rate for software-forwarded flows is substantially less than for the hardware-forwarded flows. The flows that require logging as specified by the ACL are handled in the software without impacting non-log flow forwarding in the hardware.

Note When you enter the **show ip access-list** command, the match count displayed does not account for the packets that are access controlled in the hardware.

Note The IPX Cisco IOS ACLs with the source host node number specified cannot be enforced on the switch in the hardware; the MSFC has to process the ACL in the software. This process *significantly* degrades the system performance.

Note With Supervisor Engine 720 (PFC3A/PFC3B/PFC3BXL) and Supervisor Engine 32 (PFC3B/PFC3BXL), the IPX routing is done through the software and the IPX Cisco IOS ACLs and IPX VACLs are not supported. You can match the IPX packets using the MAC VACLs. You can enter the **ipx-arpa** keyword to match the IPX ARPA frames. Use 0xffff EtherType to match on the IPX non-ARPA frames and the frames with an EtherType of 0xffff. For information on configuring the MAC VACLs, see the "Creating a Non-IP Version 4/Non-IPX VACL (MAC VACL) and Adding ACEs" section on page 15-52.

These sections describe how the different types of Cisco IOS ACLs and traffic flows are handled by the hardware and the software in the switches that are configured with the PFC2 or PFC3A/PFC3B/PFC3BXL:

Security Cisco IOS ACLs

The IP and IPX security Cisco IOS ACLs in the switches that are configured with the PFC2 or PFC3A/PFC3B/PFC3BXL are as follows:

- If either the "ip unreachables" or "ip redirect" options are enabled, most of the packets of the flows that match a "deny" statement in an ACL are dropped by the hardware. Only a few packets are processed in the software in order for the router to send the appropriate ICMP-unreachable message.
- The permit and deny actions of the standard and extended ACLs (input and output) for security access control are handled in the hardware.
- The IP accounting for an ACL access violation on a given interface is supported by forwarding all denied packets for that interface to the software without impacting other flows.
- The dynamic (lock and key) ACL flows are supported in the hardware; however, idle timeout is not supported.
- The IPX standard input and output ACLs are supported in the hardware when the ACL parameters are IPX source network, destination network, and/or destination node. If the ACL contains any other parameters, it is handled in the software.
- The IPX extended input and output ACLs are supported in the hardware when the ACL parameters are IPX source network, destination network, destination node, and/or protocol type.
- The ACL flows that require logging are handled in the software without impacting non-log flow forwarding in the hardware.

Rate Limiting for Cisco IOS ACL Logging

Rate limiting for Cisco IOS ACL logging limits the number of packets that are sent to the MSFC CPU for the bridged ACEs. An ACE is bridged when the result for the Cisco IOS ACL is a deny or permit with the log option specified. The bridge action can result in Cisco IOS ACL logging overloading the MSFC CPU. When you configure rate limiting for Cisco IOS ACL logging, the bridged ACEs are redirected to the MSFC with rate limiting.

Configuring Rate Limiting for Cisco IOS ACL Logging Guidelines

This section describes the guidelines for configuring rate limiting for Cisco IOS ACL logging:

- After entering the **set acllog ratelimit** *rate* command or the **clear acllog** command, you must either reset the MSFC or perform a shutdown/no shutdown on the MSFC interface(s) that have the ACEs with the **log** keyword applied.

  After entering the **set acllog ratelimit** *rate* command, performing a reset or shutdown/no shutdown causes the bridged ACEs to be redirected to the MSFC with rate limiting.

  After entering the **clear acllog** command, performing a reset or shutdown/no shutdown causes the switch to return to its previous behavior; the bridge action remains unchanged.
- The *rate* that is specified by entering the **set acllog ratelimit** *rate* command can be from 500 to 2000. The *rate* is the number of packets per second that hit a redirect ACE and are sent to the MSFC. If the actual number of packets per second is greater than the *rate* that you specify, the packets that exceed the specified *rate* are dropped. We recommend that you specify a *rate* of 500 packets per second.

Configuring Rate Limiting for Cisco IOS ACL Logging

To configure rate limiting for Cisco IOS ACL logging, perform this task in privileged mode:

	Task	Command
Step 1	Enable the ACL logging and specify a rate for Cisco IOS ACL logging rate limiting.	**set acllog ratelimit** *rate*
Step 2	Show the ACL logging status.	**show acllog**

This example shows how to enable the ACL logging and specify a rate of 500 for Cisco IOS ACL logging rate limiting:

```
Console> (enable) set acllog ratelimit 500
If the ACLs-LOG were already applied, the rate limit mechanism will be effective on system
restart, or after shut/no shut the interface.
Console> (enable)

Console> (enable) show acllog
ACL log rate limit enabled, rate = 500 pps.
Console> (enable)
```

This example shows how to clear (disable) ACL logging. After clearing ACL logging, the bridge action remains unchanged; the system behavior is the same as before the **set acllog ratelimit** command was issued.

```
Console> (enable) clear acllog
ACL log rate limit is cleared.
If the ACLs-LOG were already applied, the rate limit mechanism will be disabled on system
restart, or after shut/no shut the interface.
Console> (enable)
```

Reflexive ACLs

The ICMP packets are handled in the software. For the TCP/UDP flows, once the flow is established, they are handled in the hardware. When the reflexive ACLs are applied, the flow mask is changed to VLAN-full flow.

TCP Intercept

Note TCP intercept is not supported with Supervisor Engine 720 (PFC3A/PFC3B/PFC3BXL) or Supervisor Engine 32 (PFC3B/PFC3BXL).

TCP intercept implements the software to protect the TCP servers from the TCP SYN-flooding attacks, which are denial-of-service attacks. TCP intercept helps prevent the SYN-flooding attacks by intercepting and validating the TCP connection requests. In intercept mode, the TCP intercept software intercepts the TCP synchronization (SYN) packets from the clients to the servers that match an extended access list. The software establishes a connection with the client on behalf of the destination server, and if successful, establishes the connection with the server on behalf of the client and binds the two half-connections together transparently. This process ensures that the connection attempts from the unreachable hosts never reach the server. The software continues to intercept and forward the packets throughout the duration of the connection.

The hardware support for TCP intercept on a PFC2 is as follows:

1. Once you configure TCP intercept, all TCP SYN packets that match the ACEs with a permit clause in the TCP intercept ACL, *and* which are permitted by the security ACL, are sent to the software to apply the TCP intercept functionality. This process occurs even if the security ACL does not have the SYN flag specified.
2. If a connection is established successfully, the following applies:
   a. If the TCP intercept is using intercept mode with timeout, all traffic belonging to the given connection/flow is handled in the software.
   b. For the other modes of TCP intercept, once the connection is successfully established, the software installs a hardware shortcut to switch the rest of the flow in the hardware.
3. If a connection is not established successfully, no other traffic can belong to that flow.

Policy Routing

The policy routing-required flows are handled in the hardware or the software depending on the route map. If the route map contains only a match IP address clause, and the set clause contains the next hop and the next hop is reachable, then the packet is forwarded in the hardware. When a route map contains multiple match clauses, all conditions that are imposed by these match clauses must be met before a packet is policy routed. However, for the route maps that contain both a match IP address clause and match length clause, all traffic matching the ACL in the match IP address clause is forwarded to the software regardless of the match length criteria. For the route maps that contain only match length clauses, all packets that are received on the interface are forwarded to the software.

Note The **mls ip pbr** command is not required (and not supported) on the PFC2 or PFC3A/PFC3B/PFC3BXL.

WCCP

Note WCCP is not supported with Supervisor Engine 720 or Supervisor Engine 32 in software releases 8.1(x) through 8.4(x).

The HTTP requests that are subject to WCCP redirection are handled in the software; the HTTP replies from the server and the cache engine are handled in the hardware.

NAT

The NAT-required flows are handled in the software without impacting the non-NAT flow forwarding in the hardware.

Unicast RPF Check

Unicast RPF is supported in the hardware on the PFC2 and PFC3A/PFC3B/PFC3BXL. For the ACL-based RPF checks, the traffic that is denied by the unicast RPF ACL is forwarded to the MSFC2 or MSFC3 for RPF validation.

Caution With ACL-based unicast RPF, the packets that are denied by the ACL are sent to the CPU for RPF validation. In the event of DoS attacks, these packets will most likely match the deny ACE and be forwarded to the CPU. Under heavy traffic conditions, this process could cause high CPU utilization.

Note The drop-suppress statistics for the ACL-based RPF check is not supported.

Bridge-Groups

Cisco IOS bridge-group ACLs are handled in the software.

Using VACLs with Cisco IOS ACLs

To access control both the bridged and routed traffic, you can use the VACLs only or a combination of Cisco IOS ACLs and VACLs. You can define Cisco IOS ACLs on both the input and output routed-VLAN interfaces, and you can define a VACL to access control the bridged traffic.

If a flow matches a VACL deny or redirect clause in the ACL, irrespective of the Cisco IOS ACL configuration, the flow is denied or redirected. The following caveats apply to Cisco IOS ACLs when they are used with VACLs:

- Packets that require logging on the outbound ACLs are not logged if they are denied by a VACL.
- NAT—VACLs are applied on the packets before NAT translation. If the translated flow should not be access controlled, the flow might get access controlled after the translation because of the VACL configuration.

Note The VACLs have an implicit deny at the end of the list; a packet is denied if it does not match any VACL ACE.

These sections describe the Cisco IOS ACL and VACL configuration guidelines and guidelines for Layer 4 operations:

- Configuring Cisco IOS ACLs and VACLs on the Same VLAN Interface Guidelines, page 15-17
- Layer 4 Operations Configuration Guidelines, page 15-23

Configuring Cisco IOS ACLs and VACLs on the Same VLAN Interface Guidelines

This section describes the guidelines for configuring a Cisco IOS ACL *and* a VACL on the same VLAN. These guidelines do not apply to the configurations where you are mapping Cisco IOS ACLs and VACLs on different VLANs.

The Catalyst 6500 series switch hardware provides one lookup for the security ACLs for each direction (input and output); you must merge a Cisco IOS ACL and a VACL when they are configured on the same VLAN. Merging the Cisco IOS ACL with the VACL might significantly increase the number of ACEs.

If you must configure a Cisco IOS ACL and a VACL on the same VLAN, use the following guidelines for both Cisco IOS ACL and VACL configurations.

Note To display the percentage of ACL storage that is being used, enter the **show security acl resource-usage** command.

These sections provide the Cisco IOS ACL and VACL configuration guidelines and examples:

Using the Implicit Deny Action

If possible, use the implicit deny action at the end of an ACL (deny any any) and define the ACEs to permit only allowed traffic. You can achieve this same effect by defining all the deny entries and specifying **permit ip any any** at the end of the list (see Example 1, page 15-20).

Grouping Actions Together

To define multiple actions in an ACL (permit, deny, redirect), group each action type. Example 3, page 15-20 shows what can happen when you do not group each type. In the example, the deny action in line 6 was grouped with the permit actions. If this deny action is removed, the result of merging would be 53 entries, instead of 329 entries.

Limiting the Number of Actions

An ACL with only the permit ACEs has two actions: permit and deny (because of the implicit deny at the end of the list). An ACL with permit and redirect has three actions: permit, redirect, and deny (because of the implicit deny at the end of the list).

When configuring an ACL, the best merge results are obtained when you specify only two different actions: permit and deny, redirect and permit, or redirect and deny.

Note With supervisor engine software release 7.1(1) or later releases, due to an improved algorithm for merging ACLs, you do not need to limit the number of actions when configuring an ACL.

To specify a redirect and deny ACL, do not use any permit ACEs. To specify a redirect and permit ACL, use permit ACEs, redirect ACEs, and for the last ACE, specify **permit ip any any**. If you specify **permit ip any any**, you will override the implicit deny ip any at the end of the list (see Example 4, page 15-21).

Avoiding Layer 4 Port Information

Avoid including Layer 4 information in an ACL because it will complicate the merging process. You will obtain the best merge results if the ACLs are filtered based on the IP addresses (source and destination) and not on the full flow (source IP address, destination IP address, protocol, and protocol ports).

If you need to specify the full flow, follow the recommendations in the "Using the Implicit Deny Action" section on page 15-18 and "Grouping Actions Together" section on page 15-18. If you cannot follow the recommendation because the ACL has both the IP and TCP/UDP/ICMP ACEs with Layer 4 information, put the Layer 4 ACEs at the end of the list to prioritize the traffic filtering based on the IP addresses.

Estimating Merge Results with Supervisor Engine Software Releases Prior to Release 7.1(1)

Note To see a comparison of the merge results when using supervisor engine software releases before software release 7.1(1) versus software release 7.1(1) or later releases, see the "Estimating Merge Results with Supervisor Engine Software Releases 7.1(1) or Later Releases" section on page 15-21.

If you follow the ACL guidelines when configuring the ACLs, you can get a rough estimate of the merge results for the ACLs.

The following formula uses ACL A, ACL B, and ACL C. If ACL C is the result of merging ACL A and ACL B, and you know the size of ACL A and ACL B, you can estimate the upper limit of the size of ACL C when no Layer 4 port information has been specified on ACL A and ACL B, as follows:

size of ACL C = (size of ACL A) x (size of ACL B) x (2)

Note In software releases prior to release 7.1(1), the formula is used as a guideline but the number of entries could go beyond the predicted range. In software release 7.1(1) and later releases, with the new ACL merge algorithm, the formula is accurate for all cases. If Layer 4 port information is specified, the upper limit could be higher even with the new algorithm. See the "Layer 4 Operations Configuration Guidelines" section on page 15-23 for detailed information.

Two ACL-merge algorithms are available—the binary decision diagram (BDD) and the order-dependent merge (ODM). ODM is the enhanced algorithm that was introduced in software release 7.1(1). The BDD algorithm was used in releases prior to software release 7.1(1). See the "Specifying the ACL-Merge Algorithm" section on page 15-47 for detailed configuration information.

Note With software release 8.1(1) and later releases, the BDD algorithm is no longer supported on any platform (PFC, PFC2, or PFC3A/PFC3B/PFC3BXL). The default ACL-merge algorithm is ODM. In software release 8.1(1) and later releases, the following command changes appear: The **set aclmerge algo** and **set aclmerge bdd** commands have been removed. The **show aclmerge** {**bdd** | **algo**} command has been reduced to **show aclmerge algo**.

These examples show the merge results for the various Cisco IOS ACL and VACL configurations. One VACL and one Cisco IOS ACL are configured on the same VLAN.

Example 1

This example shows that the VACL does not follow the recommended guidelines (in line 9, a deny action is defined instead of using the implicit deny action at the end of the ACL), and the resultant merge increases the number of ACEs:

```
******** VACL  ***********
1  permit udp host 194.72.72.33 194.72.6.160 0.0.0.15
2  permit udp host 147.150.213.94 194.72.6.64 0.0.0.15 eq bootps
3  permit udp 194.73.74.0 0.0.0.255 host 194.72.6.205 eq syslog
4  permit udp host 167.221.23.1 host 194.72.6.198 eq tacacs
5  permit udp 194.72.136.1 0.0.3.128 194.72.6.64 0.0.0.15 eq tftp
6  permit udp host 193.6.65.17 host 194.72.6.205 gt 1023
7  permit tcp any host 194.72.6.52
8  permit tcp any host 194.72.6.52 eq 113
9  deny tcp any host 194.72.6.51 eq ftp
10 permit tcp any host 194.72.6.51 eq ftp-data
11 permit tcp any host 194.72.6.51
12 permit tcp any eq domain host 194.72.6.51
13 permit tcp any host 194.72.6.51 gt 1023
14 permit ip  any host 1.1.1.1
******** Cisco IOS ACL ************
1  deny ip any host 239.255.255.255
2  permit ip any any
******** MERGE **********
has 91 entries entries
```

Example 2

In Example 1, if you follow the guidelines and remove line 9 (the implicit deny is then used at the end of the ACL) and modify lines 11 and 12 (lines 11 and 12 are modified so that the traffic that line 9 would have dropped is not permitted), you see the following equivalent ACL with improved merge results:

```
******** VACL **********
1  permit udp host 194.72.72.33 194.72.6.160 0.0.0.15
2  permit udp host 147.150.213.94 194.72.6.64 0.0.0.15 eq bootps
3  permit udp 194.73.74.0 0.0.0.255 host 194.72.6.205 eq syslog
4  permit udp host 167.221.23.1 host 194.72.6.198 eq tacacs
5  permit udp 194.72.136.1 0.0.3.128 194.72.6.64 0.0.0.15 eq tftp
6  permit udp host 193.6.65.17 host 194.72.6.205 gt 1023
7  permit tcp any host 194.72.6.52
8  permit tcp any host 194.72.6.52 eq 113
9  permit tcp any host 194.72.6.51 eq ftp-data
10 permit tcp any host 194.72.6.51 neq ftp
11 permit tcp any eq domain host 194.72.6.51 neq ftp
12 permit tcp any host 194.72.6.51 gt 1023
13 permit ip  any host 1.1.1.1
******** Cisco IOS ACL ************
1  deny ip any host 239.255.255.255
2  permit ip any any
******** MERGE ***********
has 78 entries
```

Example 3

This example shows that the VACL does not follow the recommended guidelines (all the action types are not grouped), and the resultant merge significantly increases the number of ACEs:

```
******** VACL  ***********
1  deny ip 0.0.0.0 255.255.255.0 any
2  deny ip 0.0.0.255 255.255.255.0 any
3  deny ip any 0.0.0.0 255.255.255.0
4  permit ip any host 239.255.255.255
5  permit ip any host 255.255.255.255
```

```
6  deny ip any 0.0.0.255 255.255.255.0
7  permit tcp any range 0 65534 any range 0 65534
8  permit udp any range 0 65534 any range 0 65534
9  permit icmp any any
10 permit ip any any
******** Cisco IOS ACL **********
1  deny ip any host 239.255.255.255
2  permit ip any any
******** MERGE **********
has 329 entries
```

Example 4

This example shows that the VACL does not follow the recommended guidelines (three different actions are specified), and the resultant merge significantly increases the number of ACEs:

```
******** VACL  ***********
1 redirect 4/25 tcp host 192.168.1.67 host 255.255.255.255
2 redirect 4/25 udp host 192.168.1.67 host 255.255.255.255
3 deny tcp any any lt 30
4 deny udp any any lt 30
5 permit ip any any
*******  Cisco IOS ACL ***********
1  deny ip any host 239.255.255.255
2  permit ip any any
*******  MERGE **********
has 142 entries
```

Example 5

This example shows that if you modify the VACL in Example 4 and specify only two different actions, the merge results are significantly improved:

```
******** VACL  ***********
1 redirect 4/25 tcp host 192.168.1.67 host 255.255.255.255
2 redirect 4/25 udp host 192.168.1.67 host 255.255.255.255
3 permit ip any any
*******  Cisco IOS ACL ***********
1  deny ip any host 239.255.255.255
2  permit ip any any
*******  MERGE **********
has 4 entries
```

Estimating Merge Results with Supervisor Engine Software Releases 7.1(1) or Later Releases

In supervisor engine software releases prior to software release 7.1(1), the following formula is true for software release 7.1(1) and later releases: The size of ACL C = (size of ACL A) x (size of ACL B) x (2).

Note In software releases prior to release 7.1(1), the formula is used as a guideline but the number of entries could go beyond the predicted range. In software release 7.1(1) and later releases, with the new ACL merge algorithm, the formula is accurate for all cases. If Layer 4 port information is specified, the upper limit could be higher even with the new algorithm. See the "Layer 4 Operations Configuration Guidelines" section on page 15-23 for detailed information.

Two ACL-merge algorithms are available—the binary decision diagram (BDD) and the order dependent merge (ODM). ODM is the enhanced algorithm that was introduced in software release 7.1(1). The BDD algorithm was used in releases prior to software release 7.1(1). See the "Specifying the ACL-Merge Algorithm" section on page 15-47 for detailed software configuration information.

Note With software release 8.1(1) and later releases, the BDD algorithm is no longer supported on any platform (PFC, PFC2, or PFC3A/PFC3B/PFC3BXL). The default ACL-merge algorithm is ODM. In software release 8.1(1) and later releases, the following command changes appear: The **set aclmerge algo** and **set aclmerge bdd** commands have been removed. The **show aclmerge** {**bdd** | **algo**} command has been reduced to **show aclmerge algo**.

Examples

These examples show the merge results for the various Cisco IOS ACL and VACL configurations. One VACL and one Cisco IOS ACL are configured on the same VLAN.

Example 1

```
******** VACL  ***********
1  permit udp host 194.72.72.33 194.72.6.160 0.0.0.15
2  permit udp host 147.150.213.94 194.72.6.64 0.0.0.15 eq bootps
3  permit udp 194.73.74.0 0.0.0.255 host 194.72.6.205 eq syslog
4  permit udp host 167.221.23.1 host 194.72.6.198 eq tacacs
5  permit udp 194.72.136.1 0.0.3.128 194.72.6.64 0.0.0.15 eq tftp
6  permit udp host 193.6.65.17 host 194.72.6.205 gt 1023
7  permit tcp any host 194.72.6.52
8  permit tcp any host 194.72.6.52 eq 113
9  deny tcp any host 194.72.6.51 eq ftp
10 permit tcp any host 194.72.6.51 eq ftp-data
11 permit tcp any host 194.72.6.51
12 permit tcp any eq domain host 194.72.6.51
13 permit tcp any host 194.72.6.51 gt 1023
14 permit ip  any host 1.1.1.1
******** Cisco IOS ACL ************
1  deny ip any host 239.255.255.255
2  permit ip any any
*******  MERGE **********
Using the new algorithm - 17 entries
Using the old algorighm - 91 entries
```

Example 2

```
******** VACL  **********
1  permit udp host 194.72.72.33 194.72.6.160 0.0.0.15
2  permit udp host 147.150.213.94 194.72.6.64 0.0.0.15 eq bootps
3  permit udp 194.73.74.0 0.0.0.255 host 194.72.6.205 eq syslog
4  permit udp host 167.221.23.1 host 194.72.6.198 eq tacacs
5  permit udp 194.72.136.1 0.0.3.128 194.72.6.64 0.0.0.15 eq tftp
6  permit udp host 193.6.65.17 host 194.72.6.205 gt 1023
7  permit tcp any host 194.72.6.52
8  permit tcp any host 194.72.6.52 eq 113
9  permit tcp any host 194.72.6.51 eq ftp-data
10 permit tcp any host 194.72.6.51 neq ftp
11 permit tcp any eq domain host 194.72.6.51 neq ftp
12 permit tcp any host 194.72.6.51 gt 1023
13 permit ip  any host 1.1.1.1
******** Cisco IOS ACL ************
1  deny ip any host 239.255.255.255
2  permit ip any any
******** MERGE ***********
Using the new algorithm - 16 entries
Using the old algorithm - 78 entries
```

Example 3

```
******** VACL  ***********
1  deny ip 0.0.0.0 255.255.255.0 any
2  deny ip 0.0.0.255 255.255.255.0 any
3  deny ip any 0.0.0.0 255.255.255.0
4  permit ip any host 239.255.255.255
5  permit ip any host 255.255.255.255
6  deny ip any 0.0.0.255 255.255.255.0
7  permit tcp any range 0 65534 any range 0 65534
8  permit udp any range 0 65534 any range 0 65534
9  permit icmp any any
10 permit ip any any
******** Cisco IOS ACL **********
1  deny ip any host 239.255.255.255
2  permit ip any any
******** MERGE **********
Using the new algorithm - 12 entries
Using the old algorithm - 303 entries
```

Example 4

```
******** VACL  ***********
1 redirect 4/25 tcp host 192.168.1.67 host 255.255.255.255
2 redirect 4/25 udp host 192.168.1.67 host 255.255.255.255
3 deny tcp any any lt 30
4 deny udp any any lt 30
5 permit ip any any
*******  Cisco IOS ACL ***********
1  deny ip any host 239.255.255.255
2  permit ip any any
*******  MERGE **********

Using the new algorithm - 6 entries
Using the old algorithm - 142 entries
```

Example 5

```
******** VACL  ***********
1 redirect 4/25 tcp host 192.168.1.67 host 255.255.255.255
2 redirect 4/25 udp host 192.168.1.67 host 255.255.255.255
3 permit ip any any
*******  Cisco IOS ACL ***********
1  deny ip any host 239.255.255.255
2  permit ip any any
*******  MERGE **********

Using the new algorithm - 4 entries
Using the old algorithm - 4 entries
```

Layer 4 Operations Configuration Guidelines

These sections provide the guidelines for using Layer 4 port operations:

- Determining Layer 4 Operation Usage, page 15-24
- Determining Logical Operation Unit Usage, page 15-24

Determining Layer 4 Operation Usage

The switch hardware allows you to specify these types of operations:

- gt (greater than)
- lt (less than)
- neq (not equal)
- eq (equal)
- range (inclusive range)

We recommend that you do not specify more than nine *different* operations on the same ACL. If you exceed this number, each new operation might cause the affected ACE to be translated into more than one ACE.

Note If you have a Cisco IOS ACL and a VACL on the same VLAN interface, the recommended total number of Layer 4 operations is still nine or less.

Use the following two guidelines to determine the Layer 4 operation usage:

1. Layer 4 operations are considered different if the operator or the operand differ. In this ACL, there are four different Layer 4 operations ("gt 10" and "gt 11" are considered two different Layer 4 operations):

```
... gt 10 permit
... lt 9 deny
... gt 11 deny
... neq 6 redirect
```

Note There is no limit to the use of "eq" operators, because the "eq" operator does not use a logical operator unit (LOU) or a Layer 4 operation bit. See the "Determining Logical Operation Unit Usage" section on page 15-24 for a description of LOUs.

2. Layer 4 operations are considered different if the same operator/operand couple applies once to a source port and once to a destination port. In this ACL, there are two different Layer 4 operations because one ACE applies to the source port and one applies to the destination port.

```
... Src gt 10 ...
... Dst gt 10
```

Note Check the ACL Layer 4 port operations resource usage using the **show security acl resource-usage** command.

Determining Logical Operation Unit Usage

The LOUs are registers that store the operator/operand couples. All the ACLs use the LOUs. There can be up to 32 LOUs; each LOU can store two different operator/operand couples with the exception of the range operator. The LOU usage per Layer 4 operation is as follows:

- gt uses 1/2 LOU
- lt uses 1/2 LOU
- neq uses 1/2 LOU

- range uses 1 LOU
- eq does not require a LOU

For example, this ACL would use a single LOU to store two different operator/operand couples:

```
... Src gt 10 ...
... Dst gt 10
```

A more detailed example is as follows:

```
ACL1
... (dst port) gt 10 permit
... (dst port) lt 9 deny
... (dst port) gt 11 deny
... (dst port) neq 6 redirect
... (src port) neq 6 redirect
... (dst port) gt 10 deny

ACL2
... (dst port) gt 20 deny
... (src port) lt 9 deny
... (src port) range 11 13 permit
... (dst port) neq 6 redirect
```

The Layer 4 operations and LOU usage are as follows:

- ACL1 Layer 4 operations: 5
- ACL2 Layer 4 operations: 4
- LOUs: 4

An explanation of the LOU usage is as follows:

- LOU 1 stores “gt 10” and “lt 9”
- LOU 2 stores “gt 11” and “neq 6”
- LOU 3 stores “gt 20” (with space for one more)
- LOU 4 stores “range 11 13” (range needs the entire LOU)

Using VACLs in Your Network

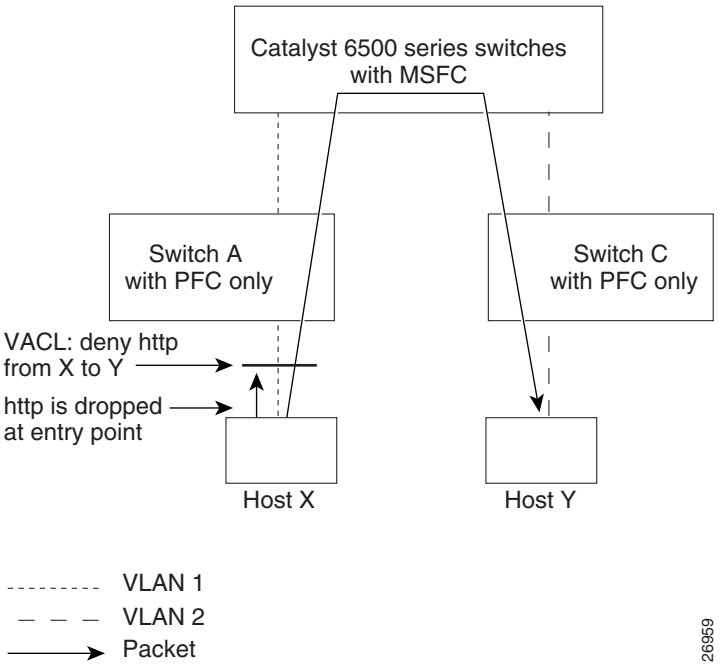
These sections describe some typical uses for the VACLs:

Wiring Closet Configuration

In a wiring closet configuration, Catalyst 6500 series switches might not be equipped with the MSFCs (routers). In this configuration, the switch can still support a VACL and a QoS ACL. Suppose that Host X and Host Y are in different VLANs and are connected to wiring closet Switch A and Switch C (see Figure 15-4). The traffic from Host X to Host Y is eventually being routed by the switch that is equipped with the MSFC. The traffic from Host X to Host Y can be access controlled at the traffic entry point, Switch A.

If you do not want the HTTP traffic that is switched from Host X to Host Y, you can configure a VACL on Switch A. All HTTP traffic from Host X to Host Y would be dropped at Switch A and not be bridged to the switch with the MSFC.

Figure 15-4 Wiring Closet Configuration



Redirecting Broadcast Traffic to a Specific Server Port

Some application traffic uses the broadcast packets that reach every host in a VLAN. With the VACLs, you can redirect these broadcast packets to the intended application server port.

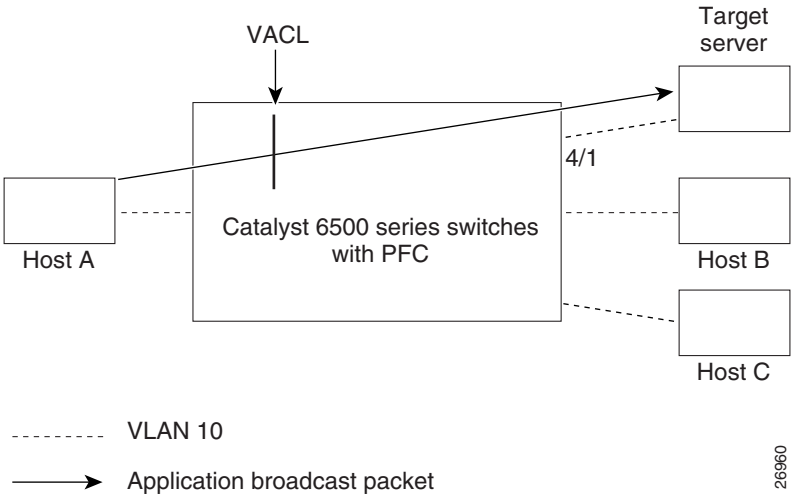
Figure 15-5 shows an application broadcast packet from Host A being redirected to the target application server port and preventing other ports from receiving the packet.

To redirect the broadcast traffic to a specific server port, perform this task in privileged mode (TCP port 5000 is the intended server application port):

	Task	Command
Step 1	Redirect the broadcast packets.	**set security acl ip SERVER redirect 4/1 tcp any host 255.255.255.255 eq 5000**
Step 2	Permit all other traffic.	**set security acl ip SERVER permit ip any any**
Step 3	Commit the VACL.	**commit security acl SERVER**
Step 4	Map the VACL to VLAN 10.	**set security acl map SERVER 10**

Note You could apply the same concept to direct the broadcast traffic to a multicast destination by redirecting the traffic to a group of ports (see Figure 15-5).

Figure 15-5 Redirecting Broadcast Traffic to a Specific Server Port



Restricting the DHCP Response for a Specific Server

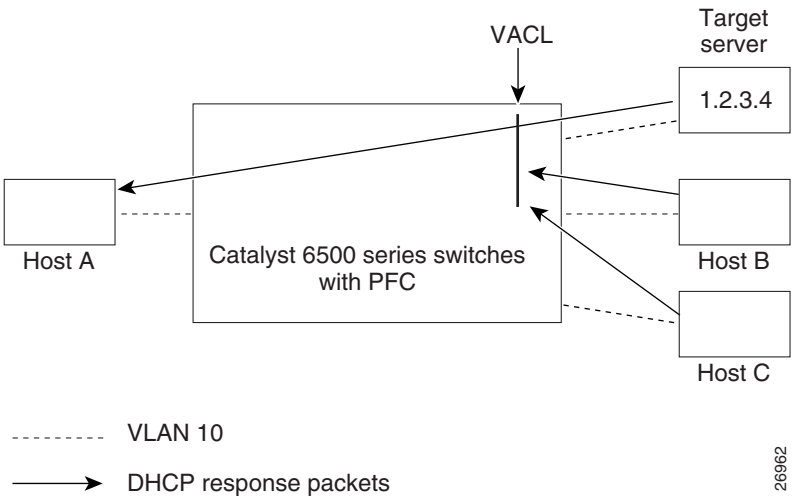
When the Dynamic Host Configuration Protocol (DHCP) requests are broadcast, they reach every DHCP server in the VLAN and multiple responses are returned. With the VACLs, you can restrict the response from a *specific* DHCP server and drop the other responses.

To restrict the DHCP responses for a specific server, perform this task in privileged mode (the target DHCP server IP address is 1.2.3.4):

	Task	Command
Step 1	Permit a DHCP response from host 1.2.3.4.	**set security acl ip SERVER permit udp host 1.2.3.4 any eq 68**
Step 2	Deny the DHCP responses from any other host.	**set security acl ip SERVER deny udp any any eq 68**
Step 3	Permit the other IP traffic.	**set security acl ip SERVER permit any**
Step 4	Commit the VACL.	**commit security acl SERVER**
Step 5	Map the VACL to VLAN 10.	**set security acl map SERVER 10**

Figure 15-6 shows that only the target server returns a DHCP response from the DHCP request.

Figure 15-6 Redirecting a DHCP Response for a Specific Server



Denying Access to a Server on Another VLAN

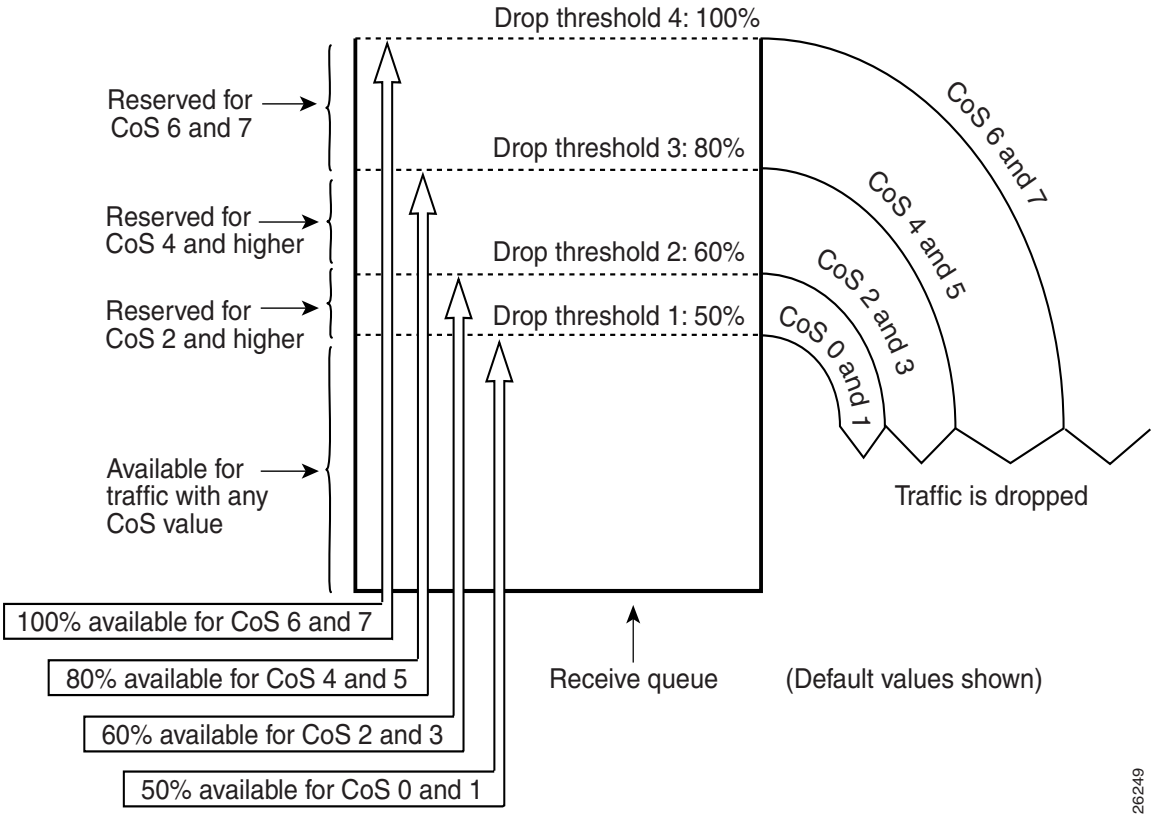
You can restrict access to a server on another VLAN. For example, server 10.1.1.100 in VLAN 10 needs to have access restricted as follows (seeFigure 15-7):

- Hosts in subnet 10.1.2.0/24 in VLAN 20 should not have access.
- Hosts 10.1.1.4 and 10.1.1.8 in VLAN 10 should not have access.

To deny access to a server on another VLAN, perform this task in privileged mode:

	Task	Command
Step 1	Deny traffic from hosts in subnet 10.1.2.0/8.	**set security acl ip SERVER deny ip 10.1.2.0 0.0.0.255 host 10.1.1.100**
Step 2	Deny traffic from host 10.1.1.4.	**set security acl ip SERVER deny ip host 10.1.1.4 host 10.1.1.100**
Step 3	Deny traffic from host 10.1.1.8.	**set security acl ip SERVER deny ip host 10.1.1.8 host 10.1.1.100**
Step 4	Permit the other IP traffic.	**set security acl ip SERVER permit ip any any**
Step 5	Commit the VACL.	**commit security acl SERVER**
Step 6	Map the VACL to VLAN 10.	**set security acl map SERVER 10**

Figure 15-7 Denying Access to a Server on Another VLAN



Restricting ARP Traffic

Note This feature is available only with Supervisor Engine 2 with PFC2, Supervisor Engine 720 with PFC3A/PFC3B/PFC3BXL, and Supervisor Engine 32 with PFC3B/PFC3BXL.

The ARP traffic is permitted on each VLAN by default. You can disallow the ARP traffic on a per-VLAN basis using the **set security acl ip** *acl_name* **deny arp** command. When you enter this command, the ARP traffic is disallowed on the VLAN to which the ACL is mapped. To allow the ARP traffic on a VLAN that has had the ARP traffic disallowed, enter the **set security acl ip** *acl_name* **permit arp** command.

Inspecting ARP Traffic

Note This feature is available only with Supervisor Engine 2 with PFC2, Supervisor Engine 720 with PFC3A/PFC3B/PFC3BXL, and Supervisor Engine 32 with PFC3B/PFC3BXL.

These sections describe the ARP traffic-inspection feature:

Overview

ARP is a simple protocol that does not have an authentication mechanism, so there is no way to ensure that the ARP requests and replies are genuine. Without an authentication mechanism, a malicious user/host can corrupt the ARP tables of the other hosts on the same VLAN in a Layer 2 network or bridge domain.

For example, user/Host A (the malicious user) can send unsolicited ARP replies (or gratuitous ARP packets) to the other hosts on the subnet with the IP address of the default router and the MAC address of Host A. With some earlier operating systems, even if a host already has a static ARP entry for the default router, the newly advertised binding from Host A is learned. If Host A enables IP forwarding and forwards all packets from the "spoofed" hosts to the router and vice versa, then Host A can carry out a man-in-the-middle attack (for example, using the program dsniff) without the spoofed hosts realizing that all of their traffic is being sniffed.

ARP traffic inspection allows you to configure a set of order-dependent rules within the security ACL (VACL) framework to prevent the ARP table attacks.

Implementation

If a specific rule in ARP traffic inspection exists in the VACL on a VLAN, all ARP packets are index-directed to the CPU through the ACEs in the VACL. The packets are inspected by the ARP traffic-inspection task for conformance to the specified rules. The conforming packets are forwarded while the nonconforming packets are dropped and logged (if logging is enabled).

The rules for ARP traffic inspection specify the ARP bindings for the specified IP addresses as shown in the example that follows:

```
permit arp-inspection host 10.0.0.1 00-00-00-01-00-02
permit arp-inspection host 20.0.0.1 00-00-00-02-00-03
deny arp-inspection host 10.0.0.1 any
deny arp-inspection host 20.0.0.1 any
permit arp-inspection any any
```

The above set of rules allows only 00-00-00-01-00-02 to be advertised as the MAC address for IP address 10.0.0.1. Similarly, MAC address 00-00-00-02-00-03 is bound to IP address 20.0.0.1. The ARP packets that advertise any other MAC addresses for 10.0.0.1 and 20.0.0.1 are dropped (achieved by the **deny** actions in lines 3 and 4). All other ARP packets are allowed to go through (achieved by the **permit** action in line 5).

ARP Traffic-Inspection Configuration Guidelines

This section describes the guidelines for configuring ARP traffic inspection:

- The ARP traffic-inspection clauses appear at the top of a VACL.
- The maximum number of ARP traffic-inspection clauses that can be configured in a VACL is 128.
- An ARP traffic-inspection ACE cannot be modified to become an IP ACE and vice versa.
- An ARP traffic-inspection ACE cannot be inserted before an IP ACE and vice versa.
- Do not use the generic deny/permit clauses with the ARP traffic-inspection clauses in the same VACL. The generic ARP deny/permit clauses are installed using the **set security acl ip** *acl_name* {**deny** | **permit**} **arp** command.
- If the MSFC is the gateway for the hosts, you must allow the MSFC IP/MAC binding. We recommend that the gateway IP/MAC binding be allowed when using ARP traffic inspection.
- ARP traffic inspection uses the existing logging facility for the VACLs. After a packet traverses the ARP traffic-inspection rules, if the result is a "permit," the packet is forwarded to the destination MAC address (or broadcast address). If the result is a "deny," the packet is dropped and sent to the VACL logging process if logging is enabled.

  VACL logging uses the source MAC address and the following fields from the ARP header to define a logging flow: source IP address, source MAC address, and ARP opcode (request, reply).

  You can limit the number of logged flows by entering the **set security acl log maxflow** *max_flows* command. However, the **set security acl log ratelimit** *max_rate* command does not apply to the ARP traffic inspection logged flows.
- The RARP packets are not used to learn the ARP entries on the hosts and are harmless from an ARP corruption perspective. The PFC2 and PFC3A/PFC3B/PFC3BXL do not distinguish between the ARP and RARP packets. An ACE that is used to redirect the ARP packets to the CPU also redirects the RARP packets. Global rate limiting is a rate limit for the ARP and RARP packets combined. The ARP traffic-inspection rules do not apply to the RARP packets; the RARP packets are simply forwarded. A generic ARP deny clause also denies the RARP packets. You can display the number of RARP packets that are forwarded by entering the **show security acl arp-inspection statistics** command.
- Mapping VACLs with the ARP traffic-inspection clauses to the management VLANs (sco/sc1 interfaces) is supported.
- Even if a port is part of an EtherChannel, the drop and shutdown thresholds remain port based. The thresholds are not part of the *match* that is required for the formation of an EtherChannel (after PAgP identifies the *matched* EtherChannel links, it groups the ports into an EtherChannel).
- Due to the way the hardware recognizes the ARP packets, the IP packets with source address 0.0.0.0, destination address 0.0.0.0, and the IP protocol ICMP, are also redirected to the ARP traffic-inspection task. Because these packets are invalid, they are dropped. The count of these packets is displayed as part of the **show security acl arp-inspection statistics** command.
- If the syslog messages are generated for every packet that is dropped by the ARP traffic-inspection task, the console is overwhelmed with messages. To avoid this situation, only 40 syslog messages are allowed per minute.

- This example shows you how to avoid a common configuration error. The following is a typical ARP traffic-inspection ACL:

```
------------------------------
set security acl ip my_arp
------------------------------------------------------
arp permit
1. permit arp-inspection host 10.6.62.86 00-b0-c2-3b-db-fd
2. deny arp-inspection host 10.6.62.86 any
3. permit arp-inspection any any
---------------------------
```

  This ACL ensures that only MAC address 00-b0-c2-3b-db-fd is advertised as the MAC address for IP address 10.6.62.86. This ACL will deny *all* IP packets because there is an implicit ip deny any any in an IP ACL.

  If you want all IP traffic to pass through, there should be an explicit permit ip any any at the end of the ACL as follows:

```
--------------------
set security acl ip my_arp
------------------------------------------------------
arp permit
1. permit arp-inspection host 10.6.62.86 00-b0-c2-3b-db-fd
2. deny arp-inspection host 10.6.62.86 any
3. permit arp-inspection any any
4. permit ip any any
----------------------
```

- This example shows a typical configuration using ARP traffic inspection. The following ACL is used to protect the two IP addresses that are specified and will not do ARP traffic inspection with any MAC addresses other than those specified:

```
set security acl ip ACL_VLAN951 permit arp-inspection host 132.216.251.129
00-d0-b7-11-13-14
set security acl ip ACL_VLAN951 deny arp-inspection host 132.216.251.129 any log
set security acl ip ACL_VLAN951 permit arp-inspection host 132.216.251.250
00-d0-00-ea-43-fc
set security acl ip ACL_VLAN951 deny arp-inspection host 132.216.251.250 any log
set security acl ip ACL_VLAN951 permit arp-inspection any any
set security acl ip ACL_VLAN951 permit ip any any
```

ARP Traffic-Inspection Configuration Procedures

These sections describe the ARP traffic-inspection configuration procedures:

Configuring ARP Traffic Inspection

Configuring Rate Limiting for ARP Traffic Inspection

Configuring Logging for ARP Traffic Inspection

Permitting or Denying ARP Packets Advertising a Specific IP-Address-to-MAC-Address Binding

To permit or deny the ARP packets that advertise a binding for a specific IP address and MAC address, perform this task in privileged mode:

	Task	Command
Step 1	Permit or deny the ARP packets that advertise a binding for a specific IP address and MAC address.	**set security acl ip** *acl_name* {**permit** | **deny**} **arp-inspection host** *ip_address mac_address*
Step 2	Commit the VACL.	**commit security acl** {**acl_name** | **all** | **adjacency**}

This example shows how to permit the ARP packets that advertise a binding of IP address 172.20.52.54 to MAC address 00-01-64-61-39-c2:

```
Console> (enable) set security acl ip ACL1 permit arp-inspection host 172.20.52.54
00-01-64-61-39-c2
Operation successful.
Console> (enable) commit security acl ACL1
Console> (enable) ACL commit in progress.


ACL 'ACL1' successfully committed.
```

Permitting or Denying ARP Packets Advertising a Particular IP Address Binding

To permit or deny the ARP packets that advertise a binding for the specified IP address, perform this task in privileged mode:

	Task	Command
Step 1	Permit or deny the ARP packets that advertise a binding for the specified IP address.	**set security acl ip** *acl_name* {**permit** | **deny**} **arp-inspection host** *ip_address* **any**
Step 2	Commit the VACL.	**commit security acl** {**acl_name** | **all** | **adjacency**}

This example shows how to permit the ARP packets that advertise a binding of IP address 172.20.52.19:

```
Console> (enable) set security acl ip ACL2 permit arp-inspection host 172.20.52.19 any
Operation successful.
Console> (enable) commit security acl ACL2
Console> (enable) ACL commit in progress.

ACL 'ACL2' successfully committed.
```

Permitting or Denying All ARP Packets

To permit or deny all ARP packets, perform this task in privileged mode:

	Task	Command
Step 1	Permit or deny all ARP packets.	**set security acl ip** *acl_name* {**permit** | **deny**} **arp-inspection any any**
Step 2	Commit the VACL.	**commit security acl** {**acl_name** | **all** | **adjacency**}

This example shows how to permit all ARP packets:

```
Console> (enable) set security acl ip ACL3 permit arp-inspection any any
Operation successful.
Console> (enable) commit security acl ACL3
Console> (enable) ACL commit in progress.

ACL 'ACL3' successfully committed.
```

Permitting or Denying ARP Packets that Advertise Bindings for IP Addresses on a Particular Network

To permit or deny the ARP packets that advertise a binding for the IP addresses on a particular network, perform this task in privileged mode:

Note The *ip_mask* is a reverse mask. The "0" bit means "match" and the "1" bit means "ignore." For example, 10.3.5.6 and 0.0.0.255 are equivalent to 10.3.5/24.

	Task	Command
Step 1	Permit or deny the ARP packets that advertise a binding for the IP addresses on a particular network.	**set security acl ip** *acl_name* {**permit** | **deny**} **arp-inspection** *ip_address ip_mask* **any**
Step 2	Commit the VACL.	**commit security acl** {*acl_name* | **all** | **adjacency**}

This example shows how to permit the ARP packets that advertise a binding for the IP addresses on the 10.3.5.0/24 subnet:

```
Console> (enable) set security acl ip ACL4 permit arp-inspection 10.3.5.6 0.0.0.255 any
Operation successful.
Console> (enable) commit security acl ACL4
Console> (enable) ACL commit in progress.
```

```
ACL 'ACL4' successfully committed.
```

Dropping Packets Without Matching MAC Addresses

To drop the packets where the source Ethernet MAC address (in the Ethernet header) is not the same as the source MAC address in the ARP header, perform this task in privileged mode. If you do not specify the **drop** keyword, the packet is not dropped but a syslog message is displayed. Use the **log** keyword to send the packets to the VACL logging facility.

Tip In most cases, using the **match-mac** clause to prevent ARP spoofing does not negate the need to create a specific ARP-inspection ACL for each VLAN. The **match-mac** clause does not catch the more sophisticated ARP table attacks. Most ARP spoofers change the source MAC address in the Ethernet header to match the address in the ARP payload.

	Task	Command
Step 1	Identify or drop the packets without the matching MAC addresses.	**set security acl arp-inspection match-mac** {**enable** [**drop** [**log**]] | **disable**}
Step 2	Commit the VACL.	**commit security acl** {*acl_name* | **all** | **adjacency**}
Step 3	Display the configuration.	**show security acl arp-inspection config**

This example shows how to drop the packets where the source Ethernet MAC address is not the same as the source MAC address in the ARP header:

```
Console> (enable) set security acl arp-inspection match-mac enable drop
ARP Inspection match-mac feature enabled with drop option.
Console> (enable)

Console> (enable) show security acl arp-inspection config
Match-mac feature is enabled with drop option.
Address-validation feature is disabled.
Dynamic ARP Inspection is disabled on vlan(s) 1.
Dynamic ARP Inspection is disabled on ports 5/1-48,7/1-2.
Logging for Dynamic ARP Inspection rules is disabled.
Console> (enable)
```

Dropping Packets with Invalid MAC or IP Addresses

The following MAC addresses are invalid:

- 00-00-00-00-00-00
- Multicast MAC addresses (the 48th bit is set)
- ff-ff-ff-ff-ff-ff (this is a special-case multicast MAC address)

The following IP addresses are invalid:

- 0.0.0.0
- 255.255.255.255
- Class D (multicast) IP addresses

To drop the packets with invalid MAC or IP addresses, perform this task in privileged mode (if you do not specify the **drop** keyword, the packet is not dropped but a syslog message is displayed):

	Task	Command
Step 1	Drop the packets with the invalid MAC or IP addresses.	**set security acl arp-inspection address-validation** {**enable** [**drop** [**log**]] | **disable**}
Step 2	Commit the VACL.	**commit security acl** {*acl_name* | **all** | **adjacency**}
Step 3	Display the configuration.	**show security acl arp-inspection config**

This example shows how to drop the packets with the invalid MAC or IP addresses:

```
Console> (enable) set security acl arp-inspection address-validation enable drop
ARP Inspection address-validation feature enabled with drop option.
Console> (enable)

Console> (enable) show security acl arp-inspection config
Address-validation feature is enabled with drop option.
Console> (enable)
```

Displaying ARP Traffic-Inspection Statistics

To display the number of packets that are permitted and denied by the ARP traffic-inspection task, perform this task in normal mode:

Task	Command
Display the number of packets that are permitted and denied by the ARP traffic-inspection task.	**show security acl arp-inspection statistics** [*acl_name*]

Note You can enter the **show security acl** commands to display certain ARP traffic-inspection configuration information.

This example shows how to display the number of packets that are permitted and denied by the ARP traffic-inspection task:

```
Console> (enable) show security acl arp-inspection statistics
ARP Inspection statistics
Packets forwarded = 0
Packets dropped = 0
RARP packets (forwarded) = 0
Packets for which Match-mac failed = 0
Packets for which Address Validation failed = 0
IP packets dropped = 0
Console> (enable)
```

Clearing the ARP Traffic-Inspection Statistics

To clear the ARP traffic-inspection statistics, perform this task in privileged mode:

Task	Command
Clear the ARP traffic-inspection statistics.	**clear security acl arp-inspection statistics** [*acl_name*]

Without the optional argument, entering the command clears the ARP traffic-inspection global statistics counters and the ARP traffic-inspection statistics counters for all the ACLs. If you supply the optional *acl_name* argument, only the ARP traffic-inspection statistics for that particular ACL are cleared.

Note You can enter the **clear security acl** commands to clear the ARP traffic-inspection configuration settings.

Configuring Rate Limiting on a Global Basis

You can rate limit the number of ARP traffic-inspection packets that are sent to the supervisor engine CPU globally. By default, the ARP traffic-inspection traffic is rate limited to 500 packets per second. The minimum value is 500, and the maximum value is 2000 packets per second. For Supervisor Engine 720, the minimum value that is enforced by the hardware is 10 packets per second (values between 1– 9 are set to 10). To disable rate limiting, set the value to 0.

Note Rate limiting might be shared by multiple features. To display the features that share rate limiting, enter the **show security acl feature ratelimit** command.

To rate limit the number of ARP traffic-inspection packets that are sent to the CPU on a global basis, perform this task in privileged mode:

	Task	Command
Step 1	Rate limit the number of ARP traffic-inspection packets that are sent to the supervisor engine CPU on a global basis.	**set security acl feature ratelimit** *rate*
Step 2	Display the global rate-limit value.	**show security acl feature ratelimit**
Step 3	Display all the rate-limiter settings that are configured on the switch processor and the route processor.	**show rate-limit**

This example shows how to rate limit the number of ARP traffic-inspection packets that are sent to the CPU to 1000:

```
Console> (enable) set security acl feature ratelimit 1000
Dot1x DHCP and ARP Inspection global rate limit set to 1000 pps.
Console> (enable)

Console> (enable) show security acl feature ratelimit
Rate limit value in packets per second = 1000
Protocols set for rate limiting = Dot1x DHCP, ARP Inspection
Console> (enable)
```

```
Console> (enable) show rate-limit
Configured Rate Limiter Settings:

Rate Limiter Type    Status  Rate (pps)     Burst
-------------------- ------  -------------- -----
VACL LOG             On      2500           1
ARP INSPECTION       On      1000           1
FIB RECEIVE          Off     *              *
FIB GLEAN            Off     *              *
L3 SEC FEATURES      Off     *              *

Console> (enable)
```

Configuring Rate Limiting on a Per-Port Basis

You can rate limit the number of ARP traffic-inspection packets that are sent to the supervisor engine CPU on a per-port basis. If the rate exceeds the **drop-threshold**, the excess packets are dropped (and counted toward the **shutdown-threshold** limit). If the rate exceeds the **shutdown-threshold**, the port that is specified by *mod/port* is shut down. By default, both threshold values are 0 (no per-port rate limiting is applied). The maximum value for both thresholds is 1000 packets-per second (pps).

To rate limit the number of ARP traffic-inspection packets that are sent to the CPU per port, perform this task in privileged mode:

	Task	Command
Step 1	Rate limit the number of ARP traffic-inspection packets that are sent to the supervisor engine CPU on a per-port basis.	**set port arp-inspection** *mod/port* **drop-threshold** *packets_per_second* **shutdown-threshold** *packets_per_second* **set port arp-inspection** *mod/port* **drop-threshold** *packets_per_second* **set port arp-inspection** *mod/port* **shutdown-threshold** *packets_per_second*
Step 2	Display the drop and shutdown thresholds.	**show port arp-inspection** {[*mod/port*] | [*mod*]}

This example shows how to rate limit the number of ARP traffic-inspection packets that are sent to the CPU on a per-port basis. The drop-threshold is set to 700, and the shutdown threshold is set to 800 for port 3/1:

```
Console> (enable) set port arp-inspection 3/1 drop-threshold 700 shutdown-threshold 800
Drop Threshold=700, Shutdown Threshold=800 set on port 3/1.
Console> (enable)

Console> (enable) show port arp-inspection 3/1
Port                      Drop Threshold Shutdown Threshold
------------------------  -------------- ------------------
 3/1                              700                800
Console> (enable)
```

Configuring the errdisable-timeout Option for ARP Traffic Inspection

You configure the errdisable-timeout option for ARP traffic inspection by using the **set errdisable-timeout** {**enable** | **disable**} **arp-inspection** command. For detailed information on the errdisable-timeout option, see the "Configuring a Timeout Period for Ports in errdisable State" section on page 4-12.

Configuring Logging for ARP Traffic Inspection

To configure the logging option to log the ARP traffic-inspection packets that are dropped, perform this task in privileged mode:

Task	Command
Log the ARP traffic-inspection packets that are dropped.	**set security acl ip** *acl_name* **deny arp-inspection** {**host** *ip_address* {**any** | *mac_address*} | *ip_address ip_mask* **any** | **any any**} [**log**]

For detailed information on the VACL logging option, see the "Configuring VACL Logging" section on page 15-59. This section also provides information on limiting the number of logged flows using the **set security acl log maxflow** *max_number* command.

To display the logged ARP traffic-inspection packets, perform this task in normal mode:

Task	Command
Display the logged ARP traffic-inspection packets.	**show security acl log flow arp** [**host** *ip_address* [**vlan** *vlan*]]

If you specify the optional **host** *IP address*, only the ARP packets that advertise a binding for the specified host IP address are displayed. If you specify the optional **vlan** *vlan* keyword and argument, the search is restricted to the specified VLAN.

Dynamic ARP Inspection

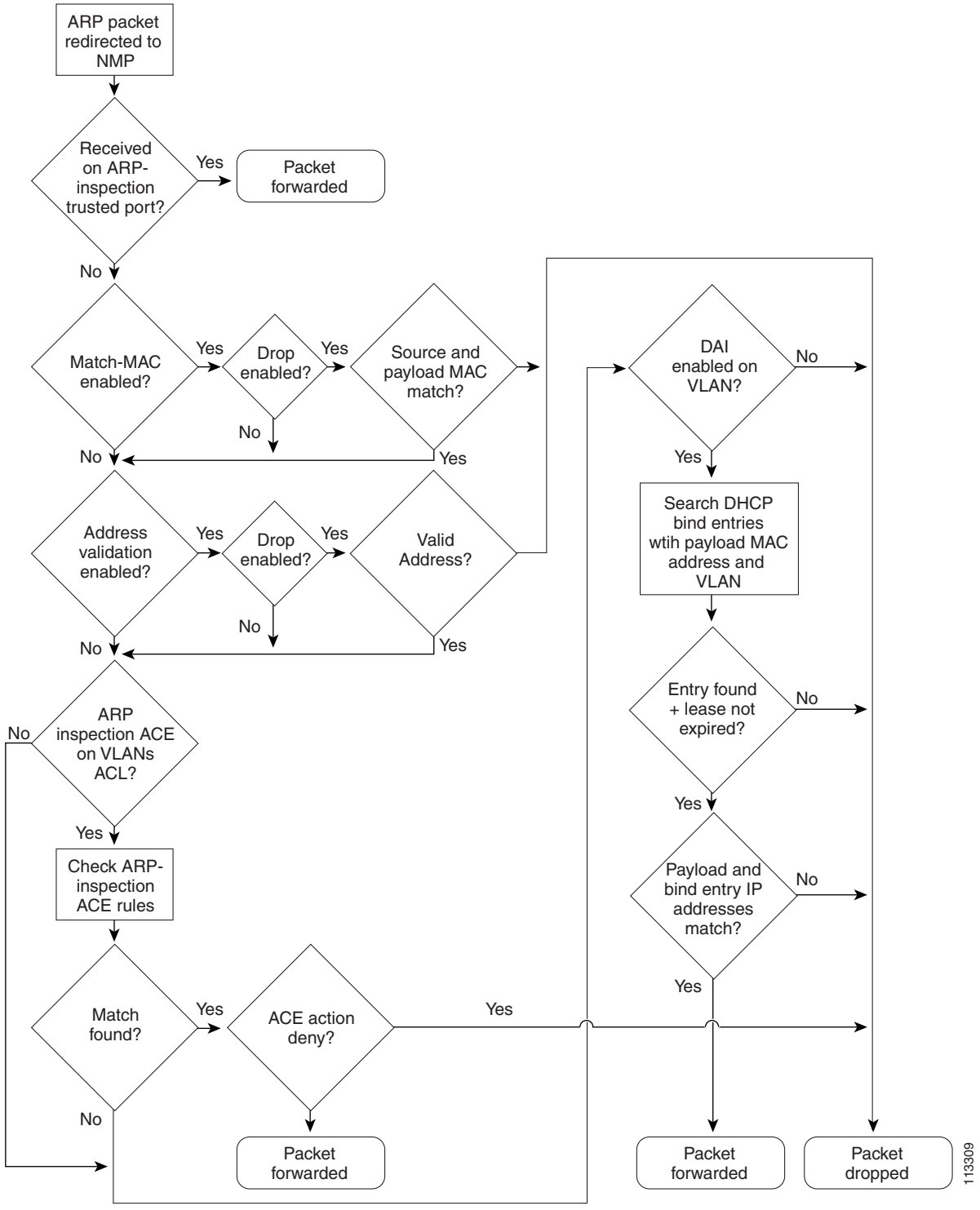
Note Dynamic ARP inspection (DAI) is available only with Supervisor Engine 2 with PFC2, Supervisor Engine 720 with PFC3A/PFC3B/PFC3BXL, and Supervisor Engine 32 with PFC3B/PFC3BXL.

These sections describe DAI:

Overview

DAI uses the binding information that is built by DHCP snooping to enforce the advertisement of bindings to prevent "man-in-the-middle" attacks. These attacks can occur when an attacker intercepts and selectively modifies communicated data to masquerade as one or more of the entries in a communication association. DAI adds an extra layer of security to ARP inspection by verifying that the ARP packet's MAC address and IP address match an existing DHCP snooping binding in the same VLAN. The basic functionality and packet flow of ARP inspection remains unchanged except for the addition of checks to ensure that a DHCP binding exists (see Figure 15-8 for a logical flow chart).

Figure 15-8 Dynamic ARP Inspection Flow Chart



Note Only the ARP packets that are sent from an untrusted port are inspected. The ARP packets that are received from a trusted port are forwarded without inspection (this process applies to both static and dynamic ARP inspection). By default, the system configures the MSFC port as ARP inspection trusted.

When you create a security ACL, you need to be careful because the statically configured ARP inspection rules have a higher priority than the DAI checks of the DHCP bindings. Do not put a **permit arp-inspection any any** clause in the security ACL because it will prevent any checks from occurring.

You can enable or disable DAI on a per-VLAN basis. If you configure the DAI ports as untrusted, you must also configure them as DHCP-snooping untrusted ports. You should enable DHCP snooping in all VLANs that have DAI enabled. Optionally, you can enable logging for the ARP packets that are denied by DAI.

Note DAI works best when enabled on VLANs where all (or most) of the IP address assignment is done using DHCP.

If the static IP address assignments exist in a VLAN, you must configure the relevant ports as ARP inspection-trusted ports or you must configure the static ARP inspection nubs to permit these MAC and IP addresses.

Dynamic ARP Inspection Configuration Procedures

Note We recommend that you enable high availability when using DAI, DHCP snooping, and IP source guard. If high availability is not enabled, the clients have to renew their IP addresses for these features to work after a switchover. For the configuration details on DHCP snooping and IP source guard, see Chapter 33, "Configuring DHCP Snooping and IP Source Guard."

Note Prior to software release 8.6(1), you could enable dynamic ARP inspection only on VLANs. In software release 8.6(1) and later releases, you can enable dynamic ARP inspection on a per-port basis.

Before you configure DAI, you will need to enable dynamic ARP inspection on a per-port basis. Perform this task in privileged mode:

	Task	Command
Step 1	Make the security ACL mode port-based on the host port.	**set port security-acl** *mod/port* **port-based**
Step 2	Enable DAI on the port using the CLI.	**set security acl arp-inspection dynamic enable port** *mod/port*
Step 3	Display the security ACL ARP inspection configuration.	**show security acl arp-inspection config**
Step 4	Create an ACL using the **permit arp-inspection any any** command (to redirect ARP packets to the software).	**set security acl ip dai permit dhcp-snooping** **set security acl ip dai permit arp-inspection any any** **set security acl ip dai permit ip any any** **commit security acl dai**
Step 5	Map the ACL to the host port.	**set security acl map dai** *mod/port*

Note To make sure DAI ports function properly, a permit arp-inspection any any ACE should be present in the PACL (ACL mapped to a DAI-enabled port).

Note For DAI to function with hosts that have static IP, make sure to add static DHCP-snooping binding entries on the port instead of a static ARP-inspection rule in the PACL (ACL mapped to a DAI-enabled port).

This example shows how to enable dynamic ARP on port 1/48:

```
Console> (enable) set port security-acl 1/48 port-based
Warning: Vlan-based ACL features will be disabled on ports 1/48
ACL interface is set to port-based mode for port(s) 1/48.
Console> (enable) set security acl arp-inspection dynamic enable port 1/48
Dynamic ARP Inspection enabled on port 1/48.
Console> (enable) show security acl arp-inspection config
Match-mac feature is disabled.
Address-validation feature is disabled.
Dynamic ARP Inspection is disabled on vlan(s) 1-20,50.
Dynamic ARP Inspection is enabled on ports 1/48.
Dynamic ARP Inspection is disabled on ports 1/1-47,4/1-48,5/1-2.
Logging for Dynamic ARP Inspection rules is disabled.
Console> (enable) set security acl ip dai permit dhcp-snooping
Successfully configured DHCP Snooping for ACL dai. Use 'commit' command to save
changes.
Console> (enable) set security acl ip dai permit arp-inspection any any
dai editbuffer modified. Use 'commit' command to apply changes.
Console> (enable) set security acl ip dai permit ip any any
dai editbuffer modified. Use 'commit' command to apply changes.
Console> (enable) commit security acl dai
Console> (enable) ACL commit in progress.
ACL 'dai' successfully committed.
Console> (enable) set security acl map dai 1/48
Mapping in progress.
```

To configure DAI, perform this task in privileged mode:

	Task	Command
Step 1	Enable DAI on a VLAN.	**set security acl arp-inspection dynamic** {**enable** | **disable**} [*vlanlist* | **port** *mod/port*]
Step 2	Enable or disable the inspection of the ARP packets.	**set port arp-inspection** *portlist* **trust** {**enable** | **disable**}
Step 3	Enable logging of the packets denied by DAI. **Note** Logging of static ARP rule denials is still controlled by the rule (ACE) CPG.	**set security acl arp-inspection dynamic log** {**enable** | **disable**}
Step 4	Verify the DAI and DAI logging configuration.	**show security acl arp-inspection config**

This example shows how to enable DAI on VLAN 100:

```
Console> (enable) set security acl arp-inspection dynamic enable 100
```

```
Dynamic ARP Inspection is enabled for vlan(s) 100.
Console> (enable) set port arp-inspection 2/2 trust enable
Port(s)  2/2 state set to trusted for ARP Inspection.
Console> (enable) set security acl arp-inspection dynamic log enable
Dynamic ARP Inspection logging enabled.
Console> show security acl arp-inspection config
Match-mac feature is disabled.
Address-validation feature is disabled.
Dynamic ARP Inspection is disabled on vlan(s) 1,1006-1013.
Dynamic ARP Inspection is enabled on vlan(s) 100.
Logging for Dynamic ARP Inspection rules is enabled.
Console>
```

Configuring ACLs on Private VLANs

Private VLANs allow you to split a primary VLAN into sub-VLANs (secondary VLANs) that can be either community VLANs or isolated VLANs. In releases prior to software release 6.1(1), you could configure ACLs on a primary VLAN only and the ACL would then be applied to all the secondary VLANs. In software release 6.1(1) and later releases, ACLs can be applied as follows:

- You can map VACLs to secondary VLANs or primary VLANs.
- Cisco IOS ACLs that are mapped to a primary VLAN get mapped to the associated secondary VLANs.
- You cannot map Cisco IOS ACLs to secondary VLANs.
- You cannot map dynamic ACEs to a private VLAN.
- You can map QoS ACLs to secondary VLANs or primary VLANs.

If you map a VACL to a primary VLAN, it filters the traffic from the router to the host and if you map a VACL to a secondary VLAN, it filters the traffic from the host to the router.

Note With software release 6.2(1) and later releases, you can use two-way community VLANs to perform an inverse mapping from the primary VLAN to the secondary VLAN when the traffic crosses the boundary of a private VLAN through a promiscuous port. Both the outbound and inbound traffic can be carried on the same VLAN allowing VLAN-based VACLs to be applied in both directions on a per-community (per-customer) basis.

Note For additional information on private VLANs, see the "Configuring Private VLANs on the Switch" section on page 11-19.

Capturing Traffic Flows

See the "Capturing Traffic Flows on Specified Ports" section on page 15-57 for complete configuration details.

Unsupported Features

Note With Supervisor Engine 720 (PFC3A/PFC3B/PFC3BXL) and Supervisor Engine 32 (PFC3B/PFC3BXL), the IPX routing is done through the software and IPX Cisco IOS ACLs and IPX VACLs are not supported. You can match the IPX packets using the MAC VACLs. You can enter the **ipx-arpa** keyword to match the IPX ARPA frames. Use 0xffff EtherType to match on the IPX non-ARPA frames and frames with an EtherType of 0xffff. For information on configuring the MAC VACLs, see the "Creating a Non-IP Version 4/Non-IPX VACL (MAC VACL) and Adding ACEs" section on page 15-52.

This section lists the ACL-related features that are not supported or have limited support on the Catalyst 6500 series switches:

- Non-IP version 4/non-IPX Cisco IOS ACLs—The following types of Cisco IOS security ACLs cannot be enforced on the switch in the hardware; the MSFC has to process the ACL in the software and this *significantly* degrades system performance:
  - Bridge-group ACLs
  - IP accounting
  - Inbound and outbound rate limiting
  - Standard IPX with source node number
  - IPX extended access lists that specify a source node number or socket numbers are not enforced in the hardware
  - Standard XNS access list
  - Extended XNS access list
  - DECnet access list
  - Extended MAC address access list
  - Protocol type-code access list
- IP packets with a header length of less than five will not be access controlled.
- Non full-flow IPX VACL—IPX VACL is based on a flow that is specified by a source/destination network number, packet type, and destination node number only. The source node number and socket number are not supported when specifying the IPX flow.

Configuring VACLs

This section describes how to configure the VACLs. Prior to performing any configuration tasks, see the "VACL Configuration Guidelines" section on page 15-45.

These sections provide the guidelines and a summary for configuring the VACLs:

VACL Configuration Guidelines

This section describes the guidelines for configuring the VACLs:

Caution All changes to the ACLs are stored temporarily in an edit buffer. You must enter the **commit** command to commit all the ACEs to NVRAM. The committed ACLs with no ACEs are deleted. We recommend that you enter the ACEs in batches and enter the **commit** command to save all the changes to NVRAM.

Note You can configure Cisco IOS ACLs and VACLs from flash memory instead of NVRAM. See the "Configuring and Storing VACLs and QoS ACLs in Flash Memory" section on page 15-64 for detailed information.

Note With Supervisor Engine 720 (PFC3A/PFC3B/PFC3BXL) and Supervisor Engine 32 (PFC3B/PFC3BXL), the IPX routing is done through the software and IPX Cisco IOS ACLs and IPX VACLs are not supported. You can match the IPX packets using the MAC VACLs. You can enter the **ipx-arpa** keyword to match the IPX ARPA frames. Use 0xffff EtherType to match on the IPX non-ARPA frames and frames with an EtherType of 0xffff. For information on configuring the MAC VACLs, see the "Creating a Non-IP Version 4/Non-IPX VACL (MAC VACL) and Adding ACEs" section on page 15-52.

- See the "Configuring Cisco IOS ACLs and VACLs on the Same VLAN Interface Guidelines" section on page 15-17.
- See the "Using VACLs in Your Network" section on page 15-25 for configuration examples.
- See the "Unsupported Features" section on page 15-44.
- See the "Specifying the ACL-Merge Algorithm" section on page 15-47.
- You must commit a VACL before you can map it to a VLAN. There are no default VACLs and no default VACL-to-VLAN mappings.
- If no Cisco IOS ACL is configured to deny the traffic on a routed VLAN interface (input or output), and *no* VACL is configured, all traffic is permitted.
- The order of ACEs in an ACL is important. A packet that comes into the switch is applied against the first ACE in the ACL. If there is no match, the packet is applied against the next ACE in the list. If no ACEs match, the packet is denied (dropped).
- Always enter the **show security acl info** *acl_name* **editbuffer** command to see the *current* list of ACEs before making any changes to the edit buffer.
- In systems with redundant MSFCs, the ACL configurations for Cisco IOS ACLs and VACLs must be the same on both MSFCs.
- The system might incorrectly calculate the maximum number of ACLs in the system if an ACL is deleted but not committed.
- The **show security acl resource-usage** and **show qos acl resource-usage** commands might not show 100 percent usage even if there is no space in the hardware to store more ACLs. This situation occurs because some ACL space is reserved in the hardware for the ACL manager to perform cleanup and mapping if necessary.
- The system might take longer to boot if you configure a very large number of ACLs.

- Note these guidelines for using the redirect option:
  - The redirected packets can only go out a port that supports the VLAN that the traffic is in.
  - The redirect option only involves taking the packets and sending them out the redirect port; there is no routing involved.
  - If the packets are coming in from many VLANs, the redirect port should have those VLANs in the forwarding state. You might have to configure the redirect port as a trunk to allow multiple VLANs to go out of the port.
  - Put caches in promiscuous mode so they can receive traffic that is not routed.
  - Use the redirect option to do some basic VLAN-based load balancing by redirecting the traffic to multiple ports. Each port transmits only those packets that belong to the VLANs that are forwarding on the port.

VACL Configuration Summary

To create a VACL and map it to a VLAN, perform these steps:

Step 1 Enter the **set security acl ip** command to create a VACL and add ACEs.

Step 2 Enter the **commit** command to commit the VACL and its associated ACEs to NVRAM.

Step 3 Enter the **set security acl map** command to map the VACL to a VLAN.

Note An IP VACL is used in this description; you can configure IPX and non-IP version 4/non-IPX VACLs using the same basic steps.

Note The VACLs have an implicit deny feature at the end of the list; a packet is denied if it does not match any VACL ACE.

Configuring VACLs from the CLI

This section describes how to create and activate the VACLs on the Catalyst 6500 series switches. These tasks are listed in the order that they should be performed.

This section describes the following tasks:

Specifying the ACL-Merge Algorithm

Two ACL-merge algorithms are available—the binary decision diagram (BDD) and the order dependent merge (ODM). ODM is the enhanced algorithm that was introduced in software release 7.1(1). The BDD algorithm was used in the releases prior to software release 7.1(1). With ODM, after the merge, the resultant ACEs are order dependent. With BDD, after the merge, the resultant ACEs are order independent.

Note With software release 8.1(1) and later releases, the BDD algorithm is no longer supported on any platform (PFC, PFC2, or PFC3A/PFC3B/PFC3BXL). The default ACL-merge algorithm is ODM. In software release 8.1(1) and later releases, the following command changes appear: The **set aclmerge algo** and **set aclmerge bdd** commands have been removed. The **show aclmerge** {**bdd** | **algo**} command has been reduced to **show aclmerge algo**.

Note For examples of the ODM algorithm, see the "Estimating Merge Results with Supervisor Engine Software Releases 7.1(1) or Later Releases" section on page 15-21.

The default algorithm is ODM. If BDD is disabled, the merge algorithm can only be ODM. When BDD is enabled, you can choose either the BDD algorithm or the ODM algorithm. You must enable BDD to change the ACL merge algorithm. Use the **set aclmerge bdd** command to enable or disable BDD. When you enable or disable BDD, the change takes effect when your system is restarted.

Caution Enabling BDD on a supervisor engine with 64-MB DRAM could cause memory to run low. To avoid this situation, upgrade the memory to 128 MB or disable BDD.

The ACL merge algorithm that you select is in effect for all new ACL merges. The ACLs that are already configured are not modified and use the ACL merge algorithm that was enabled when the ACLs were merged.

To enable or disable BDD, perform this task in privileged mode:

	Task	Command
Step 1	Enable or disable BDD.	**set aclmerge bdd** {**enable** | **disable**}
Step 2	Display the current BDD status and whether BDD will be enabled or disabled at the next system restart.	**show aclmerge** {**bdd** | **algo**}

This example shows how to disable BDD:

```
Console> (enable) set aclmerge bdd disable
Bdd will be disabled on system restart.
Console> (enable)
```

This example shows how to display the current BDD status and whether BDD will be enabled or disabled at the next system restart:

```
Console> (enable) show aclmerge bdd
Bdd is not enabled.
On system restart bdd will be disabled.
Console> (enable)
```

To specify the ACL-merge algorithm, perform this task in privileged mode:

	Task	Command
Step 1	Specify the ACL-merge algorithm.	**set aclmerge algo** {**bdd** | **odm**}
Step 2	Display the ACL-merge algorithm that is currently in use.	**show aclmerge** {**bdd** | **algo**}

This example shows how to specify the ODM algorithm:

```
Console> (enable) set aclmerge algo odm
Acl merge algorithm set to odm.
Console> (enable)
```

This example shows the ACL-merge algorithm that is currently in use:

```
Console> (enable) show aclmerge algo
Current acl merge algorithm is odm.
Console> (enable)
```

Creating an IP VACL and Adding ACEs

To create a new IP VACL and add the ACEs, or to add the ACEs to an existing IP VACL, perform one of these tasks in privileged mode:

Task	Command
If an IP protocol specification is not required, use the following syntax.	**set security acl ip** {*acl_name*} {**permit** | **deny**} {*src_ip_spec*} [**capture**] [**before** *editbuffer_index* | **modify** *editbuffer_index*] [**log**[1]]
If an IP protocol is specified, use the following syntax.	**set security acl ip** {*acl_name*} {**permit** | **deny** | **redirect** *mod_num/port_num*} {*protocol*} {*src_ip_spec*} {*dest_ip_spec*} [**precedence** *precedence*] [**tos** *tos*] [**capture**] [**before** *editbuffer_index* | **modify** *editbuffer_index*] [**log**[1]]

1. The **log** keyword provides logging messages for denied IP VACLs only.

This example shows how to create an ACE for IPACL1 to allow the traffic from source address 172.20.53.4:

```
Console> (enable) set security acl ip IPACL1 permit host 172.20.53.4 0.0.0.0
IPACL1 editbuffer modified. Use 'commit' command to apply changes.
Console> (enable)
```

Note Because the VACLs have an implicit deny feature at the end of the list, *all* other traffic is denied.

This example shows how to create an ACE for IPACL1 to allow the traffic from all source addresses:

```
Console> (enable) set security acl ip IPACL1 permit any
IPACL1 editbuffer modified. Use 'commit' command to apply changes.
Console> (enable)
```

This example shows how to create an ACE for IPACL1 to block the traffic from source address 171.3.8.2:

```
Console> (enable) set security acl ip IPACL1 deny host 171.3.8.2
IPACL1 editbuffer modified.  Use 'commit' command to apply changes.
Console> (enable)
```

This example shows how to display the contents of the edit buffer:

```
Console> (enable) show security acl info IPACL1 editbuffer
set security acl ip IPACL1
------------------------------------------------------------------
1. permit ip host 172.20.53.4 any
2. permit ip any any
3. deny ip host 171.3.8.2 any
Console> (enable)
```

This example shows how to commit the ACEs to NVRAM:

```
Console> (enable) commit security acl all
ACL commit in progress.
ACL IPACL1 is committed to hardware.
Console> (enable)
```

Note For more information about the **commit security acl all** command, see the "Committing ACLs" section on page 15-53.

Enter the **show security acl info IPACL1** command to verify that the changes were committed. If this VACL has not been mapped to a VLAN, enter the **set security acl map** command to map it to a VLAN.

This example shows how to create an ACE for IPACL2 to block the traffic from source address 172.20.3.2 and place this ACE before ACE number 2 in the VACL. Optionally, you can enter the **modify** keyword to replace an existing ACE with a new ACE. Enter the **show security acl info** *acl_name* [**editbuffer**] command to see the current ACE listing that is stored in NVRAM (enter the **editbuffer** keyword to see edit buffer contents).

```
Console> (enable) set security acl ip IPACL2 deny host 172.20.3.2 before 2
IPACL2 editbuffer modified. Use 'commit' command to apply changes.
Console> (enable)
```

This example shows how to create an ACE for IPACL2 to redirect IP traffic to port 3/1 from source address 1.2.3.4 with the destination address of 255.255.255.255. The host can be used as an abbreviation for a source and source-wildcard of 0.0.0.0. This ACE also specifies the following:

- **precedence**—IP precedence values that range between zero for low priority and seven for high priority.
- **tos**—Type of service levels that range between 0 and 15.

Note The ToS values are bits 3 through 6 of the IP ToS byte as defined by RFC 1349. The precedence values are bits 0 through 2 as defined by RFC 791.

```
Console> (enable) set security acl ip IPACL2 redirect 3/1 ip 1.2.3.4 0.0.0.255 host
255.255.255.255 precedence 1 tos min-delay
IPACL2 editbuffer modified. Use 'commit' command to apply changes.
Console> (enable)
```

This example shows how to display the contents of the edit buffer:

```
Console> (enable) show security acl info IPACL2 editbuffer
set security acl ip IPACL2
------------------------------------------------------------------
1. deny 172.20.3.2
2. redirect 1.2.3.4
Console> (enable)
```

Note For more information about the **show security acl info** command, see the "Displaying the Contents of a VACL" section on page 15-54.

This example shows how to commit the ACEs to NVRAM:

```
Console> (enable) commit security acl all
ACL commit in progress.
ACL IPACL2 is committed to hardware.
Console> (enable)
```

Note For more information about the **commit security acl all** command, see the "Committing ACLs" section on page 15-53.

Enter the **show security acl info IPACL2** command to verify that the changes were committed. If this VACL has not been mapped to a VLAN, enter the **set security acl map** command to map it to a VLAN.

Creating an IPX VACL and Adding ACEs

Note With Supervisor Engine 720 (PFC3A/PFC3B/PFC3BXL) and Supervisor Engine 32 (PFC3B/PFC3BXL), the IPX routing is done through the software and the IPX Cisco IOS ACLs and IPX VACLs are not supported. You can match the IPX packets using the MAC VACLs. You can enter the **ipx-arpa** keyword to match the IPX ARPA frames. Use 0xffff EtherType to match on the IPX non-ARPA frames and frames with an EtherType of 0xffff. For information on configuring the MAC VACLs, see the "Creating a Non-IP Version 4/Non-IPX VACL (MAC VACL) and Adding ACEs" section on page 15-52.

To create a new IPX VACL and add the ACEs, or to add the ACEs to an existing IPX VACL, perform this task in privileged mode:

Task	Command
Create a new IPX VACL and add the ACEs, or add the ACEs to an existing IPX VACL.	**set security acl ipx** {*acl_name*} {**permit** | **deny** | **redirect** *mod_num/port_num*} {*protocol*} {*src_net*} [*dest_net.*[*dest_node*] [[*dest_net_mask.*]*dest_node_mask*]] [**capture**] [**before** *editbuffer_index* **modify** *editbuffer_index*]

This example shows how to create an ACE for IPXACL1 to block all traffic from source network 1234:

```
Console> (enable) set security acl ipx IPXACL1 deny any 1234
IPXACL1 editbuffer modified. Use 'commit' command to apply changes.
Console> (enable)
```

This example shows how to create an ACE for IPXACL1 to block all traffic with destination address 1.A.3.4:

```
Console> (enable) set security acl ipx IPXACL1 deny any any 1.A.3.4
IPXACL1 editbuffer modified. Use 'commit' command to apply changes.
Console> (enable)
```

This example shows how to create an ACE for IPXACL1 to redirect broadcast traffic to port 4/1 from source network 3456:

```
Console> (enable) set security acl ipx IPXACL1 redirect 4/1 any 3456
IPXACL1 editbuffer modified. Use 'commit' command to apply changes.
Console> (enable)
```

This example shows how to display the contents of the edit buffer:

```
Console> (enable) show security acl info IPXACL1 editbuffer
set security acl ipx IPXACL1
------------------------------------------------------------------
1. deny any 1234
2. deny any any 1.A.3.4
3. redirect 4/1 any 3456
Console> (enable)
```

Note For more information about the **show security acl info** command, see the "Displaying the Contents of a VACL" section on page 15-54.

This example shows how to commit the ACEs to NVRAM:

```
Console> (enable) commit security acl all
ACL commit in progress.
ACL IPXACL1 is committed to hardware.
Console> (enable)
```

Enter the **show security acl info IPXACL1** command to verify that the changes were committed. If this VACL has not been mapped to a VLAN, enter the **set security acl map** command to map it to a VLAN.

This example shows how to create an ACE for IPXACL1 to allow all traffic from source network 1 and insert this ACE before ACE number 2:

```
Console> (enable) set security acl ipx IPXACL1 permit any 1 before 2
IPXACL1 editbuffer modified. Use 'commit' command to apply changes.
Console> (enable)
```

This example shows how to create an ACE for IPXACL1 to allow the traffic from all source addresses:

```
Console> (enable) set security acl ipx IPXACL1 permit any any
IPXACL1 editbuffer modified. Use 'commit' command to apply changes.
Console> (enable)
```

This example shows how to display the contents of the edit buffer:

```
Console> (enable) show security acl info IPXACL1 editbuffer
set security acl ipx IPXACL1
------------------------------------------------------------------
1. deny any 1234
```

```
2. permit any 1
3. deny any any 1.A.3.4
4. redirect 4/1 any 3456
5. permit any any
ACL IPXACL1 Status: Not Committed
Console> (enable)
```

This example shows how to commit the ACEs to NVRAM:

```
Console> (enable) commit security acl all
ACL commit in progress.
ACL IPXACL1 is committed to hardware.
Console> (enable)
```

Note For more information about the **commit security acl all** command, see the "Committing ACLs" section on page 15-53.

Enter the **show security acl info IPXACL1** command to verify that the changes were committed. If this VACL has not been mapped to a VLAN, enter the **set security acl map** command to map it to a VLAN.

Creating a Non-IP Version 4/Non-IPX VACL (MAC VACL) and Adding ACEs

Caution The IP and IPX traffic are not access controlled by the MAC VACLs. All other traffic types (AppleTalk, DECnet, and so on) are classified as the MAC traffic and the MAC VACLs are used to access control this traffic.

To create a new non-IP version 4/non-IPX VACL and add the ACEs, or to add the ACEs to an existing non-IP version 4/non-IPX VACL, perform this task in privileged mode:

Task	Command
Create a new non-IP version 4/non-IPX VACL and add the ACEs, or add the ACEs to an existing non-IP version 4/non-IPX VACL.	**set security acl mac** {*acl_name*} {**permit** | **deny**} {*src_mac_addr_spec*} {*dest_mac_addr_spec*} [*ethertype*] [**capture**] [**before** *editbuffer_index* | **modify** *editbuffer_index*]

This example shows how to create an ACE for MACACL1 to block all traffic from 8-2-3-4-7-A:

```
Console> (enable) set security acl mac MACACL1 deny host 8-2-3-4-7-A any
MACACL1 editbuffer modified. Use 'commit' command to apply changes.
Console> (enable)
```

This example shows how to create an ACE for MACACL1 to block all traffic to A-B-C-D-1-2:

```
Console> (enable) set security acl mac MACACL1 deny any host A-B-C-D-1-2
MACACL1 editbuffer modified. Use 'commit' command to apply changes.
Console> (enable)
```

This example shows how to create an ACE for MACACL1 to allow the traffic from all sources:

```
Console> (enable) set security acl mac MACACL1 permit any any
MACACL1 editbuffer modified. Use 'commit' command to apply changes.
Console> (enable)
```

This example shows how to display the contents of the edit buffer:

```
Console> (enable) show security acl info MACACL1 editbuffer
set security acl mac MACACL1
-----------------------------------------------------------------
1. deny 8-2-3-4-7-A any
2. deny any A-B-C-D-1-2
3. permit any any
Console> (enable)
```

Note For more information about the **show security acl info** command, see the "Displaying the Contents of a VACL" section on page 15-54.

This example shows how to commit the ACEs to NVRAM:

```
Console> (enable) commit security acl all
ACL commit in progress.
ACL MACACL1 is committed to hardware.
Console> (enable)
```

Note For more information about the **commit security acl all** command, see the "Committing ACLs" section on page 15-53.

Enter the **show security acl info MACACL1** command to verify that the changes were committed. If this VACL has not been mapped to a VLAN, enter the **set security acl map** command to map it to a VLAN.

Committing ACLs

You can commit all ACLs or a specific ACL to NVRAM with the **commit** command. Any committed ACL with no ACEs will be deleted.

To commit an ACL to NVRAM, perform this task in privileged mode:

Task	Command
Commit an ACL to NVRAM.	**commit security acl** *acl_name* | **all**

This example shows how to commit a specific security ACL to NVRAM:

```
Console> (enable) commit security acl IPACL2
ACL commit in progress.
ACL IPACL2 is committed to hardware.
Console> (enable)
```

Mapping a VACL to a VLAN

You can map a VACL to a VLAN with the **set security acl map** command. Note that there is no default ACL-to-VLAN mapping; all VACLs need to be mapped to a VLAN.

To map a VACL to a VLAN, perform this task in privileged mode:

Task	Command
Map a VACL to a VLAN.	**set security acl map** *acl_name vlans*

This example shows how to map IPACL1 to VLAN 10:

```
Console> (enable) set security acl map IPACL1 10
ACL IPACL1 mapped to vlan 10
Console> (enable)
```

This example shows the output if you try to map an ACL that has not been committed:

```
Console> (enable) set security acl map IPACL1 10
Commit ACL IPACL1 before mapping.
Console> (enable)
```

Displaying the Contents of a VACL

You can display the contents of a VACL with the **show security acl info** command.

To display the contents of a VACL, perform this task in privileged mode:

Task	Command
Display the contents of a VACL.	**show security acl info** {*acl_name* | **all**} [**editbuffer** [*editbuffer_index*]]

This example shows how to display the contents of a VACL that has been saved in NVRAM:

```
Console> (enable) show security acl info IPACL1
set security acl ip IPACL1
---------------------------------------------------------------------
1. deny A
2. deny ip B any
3. deny c
4. permit any
```

This example shows how to display the contents of a VACL that is still in the edit buffer:

```
Console> (enable) show security acl info IPACL1 editbuffer
set security acl ip IPACL1
---------------------------------------------------------------------
1. deny A
2. deny ip B any
3. deny C
4. deny D
5. permit any
Console> (enable)
```

Displaying a VACL-to-VLAN Mapping

You can display a VACL-to-VLAN mapping for a specified ACL or VLAN with the **show security acl map** command.

To display a VACL-to-VLAN mapping, perform this task in privileged mode:

Task	Command
Display a VACL-to-VLAN mapping.	**show security acl map** {*acl_name* | *vlan* | **all**}

This example shows how to display the mappings of a specific VACL:

```
Console> (enable) show security acl map IPACL1
ACL IPACL1 is mapped to VLANs:
1
Console> (enable)
```

This example shows how to display the mappings of a specific VLAN:

```
Console> (enable) show security acl map 1
VLAN 1 is mapped to IP ACL IPACL1.
VLAN 1 is mapped to IPX ACL IPXACL1.
VLAN 1 is mapped to MAC ACL MACACL1.
Console> (enable)
```

Clearing the Edit Buffer

You can clear the changes made to the ACL edit buffer since its last save with the **rollback** command. The ACL is rolled back to its state at the last **commit** command.

To clear the ACL edit buffer, perform this task in privileged mode:

Task	Command
Clear the ACL edit buffer.	**rollback security acl** {*acl_name* | **all** | **adjacency**}

This example shows how to clear the edit buffer of a specific security ACL:

```
Console> (enable) rollback security acl IPACL1
Editbuffer for 'IPACL1' rolled back to last commit state.
Console> (enable)
```

Removing ACEs from Security ACLs

You can remove a specific ACE or all ACEs from an ACL with the **clear security acl** command. This command deletes the ACEs from the edit buffer.

To remove an ACE from a security ACL, perform this task in privileged mode:

Task	Command
Remove an ACE from a security ACL.	**clear security acl all** **clear security acl** *acl_name* **clear security acl** *acl_name editbuffer_index*

This example shows how to remove the ACEs from all the ACLs:

```
Console> (enable) clear security acl all
All editbuffers modified. Use 'commit' command to apply changes.
Console> (enable)
```

This example shows how to remove a specific ACE from a specific ACL:

```
Console> (enable) clear security acl IPACL1 2
IPACL1 editbuffer modified. Use 'commit' command to apply changes.
Console> (enable)
```

Clearing the Security ACL Map

You can remove a VACL-to-VLAN mapping with the **clear security acl map** command.

To clear the security ACL map, perform this task in privileged mode:

Task	Command
Clear the security ACL map.	**clear security acl map all** **clear security acl map** *acl_name* **clear security acl map** *vlan* **clear security acl map** *acl_name vlan*

This example shows how to clear all VACL-to-VLAN mappings:

```
Console> (enable) clear security acl map all
Map deletion in progress.

Successfully cleared mapping between ACL ip1 and VLAN 10.

Successfully cleared mapping between ACL ipx1 and VLAN 10.

.... display text omitted
Console> (enable)
```

This example shows how to clear the mapping for a specific VACL on a specific VLAN:

```
Console> (enable) clear security acl map IPACL1 50
Map deletion in progress.

Successfully cleared mapping between ACL ipacl1 and VLAN 50.
Console> (enable)
```

Displaying VACL Management Information

You can display VACL management information with the **show security acl resource-usage** command.

To display VACL management information, perform this task in privileged mode:

Task	Command
Display VACL management information.	**show security acl resource-usage**

This example shows how to display VACL management information:

```
Console> (enable) show security acl resource-usage
ACL resource usage:
ACL storage (mask/value): 0.29%/0.10%
ACL to switch interface mapping table: 0.39%
ACL layer 4 port operators: 0.0%
Console (enable)
```

Capturing Traffic Flows on Specified Ports

You can enter the **capture** keyword in the **set security acl** (**ip**, **ipx**, and **mac**) commands to specify that the packets that match the specified flows are captured and transmitted out of the capture ports. You can specify the capture ports using the **set security acl capture-ports** *mod/ports...* command. When you use the **capture** keyword, the packets that match the specified flows are captured in parallel and transmitted out of the capture ports. The capture ports do not send out all the captured traffic; they send out only the traffic belonging to the VLANs of the captured port.

Configuration Guidelines

This section describes the guidelines for configuring the capture ports:

- The capture port cannot be part of an EtherChannel.
- The capture port cannot be an ATM port.
- The capture port must be in the spanning-tree forwarding state for the VLAN.
- You can specify any number of switch ports as capture ports. The capture ports are added to a capture port list, and the configuration is saved in NVRAM.
- Only permit traffic is captured. If a packet is dropped due to an ACL, the packet cannot be captured.
- The capture ports do not transmit out all captured traffic. They transmit only traffic belonging to the capture port VLAN. To capture the traffic going to many VLANs, the capture port should be a trunk carrying the required VLANs.

  For the routed traffic, the capture ports transmit the packets only after they are Layer 3 switched; the packets are transmitted out of a port only if the output VLAN of the Layer 3-switched flow is the same as the capture port VLAN. For example, assume that you have flows from VLAN 10 to VLAN 20, you add a VACL on one of the VLANs permitting these flows, and you specify a capture port. This traffic gets transmitted out of the capture port only if it belongs to VLAN 20 or if the port is a trunk carrying VLAN 20. If the capture port is in VLAN 10, it does not transmit any traffic. Whether a capture port transmits the traffic or not is independent of the VLAN on which you placed the VACL.

  If you want to capture the traffic from one VLAN going to many VLANs, the capture port has to be a trunk carrying all the output VLANs.

  For the bridged traffic, because all the traffic remains in the same VLAN, ensure that the capture port is in the same VLAN as the bridged traffic.
- To capture the traffic, you can configure one ACL and map it to a group of VLANs or you can configure a number of ACLs and map each to one VLAN. Configure as many ACEs per ACL as necessary to capture the desired traffic.

To capture the traffic flows, perform these steps:

Note An IP VACL is used in this description; you can configure IPX and non-IP version 4/non-IPX VACLs using the same basic steps.

Step 1 Enter the **set security acl ip** command to create a VACL and add the ACEs; include the **capture** keyword.

Step 2 Enter the **commit** command to commit the VACL and its associated ACEs to NVRAM.

Step 3 Enter the **set security acl map** command to map the VACL to a VLAN.

Step 4 Enter the **set security acl capture-ports** *mod/ports...* command to specify the capture ports.

Configuration Examples

This example shows how to create an ACE for my_cap and specify that the allowed traffic is captured:

```
Console> (enable) set security acl ip my_cap permit ip host 60.1.1.1 host 60.1.1.98
capture
my_cap editbuffer modified. Use 'commit' command to apply changes.
Console> (enable)
```

This example shows how to commit the my_cap ACL to NVRAM:

```
Console> (enable) commit security acl my_cap
ACL commit in progress.

ACL my_cap successfully committed.
Console> (enable)
```

This example shows how to map my_cap to VLAN 10:

```
Console> (enable) set security acl map my_cap 10
Mapping in progress.

VLAN 10 successfully mapped to ACL my_cap.
The old mapping with ACL captest was replaced with the new one.
Console> (enable)
```

This example shows how to specify the capture ports:

```
Console> (enable) set security acl capture-ports 1/1-2,2/1-2
Successfully set the following ports to capture ACL traffic:
1/1-2,2/1-2
Console> (enable)
```

This example shows how to display the ports that have been specified as the capture ports:

```
Console> (enable) show security acl capture-ports
ACL Capture Ports: 1/1-2,2/1-2
Console> (enable)
```

This example shows how to clear the capture ports:

```
Console> (enable) clear security acl capture-ports 1/1,2/1
Successfully cleared the following ports:
1/1,2/1
Console> (enable)
```

This example shows that ports 1/1 and 2/1 were cleared:

```
Console> (enable) show security acl capture-ports
ACL Capture Ports:1/2,2/2
Console> (enable)
```

Configuring VACL Logging

Note This feature is available only with Supervisor Engine 2 with PFC2, Supervisor Engine 720 with PFC3A/PFC3B/PFC3BXL, and Supervisor Engine 32 with PFC3B/PFC3BXL.

You can log the messages about the denied packets for the standard IP access list by entering the **log** keyword for the deny VACLs. Any packet that matches the access list causes an informational logging message about the packet to be sent to the console. The level of messages that is logged to the console is controlled by the **set logging level acl** *severity* command.

The first packet that triggers the access list causes a logging message right away, and the subsequent packets are collected over 5-minute intervals before they are displayed or logged. The logging message includes the flow pattern and the number of packets that are received in the past 5 minutes.

By default, the system logging messages are sent to the console. You can configure the switch to send the system logging messages to a syslog server. For information on configuring system message logging, see Chapter 29, "Configuring System Message Logging."

Configuration Guidelines

This section describes the guidelines for configuring VACL logging:

- Log only the deny traffic from the IP VACLs.
- You must set the logging level to 6 (information) or 7 (debugging).

To enable VACL logging, perform these steps:

Step 1 Enter the **set logging level acl** *severity* command to set the logging level to 6 (information) or 7 (debugging).

Step 2 (Optional) Enter the **set security acl log maxflow** *max_number* to allocate a new log table that is based on the maximum flow pattern number to store the logged packet information. If successful, the new buffer replaces the old one and all flows in the old table are cleared. If either memory is not enough or the maximum number is over the limit, an error message is displayed and the command is dropped. The valid values are from 256 to 2048; the default value is 500.

Note If the maximum flow pattern is over the max_num limit, an error message is displayed and the command is dropped. The messages are not logged for these packets.

Step 3 (Optional) Enter the **set security acl log ratelimit** *pps* command to set the redirect rate in pps (packets per second). If the configuration is over the range, the command is discarded and the range is displayed on the console. The valid values are from 500 to 5000; the default value is 2500. To disable rate limiting, set the value to 0.

Note If the redirect rate is over the pps range, the command is dropped and the range is displayed on the console. The messages are not logged for these packets.

Step 4 Enter the **set security acl ip** *acl_name* **deny log** command to create an IP VACL and enable logging.

Step 5 Enter the **commit security acl** *acl_name* command to commit the VACL to NVRAM.

Step 6 Enter the **set security acl map** *acl_name vlan* command to map the VACL to a VLAN.

Configuration Examples

This example shows how to set the logging level:

```
Console> (enable) set logging level acl 6
System logging facility <acl> for this session set to severity 6(information)
```

This example shows how to allocate a new log table that is based on the maximum flow:

```
Console> (enable) set security acl log maxflow 512
Set VACL Log table to 512 flow patterns.
```

This example shows how to set the redirect rate:

```
Console> (enable) set security acl log ratelimit 1000
Max logging eligible packet rate set to 1000pps.
```

This example shows how to display the VACL log configuration:

```
Console> (enable) show security acl log config
VACL LOG Configration
------------------------------------------------------------
Max Flow Pattern    : 512
Max Logging Eligible rate (pps) : 1000
```

This example shows how to create an ACE for my_cap and specify that the denied traffic is logged:

```
Console> (enable) set security acl ip my_cap deny ip host 21.0.0.1 log
my_cap editbuffer modified. Use 'commit' command to apply changes.
Console> (enable)
```

This example shows how to commit the my_cap ACL to NVRAM:

```
Console> (enable) commit security acl my_cap
ACL commit in progress.

ACL my_cap successfully committed.
Console> (enable)
```

This example shows how to map the VACL to a VLAN:

```
Console> (enable) set security acl map my_cap 1
Mapping in progress.
ACL my_cap successfully mapped to VLAN 1.
:
:
2000 Jul 19 01:14:06 %ACL-6-VACLLOG:VLAN 1(Port 2/1) denied ip tcp 21.0.0.1(2000) ->
255.255.255.255(3000), 1 packet
2000 Jul 19 01:19:06 %ACL-6-VACLLOG:VLAN 1(Port 2/1) denied ip tcp 21.0.0.1(2000) ->
255.255.255.255(3000), 7 packets
2000 Jul 19 01:25:06 %ACL-6-VACLLOG:VLAN 1(Port 2/2) denied ip tcp 21.0.0.1(2000) ->
255.255.255.255(3000), 1 packets
```

This example shows how to display the flow information in the log table:

```
Console> (enable) show security acl log flow ip any any
Total matched entry number = 1
Entry No. #1, IP Packet
----------------------------------------
Vlan Number             : 1
Mod/Port Number         : 2/1
Source IP address       : 21.0.0.1
Destination IP address  : 255.255.255.255
TCP Source port         : 2000
TCP Destination port    : 3000
Received Packet Number  : 10
```

This example shows how to clear the log table:

```
Console> (enable) clear security acl log flow
Log table is cleared.
Console> (enable)
```

Configuring MAC-Based ACL Lookups for All Packet Types

Note This feature is only available with PFC3B and PFC3BXL.

These sections describe how to configure the MAC-based ACL lookups for all packet types:

Overview of MAC-Based ACLs

PFC3A supports two ACL protocol types, IP and MAC. The IP ACL matches only the IP version 4 packets and the MAC ACL matches all packet types *unsupported* by PFC3A (for more information, see the "Creating a Non-IP Version 4/Non-IPX VACL (MAC VACL) and Adding ACEs" section on page 15-52). The packet types that are supported by PFC3A are as follows: IP version 4, MPLS, ARP/RARP, and IP version 6. However, only IP version 4 ACLs can be created in software release 8.4(1) and earlier releases. The unsupported packet types, such as the IPX packet types, are matched using the MAC ACL.

Note The IPX packet types are supported with the PFC and PFC2.

Using MAC-Based ACL Lookups for All Packet Types

PFC3B and PFC3BXL allow the ACL lookups on *all* packet types using the MAC ACL. This feature is useful for doing MAC-based matching on all packets regardless of whether the packet is IP version 4, IP version 6, IPX, MPLS, and so on. You can utilize this feature to rate limit all traffic ingressing a VLAN to some specific value by coupling an aggregate policer with a match-all MAC ACL.

This feature is enabled on a per-ingress VLAN basis and affects the security ACLs (VACLs) and the QoS ACLs. When this feature is enabled on an incoming VLAN, all packets coming in on that VLAN are matched against the MAC-based ACLs, even if they are, for example, IP version 4 packets.

The *ethertype* option has been extended in MAC ACLs to include the IP version 4 EtherType that allows you to set up an ACE to specifically target the IP version 4 packets.

Including the VLAN and CoS in MAC-Based ACLs

With PFC3B and PFC3BXL, you can include the CoS and the VLAN as part of the MAC ACL lookup key that provides support for port-VLAN lookups. This capability is useful on trunk ports where each VLAN can be treated independently. This enhancement affects the VACLs and the QoS MAC ACLs. PFC3B and PFC3BXL overload the VLAN field with the frame type field in the MAC lookup key. Because CoS and VLAN fields are maskable, both fields are added as optional parameters that allow support for the old MAC ACL configurations.

VLAN Matching

With PFC3B and PFC3BXL, if the MAC ACL is mapped to the input, the packet's input VLAN is used to match against the MAC ACL. Similarly, if the MAC ACL is mapped to the output, the output VLAN that is associated with the packet is used to match against the MAC ACL.

Note The MAC ACLs with VLAN matching can be applied only to ports.

VLAN matching can be used in with, or independent of, the MAC-based ACL lookup feature and can do lookups on a port-VLAN basis (the entire VLAN range is supported).

CoS Matching

In both the ingress and egress cases, the CoS that is used to match against the MAC ACL is the input CoS that is associated with the packet. The input CoS is the CoS in the DBus header and is constructed after consulting the port trust (trust-CoS/DSCP/IPprec/untrusted), the default CoS, and the CoS-to-CoS mapping table for the 802.1Q-enabled ports.

Note The CoS matching behavior may be different for the egress ACLs (VACLs and QoS ACLs) depending on how the packet is forwarded. For the normal hardware shortcut packet, the egress ACL matches on the same CoS as the ingress ACL. However, if the packet is forwarded through an intermediate forwarding entity, such as a router or multicast read/write engine, the DBus CoS probably will not be the same as the ingress DBus CoS.

CoS matching can be used with, or independent of, the MAC-based ACL lookup feature.

Configuration Guidelines

Use the following guidelines when configuring MAC-based ACL lookups:

- This feature should be enabled on Layer 2 VLANs only. (This recommendation is for Metro customers.)
- If you enable the feature on a Layer 3 VLAN, be aware of the following:
  - You will lose some Layer 3 features, indicated in the warning message below:

    ```
    Warning:IP RACLs, VACLs & some IP features will be ineffective on these vlans.
    ```

  - You might see inconsistencies in the egress ACL lookup depending on whether the packet is hardware or software forwarded. We recommend that you enable this feature on all VLANs to eliminate any inconsistencies. (This recommendation is for Enterprise customers.)

Configuring MAC-Based ACL Lookups for All Packet Types

The commands described in this section affect both VACLs and QoS MAC ACLs. The **set acl mac-packet-classify** *vlans* command enables the MAC lookup for all packet types incoming on the source VLAN. The **clear acl mac-packet-classify** [*vlans*] command reverts the configuration back to the default for the specified VLAN. The default behavior is to match only MAC packets with MAC ACLs. If you do not specify a VLAN with the **clear acl mac-packet-classify** [*vlans*] command, the feature is disabled for all VLANs. The **show acl mac-packet-classify** command displays the list of VLANs that have the MAC packet classify feature enabled.

Include CoS, VLAN and Packet Type in MAC ACLs and Extend EtherType

The VACL and QoS ACL CLI has been enhanced to include optional parameters for matching on the CoS and VLAN. The commands are as follows:

```
Usage: set security acl mac {acl_name} {permit | deny}
            <src_mac_addr_spec> <dest_mac_addr_spec>
            [<ethertype>] [capture]
            [cos <cos_value>]
            [vlan <vlan>]
            [before <editbuffer_index>|modify <editbuffer_index>]
            (mac_addr_spec = <addr> <mask> or host <addr> or any
     example: 11-22-33-44-00-00 00-00-00-00-ff-ff, host 11-22-33-44-55-66)
            ethertype = names or 0x0, 0x05ff - 0xffff,
            cos_value = 0..7, vlan = 1..4094,

Usage: set qos acl mac {acl_name} {dscp dscp | trust-cos}
            [aggregate <aggregate_name>]
            <src_mac_addr_spec> <dest_mac_addr_spec> [<ethertype>]
            [cos <cos_value>]
            [vlan <vlan>]
            [before <editbuffer_index>|modify <editbuffer_index>]
            (mac_addr_spec = <addr> <mask> or host <addr> or any
     example: 11-22-33-44-00-00 00-00-00-00-ff-ff, host 11-22-33-44-55-66)
            ethertype = names or 0x0, 0x05ff - 0xffff,
            cos_value = 0..7, vlan = 1..4094,
```

The CoS and VLAN fields are optional and if left unspecified, they will match any CoS or VLAN value.

Note An ACL with the VLAN match option can only be mapped to a port.

Note All Cisco IOS ACLs become inoperable when the **set acl mac-packet-classify** *vlans* command is used.

The EtherType has been extended to include an IP version 4 option to allow you to specifically target the IP version 4 packets using the MAC ACL lookup. If you select the IP version 4 option, you must ensure that the corresponding VLAN is enabled using the **set acl mac-packet-classify** *vlans* command. The IP version 4 option was added as follows:

```
Console> (enable) set security acl mac macacl1 permit any any ?
  <0x0, 0x0600 - 0xffff>     Match an EtherType value
  ipv4                       (0x8000)
  ipx-arpa                   (0x8137) Use 0xffff to match on non-arpa IPX
  .......
Console> (enable)
```

This example shows the MAC-based ACL lookup CLI:

```
Console> (enable) set acl mac-packet-classify 5
Enabled mac-packet-classify on vlan(s) 5.
Warning:IP RACLs, VACLs & some IP features will be ineffective on these vlans.
Console> (enable) show acl mac-packet-classify
Feature enabled on source vlan(s) 1,5.
Console> (enable) clear acl mac-packet-classify 5
Disabled mac-packet-classify on vlan(s) 5.
Console> (enable)
```

Note The **all** keyword with the **set** and **clear** commands allow you to specify all VLANs.

Configuring and Storing VACLs and QoS ACLs in Flash Memory

This section describes how to configure and store the VACLs and the QoS ACLs in flash memory instead of NVRAM. Before this feature, all configuration information was stored in NVRAM. With the addition of the QoS and security ACLs (VACLs), NVRAM could become full. In addition to limiting the ACL configuration, filling up NVRAM can cause problems when you attempt to upgrade from one software version to another.

Note In most cases, the 512-KB NVRAM is sufficient for storing the VACLs and QoS ACLs; all ACL configurations are stored in NVRAM by default.

This section describes these tasks:

Note See Chapter 25, "Modifying the Switch Boot Configuration," for additional information on using the commands that are described in this section.

Automatically Moving the VACL and QoS ACL Configuration to Flash Memory

Moving the VACL and QoS ACL configuration to flash memory is done automatically only during the system software upgrades and then only if there is not sufficient NVRAM for the upgrade. If there is not enough NVRAM to perform a software upgrade, the QoS ACL and VACL configuration is deleted from NVRAM and the ACL configuration is automatically moved to flash memory. When this occurs, these syslog messages display:

```
1999 Sep 01 17:00:00 %SYS-1-CFG_FLASH:ACL configuration moved to bootflash:switchapp.cfg
1999 Sep 01 17:00:00 %SYS-1-CFG_ACL_DEALLOC:NVRAM full. Qos/Security ACL configuration
deleted from NVRAM.
```

The VACL and QoS ACL configuration has now been successfully moved to flash memory. During this process, the system also does the following:

- Sets the CONFIG_FILE variable to bootflash:switchapp.cfg
- Enables the **set boot config-register auto-config** command **recurring**, **append**, and **sync** options

If an error occurs during the upgrade, these syslog messages display:

```
1999 Sep 01 17:00:00 %SYS-1-CFG_FLASH_ERR:Failed to write ACL configuration to
bootflash:switchapp.cfg
1999 Sep 01 17:00:00 %SYS-1-CFG_ACL_DEALLOC:NVRAM full. Qos/Security ACL configuration
deleted from NVRAM.
```

If you receive these error messages, the VACL and QoS ACL configuration is stored in DRAM only. You need to make more space available in flash memory and then save the configuration to flash memory (as described in the "Moving the VACL and QoS ACL Configuration Back to NVRAM" section on page 15-67). Alternatively, you might try to delete the unneeded VACLs and the QoS ACLs and save the ACL configuration to NVRAM using the **set config acl nvram** command.

Manually Moving the VACL and QoS ACL Configuration to Flash Memory

If your VACL and QoS ACL configuration requirements require more memory than the 512-KB NVRAM, you can manually move the VACL and QoS ACL configuration to flash memory as follows:

Step 1 Specify the VACL and QoS ACL auto-config file to use to configure the switch at startup.

```
Console> (enable) set boot auto-config bootflash:switchapp.cfg
CONFIG_FILE variable = bootflash:switchapp.cfg
Console> (enable)
```

Step 2 Specify if the switch should retain (**recurring** keyword) or clear (**non-recurring** keyword) the contents of the CONFIG_FILE environment variable after a reset or power cycle.

```
Console> (enable) set boot config-register auto-config recurring
Configuration register is 0x12F
ignore-config: disabled
auto-config: recurring, overwrite, sync disabled
console baud: 9600
boot: image specified by the boot system commands
Console> (enable)
```

Step 3 Specify if the auto-config file should be used to overwrite the NVRAM configuration or be appended to what is currently in NVRAM.

```
Console> (enable) set boot config-register auto-config append
Configuration register is 0x12F
ignore-config: disabled
auto-config: recurring, append, sync disabled
console baud: 9600
boot: image specified by the boot system commands
Console> (enable)
```

Step 4 Specify if synchronization should be enabled or disabled. With synchronization enabled, the auto-config file(s) synchronize automatically to the standby supervisor engine.

```
Console> (enable) set boot config-register auto-config sync enable
Configuration register is 0x12F
ignore-config: disabled
auto-config: recurring, append, sync enabled
console baud: 9600
boot: image specified by the boot system commands
Console> (enable)
```

Step 5 Save the committed VACL and QoS ACL configuration changes to the auto-config file.

```
Console> (enable) copy acl-config bootflash:switchapp.cfg
Upload ACL configuration to bootflash:switchapp.cfg
2843644 bytes available on device bootflash, proceed (y/n) [n]? y
ACL configuration has been copied successfully.
Console> (enable)
```

Step 6 Delete the VACL and QoS ACL configuration from NVRAM.

```
Console> (enable) clear config acl nvram
ACL configuration has been deleted from NVRAM.
Warning: Use the copy commands to save the ACL configuration to a file and
the 'set boot config-register auto-config' commands to configure the
auto-config feature.
```

Note The VACL and QoS ACL mapping commands (**set qos acl map** and **set security acl map**) are also stored in the auto-config file. If the VACL and QoS ACL configuration is in flash memory and you use the mapping commands, you need to enter the **copy** command to save the configuration to flash memory.

The VACL and QoS ACL configuration is no longer in NVRAM. It is saved in the auto-config file bootflash:switchapp.cfg and is appended to the NVRAM configuration at system startup.

After making any additional changes to the VACL and QoS ACL configuration and committing those changes, you must enter the **copy acl-config bootflash:switchapp.cfg** command to save the configuration to the auto-config file.

The auto-config file is synchronized automatically to the standby supervisor engine because synchronization was enabled.

If you cannot write the VACL and QoS ACL configuration to flash memory, it is removed from NVRAM and then the VACL and QoS ACL configuration exists in DRAM only. A system reset can cause the VACL and QoS ACL configuration to revert to the default.

Note If you cannot write the configuration to flash memory, you must copy the configuration to a file, make additional room available in flash memory, and then try to write the VACL and QoS ACL configuration to flash memory.

At system startup, if the VACL and QoS ACL configuration location is set to flash memory but either the CONFIG_FILE variable is not set or none of the files specified exist, this syslog message displays:

```
1999 Sep 01 17:00:00 %SYS-0-CFG_FLASH_ERR:ACL configuration set to flash but no ACL
configuration file found.
```

Running with the VACL and QoS ACL Configuration in Flash Memory

After you move the VACL and QoS ACL configuration to flash memory, the QoS ACLs and VACL commit operations are no longer written to NVRAM. You have to copy the configuration to the flash file manually as follows:

- If you use the **set boot config-register auto-config append** option, the configuration from the auto-config file is appended to the NVRAM configuration. You then only have to copy the VACL and QoS ACL configuration to this file after the commit operations.
- If you do not use the **set boot config-register auto-config append** option, the auto-config feature clears the configuration before executing the auto-config file at system startup. Any changes made in NVRAM are lost. You should always copy your entire configuration (not just the VACL and QoS ACL configuration) to the auto-config file when you want to save it.

Moving the VACL and QoS ACL Configuration Back to NVRAM

This example shows how to move the VACL and QoS ACL configuration back to NVRAM:

```
Console> (enable) set config acl nvram
ACL configuration copied to NVRAM.
Console> (enable)

Console> (enable) clear boot auto-config
CONFIG_FILE variable =
Console> (enable)
```

Redundancy Synchronization Support

The **set boot** commands contain an option to synchronize the auto-config file automatically.

When you enable the **auto-config** option, if the VACL and QoS ACL configuration resides in flash memory, the auto-config file on the active supervisor engine is automatically synchronized to the standby supervisor engine whenever a change is made. For example, deleting the auto-config file on the active supervisor engine causes the file to be deleted on the standby supervisor engine. Similarly, if you insert a new standby supervisor engine, the active supervisor engine automatically synchronizes the auto-config file.

Interacting with High Availability

After a supervisor engine switchover, the VACL and QoS ACL configuration on the standby supervisor engine is consistent with the configuration on the active supervisor engine, just as in the case where the VACL and QoS ACL configuration is saved in NVRAM. The only difference is that the data is stored in DRAM, but the functional behavior of a switchover does not change.

Configuring Port-Based ACLs

Note This feature is available only with Supervisor Engine 720 with PFC3A/PFC3B/PFC3BXL and Supervisor Engine 32 with PFC3B/PFC3BXL.

These sections describe the port ACLs (PACLs):

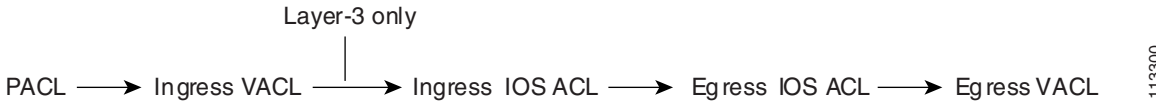
PACL Configuration Overview

Before software release 8.3(1), there were only two types of access lists—the VACLs and Cisco IOS ACLs. The VACLs were applied to Layer 2 and Layer 3 forwarded traffic while Cisco IOS ACLs were applied only to the Layer 3 forwarded packets. Both access list types were applied to the VLANs and filtered traffic based on the packet header information.

In software release 8.3(1), there is an additional type of access list—a PACL. A PACL is an access list that is mapped to a physical port (typically, a VLAN is composed of many physical ports). A PACL provides you with the extra granularity to filter traffic on a specific physical port. Like the VACLs, the PACLs are applied to both the Layer 2 and Layer 3 forwarded packets.

Figure 15-9 shows the logical relationship between the access list types. A PACL is first applied on an incoming packet on a physical port. If the packet is permitted by the PACL, it is filtered by the VACL that is applied to the corresponding ingress VLAN. If the packet is Layer 3 forwarded and is permitted by the VACL, it is filtered by the Cisco IOS ACL on the same VLAN. The same process happens in reverse in the egress direction. However, there is currently no hardware support for the egress PACLs.

Figure 15-9 Logical Relationship Between Access List Types



The PACLs have three modes of operation that are configurable on a per-port basis:

- Port-based—The PACL overrides the existing VACL and Cisco IOS ACL. With this mode, the features such as context-based access control (CBAC) and network address translation (NAT) are not functional on the physical port.
- VLAN-based—The VACL and the Cisco IOS ACL override the PACL.
- Merge—With this mode, the *ingress* PACL, VACL, and Cisco IOS ACL are merged together following the logical serial model in Figure 15-9.

A PACL can be configured on a trunking port except when the port is in merge mode. This restriction occurs because the trunking ports can have multiple VLANs with each VLAN having its own ACL. It would be incorrect to apply a VACL that is meant for VLAN x to a packet that is tagged with VLAN y. Because the PFC3A cannot perform a lookup based on a port-VLAN pair, you cannot map a PACL to a port in merge mode.

Note The CLI syntax for creating a PACL is identical to that of a VACL. An instance of an ACL that is mapped to a port is called a PACL. An instance of an ACL that is mapped to a VLAN is called a VACL. The same ACL can be mapped to both a port and a VLAN. Like the VACLs, the PACLs are supported for all protocol types.

PACL Configuration Guidelines

These sections describe the guidelines for configuring the PACLs:

PACL Interaction with VACLs and Cisco IOS ACLs

This section describes the guidelines for the PACL interaction with the VACLs and Cisco IOS ACLs:

- The PACLs override both the VACLs and Cisco IOS ACLs when the port is configured in port-based mode. The one exception to this rule is when the packets are forwarded in the software by the MSFC. The packets get the ingress Cisco IOS ACL applied regardless of the PACL mode. Two examples where the packets are forwarded in the software are as follows:
  - The packets that are egress bridged (due to logging or features such as NAT)
  - The packets with IP options

  The MSFC reapplies the ingress and egress Cisco IOS ACLs on any packet it sees. The PACL override model for the Layer 3 hardware- and software-forwarded packets is slightly different for Cisco IOS ACLs.
- If a PACL is configured to permit capture and a VACL is configured to deny the same packet, the result of the merge would be a misconfiguration. In this situation, the PACL is placed in the "merge disabled" state.

EtherChannel and PACL Interactions

This section describes the guidelines for the EtherChannel and PACL interactions:

- The ports with different PACL configurations cannot form a port channel; the ports must have the same PACL mode (port-based, VLAN-based, or merge) and the same ACL name to form a port channel.
- If you change one port in an EtherChannel from a port-based ACL to a VLAN-based ACL, all ports in the channel are changed to VLAN-based ACL mode.
- Changing the configuration on one port affects all the ports in the channel. When an ACL is mapped to a port belonging to a channel, it is mapped to all ports in the channel including the logical port that is associated with the channel. The mapping to all physical ports is retained in the hardware and NVRAM even after the port channel is broken; only the mapping to the logical port is removed.
- If a new PACL is applied to one of the ports in an EtherChannel, all the ports in the channel are configured to use the new ACL map.

Dynamic ACLs (Applies to Merge-Mode Only)

The dynamic ACLs are VLAN based and are used by two features: CBAC and IGMP. The merge mode *does not* support the merging of the dynamic ACLs with the PACLs. In merge mode, the following configurations are not allowed:

- Attempting to apply a PACL on a port where its corresponding VLAN has a dynamic ACL mapped.
- Attempting to apply a dynamic ACL on a VLAN where one of its constituent ports has a PACL installed. The dynamic ACL will be mapped successfully, but the port in conflict is placed in "merge disable" mode. The port is reactivated after the dynamic ACL is removed.

Trunking Mode (Applies to Merge-Mode Only)

The PACLs in merge mode are incompatible with the trunking ports. The trunking mode on a port must be set to **off** to allow it to be configured in merge mode. Conversely, a port in merge mode cannot be changed to trunking mode.

Auxiliary VLANs (Applies to Merge-Mode Only)

You cannot configure merge mode on a port that is auxiliary-VLAN enabled. Conversely, a port that is auxiliary-VLAN enabled cannot be changed to merge mode.

Private VLANs (Applies to Merge-Mode Only)

You can map the VACLs to either the primary or the secondary private VLAN. In contrast, you can map only Cisco IOS ACLs to the primary VLANs. An ingress Cisco IOS ACL that is mapped to the primary VLAN gets mapped to all the corresponding secondary VLANs and not to the primary VLAN. An egress Cisco IOS ACL that is mapped to the primary VLAN gets mapped to the primary VLAN.

The ingress lookups on the private VLANs are performed on the secondary VLAN only. In merge mode, the PACLs are merged with the ingress VACLs and Cisco IOS ACLs that are applied to the secondary VLANs.

Port-VLAN Association Changes (Applies to Merge-Mode Only)

The port-VLAN association changes are allowed in all cases. However, when a port is configured in merge mode, it is possible that a change in the port-VLAN association can result in a merge failure. In such cases, the port is placed in “merge disable” mode.

Unmapping and then mapping a PACL, VACL, or Cisco IOS ACL automatically triggers a remerge. This example shows where port 3/1 is associated with VLAN 1 and then VLAN 2:

```
Console> (enable) set port security-acl 3/1 merge
ACL interface is set to merge mode for port(s) 3/1.

Console> (enable) set security acl map ipacl1 3/1
ACL ipacl1 is successfully mapped to port(s) 3/1.

Console> (enable) set security acl map ipacl2 1
ACL ipacl2 is successfully mapped to VLAN 1.

Console> (enable) set security acl map ipacl3 2
ACL ipacl3 is successfully mapped to VLAN 2.

Console> (enable) set vlan 2 3/1
2003 Sep 05 22:34:50 %ACL-3-PACLMERGEFAILED:Failed to merge Security ACLs on Port(s) 3/1
with Vlan 2.
VLAN 2 modified.
VLAN 1 modified.
VLAN  Mod/Ports
---- ----------------------
2     3/1

Console> (enable) show port security-acl 3/1
Port  Interface Type Interface Type Interface Merge Status
      config         runtime        runtime
----- -------------- -------------- ----------------------
 3/1           merge          merge      (VLAN=2) disabled
```

```
Config:
Port  ACL name                          Type
----- -------------------------------- ----
 3/1  ipacl1                           IP

Runtime:
Port  ACL name                          Type
----- -------------------------------- ----
No ACL is mapped to port 3/1.


dhcp-snooping:
Port     Trust     Source-Guard    Source-Guarded IP Addresses
-----  ----------  ------------    ---------------------------
 3/1   untrusted      disabled

Console> (enable) show security acl map runtime 1
Vlan ACL name                          Type
---- -------------------------------- ----
   1 ipacl2                           IP

Console> (enable) show security acl map runtime 2
Vlan ACL name                          Type
---- -------------------------------- ----
   2 ipacl3                           IP
Console> (enable)
```

Online Insertion and Removal

When you remove or reset a module, all the PACLs that are attached to the module are removed from the run-time configuration (which is programmed in the hardware) and the NVRAM configuration (which is stored in NVRAM). The configuration is retained in NVRAM but is not displayed. When you insert or bring a module online, the configuration is repopulated from NVRAM (or text-configuration file) and remapped in runtime.

Enabling or disabling a port has no impact on the ACL mapping or the security-ACL mode, unless the port is in merge mode. In the merge mode, a port that is disabled or cleared from a VLAN is placed in the “merge disable” state because the VLAN that is associated with the port is no longer available and the port cannot forward the packets or merge with any VLAN.

Configuring PACLs from the CLI

These sections describe how to create and activate PACLs on the Catalyst 6500 series switches:

Specifying the PACL Mode

The default PACL mode is VLAN based and keeps any existing VACL configurations active.

To specify the PACL mode, perform this task in privileged mode:

Task	Command
Specify the PACL mode.	**set port security-acl** *mod/ports.*. [**port-based** | **vlan-based** | **merge**]

This example shows how to specify the PACL mode for port 3/1:

```
Console> (enable) set port security-acl 3/1 port-based
Warning: Vlan-based ACL features will be disabled on port(s) 3/1.
ACL interface is set to port-based mode for port(s) 3/1.

Console> (enable) set port security-acl 3/1 merge
ACL interface is set to merge mode for port(s) 3/1.

Console> (enable) set port security-acl 3/1 vlan-based
ACL interface is set to vlan-based mode for port(s) 3/1.
Console> (enable)
```

This example shows the response when trying to configure a trunk port (port 3/1) to merge mode:

```
Console> (enable) set port security-acl 3/1-4 merge
ACL interface cannot be in merge mode on multi-vlan access port 3/1.
ACL interface is set to merge mode for port(s) 3/2.
ACL interface is set to merge mode for port(s) 3/3.
ACL interface is set to merge mode for port(s) 3/4.
```

Displaying PACL Information

The **show port security-acl** *mod/port* command displays PACL information for the specified port. The Config field displays what is stored in NVRAM. The Runtime field displays what is actually programmed in the hardware. The display also shows the status of the merge operation as follows:

- active—There is a PACL configured on the port and it is successfully merged with the VLAN.
- inactive—There is no PACL configured on the port.
- disabled—There is a PACL configured on the port but the merge was unsuccessful (for any number of reasons).

The **show port security-acl** command also displays the VLAN with which the port is configured to merge.

To display PACL information, perform this task in normal mode:

Task	Command
Display PACL information.	**show port security-acl** *mod/port*

This example shows how to display PACL information for port 3/1:

```
Console> (enable) show port security-acl 3/1
Port  Interface Type Interface Type Interface Merge Status
      config         runtime        runtime
----- -------------- -------------- ----------------------
 3/1      port-based     port-based         not applicable

Config:
Port  ACL name                          Type
----- --------------------------------- ----
 3/1  ipacl1                            IP

Runtime:
Port  ACL name                          Type
----- --------------------------------- ----
No ACL is mapped to port 3/1.


dhcp-snooping:
Port     Trust     Source-Guard    Source-Guarded IP Addresses
-----  -----------  ------------    ---------------------------
 3/1   untrusted      disabled

Console> (enable)
```

Mapping an ACL to Ports or to VLANs

An ACL may be mapped to a port even if the port is in VLAN-based mode. In such cases, the configuration is committed to NVRAM and is later restored to the hardware when the port is changed to port-based or merge mode. This functionality is similar to QoS.

Mapping an ACL to a VLAN causes the following operations to occur:

1. The ACL is mapped to the VLAN.
2. A merge is automatically triggered with all the constituent ports that are in merge mode.

If (1) fails, the operation fails and a syslog message is generated. For (2), a syslog is generated for any ports that failed to merge with the VACL. These ports are temporarily placed in VLAN-based mode. If any ports fail to merge, the status of the merge displayed through the **show port security-acl** *mod/port* command is "merge disabled." For an example of the "merge disabled" status, see "Example 6" in the "PACL Configuration Examples" section on page 15-76.

To map an ACL to a port or a VLAN, perform this task in privileged mode:

Task	Command
Map an ACL to a port or a VLAN.	**set security acl map** *acl_name* [*mod/ports* | *vlans*]

This example shows how to map an ACL to port 3/1:

```
Console> (enable) set security acl map ipacl1 3/1
Mapping in progress.
ACL ipacl1 is successfully mapped to port(s) 3/1.
```

```
Console> (enable) set port security-acl 3/1 vlan-based
ACL interface is set to vlan-based mode for port(s) 3/1.

Console> (enable) set security acl map ipacl1 3/1
Port 3/1 is set to vlan-based mode, config is saved in Nvram.
Config will be applied when the port is set to port-based/merge mode.
Console> (enable)
```

Displaying ACL Mapping Information

The **show security acl map** command is extended to display the port mappings as follows:

- Added mandatory keywords (**config** and **runtime**) to display the configuration and run-time mappings.
- Added optional keywords (**all-vlans** and **all-ports**) to selectively display the configured VACLs and PACLs.

To display the ACL mapping information, perform this task in normal mode:

Task	Command
Display the ACL mapping information.	**show security acl map** [**config** | **runtime**] [*acl_name* | *mod_num*/*port_num* | *vlan* | **all** | **all-vlans** | **all-ports**]

These examples show how to display the ACL mapping information:

```
Console> (enable) show security acl map config all
ACL Name                         Type Ports/Vlans
-------------------------------- ---- --------------
ipacl1                           IP   11
ipacl2                           IP   3/1

Console> (enable) show security acl map config all-ports
ACL Name                         Type Ports
-------------------------------- ---- --------------
ipacl2                           IP   3/1

Console> (enable) show security acl map runtime 3/1
Port  ACL name                          Type
----- -------------------------------- ----
3 / 1 ipacl1                           IP
Console> (enable)
```

Displaying ACL Information for an EtherChannel

The **show port channel** command is extended to display the PACL mappings on the port channels. For *type*, you can specify *security-acl*.

To display the ACL information for an EtherChannel, perform this task in normal mode:

Task	Command
Display the ACL information for an EtherChannel.	**show port channel** [**all** | *mod*[/*port*]] {**info** [*type*]}

This example shows how to display the ACL information for an EtherChannel:

```
Console> (enable) show port channel 3/40 info security-acl
Port  ACL-Interface Type
----- ------------------
 3/37 port-based
 3/38 port-based

Port  ACL name                         Type
----- -------------------------------- ------
 3/37 ipacl1                           IP
 3/38 ipacl1                           IP
Console> (enable)
```

PACL Configuration Examples

This section contains the PACL configuration examples.

Note If no ACL is mapped to a port, the port reverts internally to VLAN-based mode.

Example 1

This example shows how to map an ACL to a port when the port is in VLAN-based mode:

```
Console> (enable) set port security-acl 3/1 vlan-based
ACL interface is set to vlan-based mode for port(s) 3/1.

Console> (enable) set security acl map ipacl1 3/1
Port 3/1 is set to vlan-based mode, config is saved in Nvram.
Config will be applied when the port is set to port-based/merge.

Console> (enable) show security acl map config 3/1
Port  ACL name                         Type
----- -------------------------------- ----
 3/1  ipacl1                           IP

Console> (enable) show security acl map runtime 3/1
Port  ACL name                         Type
----- -------------------------------- ----
No ACL mapped to port 3/1.

Console> (enable) set port security-acl 3/1 port-based
Warning: Vlan-based ACL features will be disabled on port(s) 3/1.
ACL interface is set to port-based mode for port(s) 3/1.

Console> (enable) show security acl map config 3/1
Port  ACL name                         Type
----- -------------------------------- ----
 3/1  ipacl1                           IP

Console> (enable) show security acl map runtime 3/1
Port  ACL name                         Type
----- -------------------------------- ----
 3/1  ipacl1                           IP
Console> (enable)
```

Example 2

This example shows a failure that occurs when changing the security ACL mode due to an ACL mapping error. In this example, the ACL is mapped only in NVRAM and not in the hardware.

```
Console> (enable) set port security-acl 3/1 vlan-based
ACL interface is set to vlan-based mode for port(s) 3/1.

Console> (enable) set security acl map ipacl1 3/1
Port 3/1 is set to vlan-based mode, config is saved in Nvram.
Config will be applied when the port is set to port-based/merge.

Console> (enable) set port security-acl 3/1 port-based
Warning: Vlan-based ACL features will be disabled on port(s) 3/1.
ACL interface is set to port-based mode for port(s) 3/1
2003 Sep 05 22:34:50 %ACL-3-TCAMFULL:Acl engine TCAM table is full
2003 Sep 05 22:34:50 %ACL-3-PACLMAPCOMMITFAIL:Failed to Map Security ACL ipacl1 to Port
3/1

Console> (enable) show security acl map config 3/1
Port  ACL name                          Type
----- -------------------------------- ----
 3/1  ipacl1                           IP

Console> (enable) show security acl map runtime 3/1
Port  ACL name                          Type
----- -------------------------------- ----
No ACL is mapped to port 3/1.


Console> (enable) show port security-acl 3/1
Port  Interface Type Interface Type Interface Merge Status
      config         runtime        runtime
----- -------------- -------------- ----------------------
 3/1      port-based     port-based         not applicable

Config:
Port  ACL name                          Type
----- -------------------------------- ----
 3/1  ipacl1                           IP

Runtime:
Port  ACL name                          Type
----- -------------------------------- ----
No ACL is mapped to port 3/1.


dhcp-snooping:
Port     Trust     Source-Guard    Source-Guarded IP Addresses
----- -----------  ------------    ---------------------------
 3/1   untrusted      disabled

Console> (enable)
```

Example 3

This example shows a port that is configured in merge mode but the port has not been mapped to an ACL:

```
Console> (enable) set port security-acl 3/1 merge
ACL interface is set to merge mode for port(s) 3/1.

Console> (enable) show port security-acl 3/1
Port  Interface Type Interface Type Interface Merge Status
      config         runtime        runtime
----- -------------- -------------- ----------------------
 3/1           merge          merge      (VLAN 5) inactive

Config:
Port  ACL name                          Type
----- --------------------------------- ----
No ACL is mapped to port 3/1.

Runtime:
Port  ACL name                          Type
----- --------------------------------- ----
No ACL is mapped to port 3/1.


dhcp-snooping:
Port     Trust     Source-Guard    Source-Guarded IP Addresses
-----  -----------  ------------    ---------------------------
 3/1   untrusted      disabled

Console> (enable) set security acl map ipacl1 3/1
ACL ipacl1 is successfully mapped to port(s) 3/1.

Console> (enable) show port security-acl 3/1
Port  Interface Type Interface Type Interface Merge Status
      config         runtime        runtime
----- -------------- -------------- ----------------------
 3/1           merge          merge        (VLAN 5) active

Config:
Port  ACL name                          Type
----- --------------------------------- ----
 3/1  ipacl1                            IP

Runtime:
Port  ACL name                          Type
----- --------------------------------- ----
 3/1  ipacl1                            IP


dhcp-snooping:
Port     Trust     Source-Guard    Source-Guarded IP Addresses
-----  -----------  ------------    ---------------------------
 3/1   untrusted      disabled

Console> (enable)
```

Example 4

This example shows that a merge failure occurs when mapping an ACL to a port. In this case, the configuration is not saved.

```
Console> (enable) set port security-acl 3/1 merge
ACL interface is set to merge for port(s) 3/1.

Console> (enable) set security acl map ipacl1 3/1
Mapping in progress.
2003 Oct 01 19:44:31 %ACL-3-PACLMAPCOMMITFAIL:Failed to Map Security ACL ipacl1 to Port
3/15
Failed to attach ACL ipacl1 to port(s) 3/1.

Console> (enable) show security acl map config 3/1
Port  ACL name                         Type
----- -------------------------------- ----
No ACL is mapped to port 3/1.

Console> (enable) show security acl map runtime 3/1
Port  ACL name                         Type
----- -------------------------------- ----
No ACL is mapped to port 3/1.
Console> (enable)
```

Example 5

This example shows that you cannot change the mode if a failure occurs when changing port-based mode to merge mode:

```
Console> (enable) set port security-acl 3/1 port-based
ACL interface is set to port-based for port(s) 3/1.

Console> (enable) set security acl map ipacl1 3/1
ACL ipacl1 is successfully mapped to port 3/1.

Console> (enable) show security acl map config 3/1
Port  ACL name                         Type
----- -------------------------------- ----
 3/1  ipacl1                           IP

Console> (enable) show security acl map runtime 3/1
Port  ACL name                         Type
----- -------------------------------- ----
 3/1  ipacl1                           IP

Console> (enable) set port security-acl 3/1 merge
Failed to set interface to merge mode for port(s) 3/1.
2003 Oct 01 19:53:01 %ACL-3-TCAMFULL:Acl engine TCAM table is full
Console> (enable)
```

Example 6

This example shows that a syslog is generated for any ports that fail to merge with the VACL and these ports are temporarily placed in VLAN-based mode. The status of the merge is “merge disabled.”

```
Console> (enable) show port security-acl 3/1
Port  Interface Type Interface Type Interface Merge Status
      config         runtime        runtime
----- -------------- -------------- ----------------------
 3/1           merge          merge        (VLAN=5) active

Config:
Port  ACL name                         Type
----- -------------------------------- ----
 3/1  ipacl1                           IP
 3/1  macacl1                          MAC

Runtime:
Port  ACL name                         Type
----- -------------------------------- ----
 3/1  ipacl1                           IP
 3/1  macacl1                          MAC


dhcp-snooping:
Port     Trust     Source-Guard    Source-Guarded IP Addresses
-----  -----------  ------------    ---------------------------
 3/1   untrusted      disabled

Console> (enable) set security acl map ipacl2 5
ACL ipacl2 is successfully mapped to VLAN 5.
2003 Oct 01 20:01:04 %ACL-3-MERGEFAILED:Failed to merge Security ACLs on ports(s) 3/1-4
with VLAN 5
2003 Oct 01 20:01:04 %ACL-3-PACLSMERGEDFORVLAN:Merge completed for all ports on Vlan 5

Console> (enable) show port security-acl 3/1
Port  Interface Type Interface Type Interface Merge Status
      config         runtime        runtime
----- -------------- -------------- ----------------------
 3/1           merge          merge      (VLAN=5) disabled
Config:
Port  ACL name                         Type
----- -------------------------------- ----
 3/1  ipacl1                           IP
 3/1  macacl1                          MAC

Runtime:
Port  ACL name                         Type
----- -------------------------------- ----
 3/1  ipacl1                           IP
 3/1  macacl1                          MAC


dhcp-snooping:
Port     Trust     Source-Guard    Source-Guarded IP Addresses
-----  -----------  ------------    ---------------------------
 3/1   untrusted      disabled

Console> (enable)
```

Example 7

This example is a continuation from Example 6 and shows that you can recover from the failure state by either mapping or unmapping the VACL or PACL. This example shows that detaching the MAC PACL can release some TCAM resources, allowing the merge to succeed. A syslog is generated when the merge is reenabled.

```
Console> (enable) clear security acl map macacl1
Map deletion in progress.
Successfully cleared mapping between ACL macacl1 and port 3/1.
2003 Oct 01 20:01:04 %ACL-3-PACLMERGED:Merged Security ACLs on port(s) 3/1

Console> (enable) show port security-acl 3/1
Port  Interface Type Interface Type Interface Merge Status
      config         runtime        runtime
----- -------------- -------------- ----------------------
 3/1           merge          merge        (VLAN=5) active

Config:
Port  ACL name                         Type
----- -------------------------------- ----
 3/1  ipacl1                           IP

Runtime:
Port  ACL name                         Type
----- -------------------------------- ----
 3/1  ipacl1                           IP


dhcp-snooping:
Port     Trust      Source-Guard    Source-Guarded IP Addresses
-----  -----------  ------------    ---------------------------
 3/1   untrusted      disabled

Console> (enable)
```

Configuring ACL Statistics

These sections describe how to configure the ACL statistics:

- ACL Statistics Overview, page 15-81
- Configuring ACL Statistics from the CLI, page 15-82

ACL Statistics Overview

When you select the **statistics** keyword with the **set security acl** command set, the statistics are stored for the ACEs or the ACLs (VACLs and PACLs). The ACL statistics are disabled by default and can be enabled on a per-ACL, per-VLAN, or per-ACE basis.

Before an ACL is programmed in the TCAM, it is passed to the ACL compiler. The ACL compiler optimizes the ACL in terms of the number of ACEs and promotes mask sharing, where possible, to reduce the number of TCAM masks used. When there are multiple features/policies configured through the ACLs on an interface, the ACLs are merged and the resultant ACL is optimized. The resultant ACL is logically equivalent to the original ACL(s).

Optimizing an ACL involves removing the redundant ACEs, merging the ACEs, and reordering the ACEs. Removing the redundant ACEs and merging the ACEs reduces the number of TCAM entries. Reordering the ACEs reduces the number of TCAM entries and the number of TCAM masks.

The ACL statistics are derived from the counters of the ACEs that comprise the optimized ACL. A mapping function maps these ACEs to the ACEs corresponding to the original user-configured ACLs.

Note With PFC2 and PFC3A, the counters are based on software sampling and are not accurate. PFC3B/PFC3BXL use the hardware counters that provide accurate statistics. With PFC2/PFC3A, the counters report if a particular ACE was hit during a 300-ms window but the counters do not indicate how much traffic hit the entry. For example, if you have two flows where one flow is 1000 packets per second and the second flow is 10 packets per second, both flows return the same result with a PFC2/PFC3A. PFC3B/PFC3BXL and later PFCs do not have this limitation.

Note The ACL statistics could differ between the active and standby supervisor engines because the ACLs cannot be programmed into the active/standby TCAMs at the exact time. However, if the traffic starts hitting the TCAM after the TCAM is programmed, the ACL statistics should be the same.

Configuring ACL Statistics from the CLI

This section provides these example procedures:

Enabling the ACL Compiler Optimization

Enter the **set security acl comp-opt** command to optimize the ACL compiler.

To enable ACL compiler optimization, perform this task in privileged mode:

Task	Command
Enable ACL compiler optimization.	**set security acl comp-opt** {**enable** | **disable**}

This example shows how to enable ACL compiler optimization:

```
Console> (enable) set security acl comp-opt enable
Acl Compiler Optimization Enabled.
Console> (enable) show security acl comp-opt
Acl Compiler Optimization Enabled
Console> (enable)
```

Enabling ACL Statistics on a Per-ACL Basis

Note The ARP entry statistics collection is always enabled because the ARP ACE entry is added after the ACL merge and is always the first ACE in the TCAM list.

Enter the **set security acl statistics** {*acl_name* | **all**} command to enable the aggregated ACL statistics on a per-ACL basis or for all ACLs. In the aggregated statistics mode, the statistics are enabled for all the ACEs in the specified ACL. This command is effective only after you enter the **commit** command to commit all ACEs to NVRAM.

Note The **set security acl statistics** {*acl_name* | **all**} command overwrites the per-ACE command, **set security acl ip/mac** *acl_name* … [**statistics**].

Note The aggregated statistics mode disables the merge optimization and can result in a larger number of ACEs. In some cases, an ACL that was previously installed in the TCAM, might not fit in the TCAM after the aggregated statistics mode is enabled.

To enable the aggregated ACL statistics on a per-ACL basis, perform this task in privileged mode:

Task	Command
Enable the aggregated ACL statistics on a per-ACL basis.	**set security acl statistics** {*acl_name* | **all**}

This example shows how to enable the aggregated ACL statistics on a per-ACL basis:

```
Console> (enable) set security acl statistics ACL1
ACL1 editbuffer modified. Use 'commit' command to save changes.
Console> (enable) commit security acl ACL1
ACL commit in progress.

ACL 'ACL1' successfully committed.
Console> (enable)

Console> (enable) show security acl info ACL1
set security acl ip ACL1 statistics
--------------------------------------------------
arp permit
1. permit ip any any
Console> (enable)
```

Enabling ACL Statistics on a Per-VLAN Basis

Enter the **set security acl map** *acl-name* {*vlan*/*mod_port*} [**statistics enable | disable**] command to enable the ACL statistics on a per-VLAN basis.

Note In the per-VLAN mode, label sharing is disabled. For example, if you have an ACL that is mapped to 10 VLANs and you enable per-VLAN statistics on one of the VLANs, you will have nine VLANs sharing a label. The VLAN on which you enabled VLAN statistics will have a different label, but this does not imply that statistics are enabled. If the ACL that you mapped does not have the statistics enabled (either per-ACE or per-ACL), you will not see any statistical information except for the ARP packets.

If the per-VLAN statistics are enabled on a VLAN, the subsequent maps that are configured on the same VLAN will also have the per-VLAN statistics enabled. If the per-VLAN statistics are disabled on a VLAN, the previous maps that are configured on the same VLAN will also have the per-VLAN statistics disabled.

For example, if you enter the **set security acl map ip1 1 statistics enable** command followed by the **set security acl map mac1 1** command, the mac1 ACL will also have the per-VLAN statistics enabled.

If you enter the **set security acl map ip1 1 statistics enable** command followed by the **set security acl map mac1 1 statistics disable** command, the ip1 ACL will also have the per-VLAN statistics disabled.

To enable the ACL statistics on a per-VLAN basis, perform these tasks in privileged mode:

Task	Command
Enable the ACL statistics on a per-VLAN basis.	**set security acl map** *acl-name* {*vlan*/*mod_port*} [**statistics enable	disable**]
Display the configuration.	**show security acl**

This example shows how to enable the ACL statistics on a per-VLAN basis:

```
Console> (enable) set security acl map ACL1 1 statistics enable
Mapping in progress.

ACL ACL1 successfully mapped to VLAN 1.
Console> (enable)

Console> (enable) show security acl info ACL1
set security acl ip ACL1 statistics
-------------------------------------------------
arp permit
1. permit ip any any
Console> (enable)
```

Enabling ACL Statistics on a Per-ACE Basis

Enter the **set security acl ip/mac** *acl_name* … [**statistics**] command to enable the ACL statistics on a per-ACE basis. This option allows you to collect the statistics for the configured ACEs even if the ACL statistics are not enabled. This command is effective only after you enter the **commit** command to commit all ACEs to NVRAM.

To enable the ACL statistics on a per-ACE basis, perform this task in privileged mode:

Task	Command
Enable the ACL statistics on a per-ACE basis.	**set security acl ip/mac** *acl_name* … [**statistics**]

This example shows how to enable the ACL statistics on a per-ACE basis:

```
Console> (enable) set security acl ip ACL1 permit ip any any statistics
ACL1 editbuffer modified. Use 'commit' command to apply changes.
Console> (enable) commit security acl ACL1
ACL commit in progress.

ACL 'ACL1' successfully committed.
Console> (enable)

Console> (enable) show security acl info ACL1
set security acl ip ACL1 statistics
-------------------------------------------------
arp permit
1. permit ip any any
2. permit ip any any statistics
Console> (enable)
```

Clearing ACL Statistics

Use the commands described in this section to clear the ACL statistics:

- **clear security acl statistics** *acl_name*

  Disables the collection of statistics for all the ACEs in the specified ACL. This command works only for the ACL statistics that are configured on a per-ACL basis. The command does not work for the ACL statistics that are configured on a per-VLAN or per-ACE basis. This command is effective only after you enter the **commit** command to commit all ACEs to NVRAM.

  An example is as follows:

  ```
  Console> (enable) clear security acl statistics ACL1
  ACL1 editbuffer modified. Use 'commit' command to save changes.
  Console> (enable) commit security acl ACL1
  ACL commit in progress.

  ACL 'ACL1' successfully committed.
  Console> (enable)
  ```

- **clear security acl counters**

  Clears all statistic counters.

  An example is as follows:

  ```
  Console> (enable) clear security acl counters
  Operation Successful.
  Console> (enable)
  ```

Displaying ACL Statistics Information

Use the commands described in this section to display information about the ACL statistics:

- **show security acl info** *acl_name* [**statistics** [*ace_index*]]

  Displays the statistics for the specified ACL. The *ace_index* is the index in the ACL list (committed ACLs).

  An example is as follows:

```
Console> (enable) show security acl info ACL1 statistics
Vlan: 1
set security acl ip ACL1 statistics
-------------------------------------------------
arp permit in: 132 out: 132
1. permit ip any any
2. permit ip any any statistics  in: 0 out: 0

Console> (enable)
```

- **show security acl tcam interface** *vlan*

  Displays the TCAM details for the specified VLAN.

  An example is as follows:

```
Console> (enable) show security acl tcam interface 1
Input
0. permit arp (matches 45745)
1. deny (l3) tcp any any fragment (matches 0)
2. deny (l3) ip host 21.0.0.130 any (matches 0)
3. deny (l3) udp 1.2.2.0 0.0.0.255 any (matches 0)
4. deny (l3) tcp any any 2001 (matches 0)
5. deny (l3) ip host 21.0.0.128 any (matches 0)
6. deny ip any any (matches 3)


Output
0. permit arp (matches 0)
1. deny (l3) tcp any any fragment (matches 0)
2. deny (l3) ip host 21.0.0.130 any (matches 0)
3. deny (l3) udp 1.2.2.0 0.0.0.255 any (matches 0)
4. deny (l3) tcp any any 2001 (matches 0)
5. deny (l3) ip host 21.0.0.128 any (matches 0)
6. deny (l3) ip any any (matches 0)
Console> (enable)
```

  The fields are described as follows:

  - deny (l3): Layer 3 traffic is denied; Layer 2 traffic is permitted.
  - redirect (l3): Only Layer 3 traffic is redirected.
  - bridge: Traffic that hits this entry is bridged.
  - Redirect (adj): Traffic is rewritten by the adjacency information.

- **show security acl** and **show security acl map** *acl_name*

  A new field is added to these commands to display the type of statistics that are enabled for a specific ACL or VLAN.

  An example is as follows:

```
Console> (enable) show security acl
Information in the bracket.
```

```
   Disable - statistics are not enabled per ACL
   Enable - stats are enabled per ACL
   The number shows the VLANs where per-vlan stattistics are enabled
ACL                              Type VLANS (Statistics)
-------------------------------  ---- ----------------
ip1                              IP   2-9    (2-3 Enable)
ip2                              IP   10     (Disable)
ip3                              IP   11     (Disable)
Console> (enable)
```

The fields are described as follows:

- Disable: The statistics are not enabled on the ACL.
- Enable: The statistics are enabled on the ACL.
- The numbers show the VLANs where per-VLAN statistics are enabled (“2-3” in the example).

Configuring the Compression and Reordering of ACL Masks

The compression and reordering of the ACL masks (CRAM) feature optimizes the mask usage across the different ACLs. This optimization promotes mask sharing and results in more efficient usage of the TCAM and the ability to program more ACLs in the TCAM.

The TCAM is used for implementing the ACLs in the hardware. One mask entry is shared among eight value entries. When programming the ACLs, it is possible to see the error condition where the TCAM is full and can no longer program any new ACLs into the TCAM hardware. This problem is almost always caused by a shortage of TCAM masks.

You can run CRAM in two modes. In the manual mode, you execute the feature when desired. In the automatic mode, the feature is run whenever a TCAM full exception is seen. When the feature is executed, the new mask ordering is computed and the ACL hardware is programmed accordingly.

Note With software release 8.4(1), CRAM is supported only for the security ACLs. This feature works for the QoS ACLs but you cannot specifically run the feature on the QoS ACLs.

Configuring the CRAM Feature from the CLI

Note When the CRAM feature is run, the traffic is disrupted (denied) for a period of less than 0.5 seconds during the programming of the hardware.

This section contains these example procedures:

- Enabling a Test Run of the CRAM Feature, page 15-88
- Enabling the CRAM Feature Manually, page 15-88
- Enabling the Automatic Execution of the CRAM Feature, page 15-88
- Displaying the CRAM Feature Status Information, page 15-89
- Disabling the CRAM Feature Automatic Mode, page 15-89

Enabling a Test Run of the CRAM Feature

Enter the **set security acl cram testrun** command to determine the ACL mask usage. Running this command is for informational purposes only; no software or hardware structures are modified and there is no disruption of traffic.

To enable a test run of the CRAM feature, perform this task in privileged mode:

Task	Command
Enable a test run of the CRAM feature.	**set security acl cram testrun**

This example shows how to enable a test run of the CRAM feature:

```
Console> (enable) set security acl cram testrun
CRAM execution in progress.

CRAM execution complete.
Current ACL storage mask usage 60.0%
ACL storage mask usage if CRAM is run is 41.0%
Console> (enable)
```

Enabling the CRAM Feature Manually

Enter the **set security acl cram run** command to manually enable the CRAM feature.

To manually enable the CRAM feature, perform this task in privileged mode:

Task	Command
Manually enable the CRAM feature.	**set security acl cram run**

This example shows how to manually enable the CRAM feature:

```
Console> (enable) set security acl cram run
Traffic may be disrupted for some time while programming hardware. Agree (y/n)[n] ? y
CRAM execution in progress.

CRAM execution complete.
Previous ACL storage mask usage 60.0%
Current ACL storage mask usage 41.0%
Console> (enable)
```

Enabling the Automatic Execution of the CRAM Feature

Enter the **set security acl cram auto** [*nsec*] command to enable the automatic execution of the CRAM feature. When automatic execution is enabled, the feature is run automatically when there is a TCAM full condition. The default timer setting is 300 seconds. You can specify *nsec* from 60 to 3600 seconds. If there have been no changes to the TCAM since the last time the feature was run, the feature is not automatically executed.

To enable the automatic execution of the CRAM feature, perform this task in privileged mode:

Task	Command
Enable the automatic execution of the CRAM feature.	**set security acl cram auto** [*nsec*]

These examples show how to enable the automatic execution of the CRAM feature:

```
Console> (enable) set security acl cram auto
Cram auto mode enabled. Timer is default =  300 seconds
Console> (enable)

Console> (enable) set security acl cram auto 1000
Cram auto mode enabled. Timer is 1000 seconds
Console> (enable)
```

Displaying the CRAM Feature Status Information

Enter the **show security acl cram** command to display the CRAM feature status information.

To display the CRAM feature status information, perform this task in normal mode:

Task	Command
Display the CRAM feature status information.	**show security acl cram**

This example shows how to display the CRAM feature status information:

```
Console> (enable) show security acl cram
Cram auto mode is enabled. Timer is 300.
Cram last run on Fri Jun 18 2004, 10:06:29
Security ACL mask usage before: 0.17%
Security ACL mask usage after: 0.12%
Total number of cram executions = 2
Console> (enable)
```

Disabling the CRAM Feature Automatic Mode

Enter the **clear security acl cram auto** command to disable the CRAM automatic mode.

To disable the CRAM automatic mode, perform this task in privileged mode:

Task	Command
Disable the CRAM automatic mode.	**clear security acl cram auto**

This example shows how to disable the CRAM automatic mode:

```
Console> (enable) clear security acl cram auto
Cram auto mode disabled.
Console> (enable)
```

Configuring Policy-Based Forwarding

Policy-based forwarding (PBF) is an extension of VACL redirection that is supported by the PFC2 and PFC3A/PFC3B/PFC3BXL. PBF is particularly beneficial in any flat Layer 2 network that is used for transparent bridging where a limited amount of inter-VLAN communication is required and in server farms or demilitarized zones (DMZs) where bridging devices (like server load-balancing appliances) are involved or where firewall load balancing is performed.

Note Software release 7.5(1) and later releases have PBF enhancements that simplify the process of setting and committing the security ACLs and adjacency information. For more information, see the "Enhancements to PBF Configuration (Software Releases 7.5(1) and Later)" section on page 15-102.

Note Software release 8.3(1) and later releases have further PBF enhancements that simplify the process of setting and committing the security ACLs and adjacency information. For more information, see the "Enhancements to the PBF Configuration (Software Releases 8.3(1) and Later)" section on page 15-105.

Note PBF does not support IPX and multicast traffic.

Note PBF does not work with 802.1Q tunnel traffic. PBF is supported on the Layer 3 IP unicast traffic; it is not applicable to the Layer 2 traffic. At the intermediate (PBF) switch, all 802.1Q tunnel traffic appears as Layer 2 traffic.

Note PBF may require some configuration on the attached hosts. When a router is not present in the network, the ARP table entries have to be statically added on each host participating in PBF.

PBF is described in these sections:

Understanding How PBF Works

The PBF configuration involves these tasks:

- Enabling PBF and specifying a MAC address for the PFC2 or PFC3A/PFC3B/PFC3BXL
- Configuring the VACLs for PBF
- Configuring the attached hosts for PBF

You enable PBF by specifying a MAC address for the PFC2 or PFC3A/PFC3B/PFC3BXL. The MAC address can be a default or user-specified MAC address. When the packets are sent, the destination MAC address has to be identical to the PFC2 or PFC3A/PFC3B/PFC3BXL MAC address. The PFC2 or PFC3A/PFC3B/PFC3BXL must think that the packet is a Layer 3 packet or no rewrite operation occurs. If the packets are not sent with the PFC2 or PFC3A/PFC3B/PFC3BXL MAC address, the PFC2 or PFC3A/PFC3B/PFC3BXL treats the packets as the Layer 2 packets.

The PBF VACL is created by using the **set security acl** commands. The PBF VACL contains an adjacency table entry for the PFC2 or PFC3A/PFC3B/PFC3BXL and a redirect ACE. You must set the VACLs on both VLANs that participate in PBF. When the packet from the source VLAN comes into the PFC2 or PFC3A/PFC3B/PFC3BXL, it hits the PBF VACL. Based on the information that is provided in the adjacency table, the packet header (the destination VLAN and source and destination MAC addresses) is rewritten and the packet is forwarded to the destination VLAN. The packets are forwarded between VLANs only if they hit the VACL entries that are associated with the adjacency information.

Note Because the VACLs are applied to the incoming and outgoing traffic, you must configure all VACLs carefully when using PBF. If the VACLs are not specific, a rewritten packet could hit a deny statement in the outgoing VACL and get dropped.

When a router is not present in the network, you need to specify the static ARP entries on the participating hosts.

PBF Hardware and Software Requirements

The PBF hardware and software requirements are as follows:

- PBF requires Supervisor Engine 2 with PFC2, Supervisor Engine 720 with PFC3A/PFC3B/PFC3BXL, or Supervisor Engine 32 with PFC3B/PFC3BXL.
- PBF *is not* supported with an operating (booted) MSFC2, MSFC2A, or MSFC3 in the Catalyst 6500 series switch that is being used for PBF.

  If you try to configure PBF with an MSFC2, MSFC2A, or MSFC3 present and booted, the system responds with a message indicating that the feature is not supported with an MSFC2, MSFC2A, or MSFC3.

  If an MSFC2, MSFC2A, or MSFC3 is present but has not booted, you can configure PBF.
- For Supervisor Engine 2, PBF requires supervisor engine software release 6.3(1) or later releases.
- For Supervisor Engine 720, PBF requires supervisor engine software release 8.1(1) or later releases.
- For Supervisor Engine 32, PBF requires supervisor engine software release 8.4(1) or later releases.

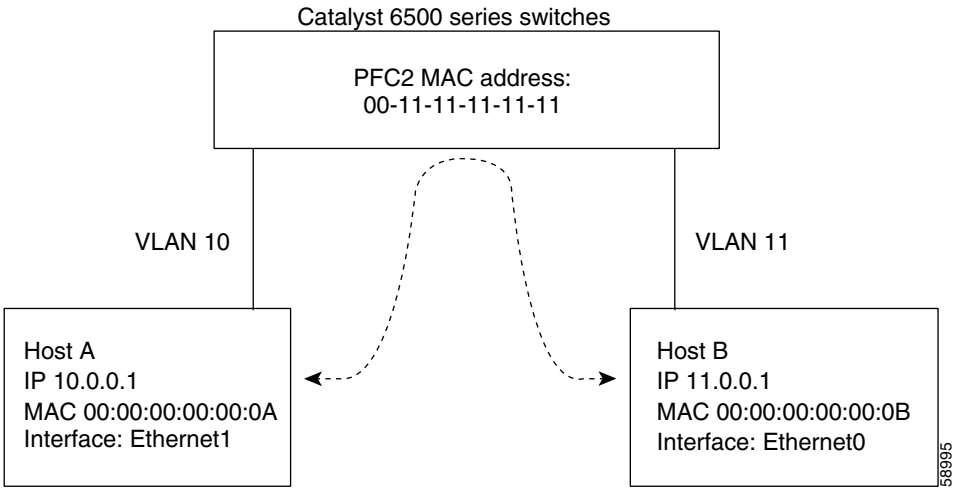
Configuring PBF from the CLI

Note In addition to the guidelines and configuration examples in this section, see the "Enhancements to PBF Configuration (Software Releases 7.5(1) and Later)" section on page 15-102 and the "Enhancements to the PBF Configuration (Software Releases 8.3(1) and Later)" section on page 15-105.

This section describes the guidelines and the configuration examples for PBF. The configuration examples use the example configuration that is shown in Figure 15-10. The Catalyst 6500 series switch redirects all the traffic coming from Host A on VLAN 10 to Host B on VLAN 11 and redirects the traffic from Host B to Host A. This section contains these example procedures:

- Enabling PBF and Specifying a MAC Address for the PFC2 or PFC3A/PFC3B/PFC3BXL, page 15-92
- Specifying the PBF MAC Address on a VLAN, page 15-94
- Configuring VACLs for PBF, page 15-94
- Displaying PBF Information, page 15-96
- Clearing Entries in PBF VACLs, page 15-97
- Rolling Back Adjacency Table Entries in the Edit Buffer, page 15-98
- Configuring Hosts for PBF, page 15-98

Figure 15-10 Policy-Based Forwarding



Enabling PBF and Specifying a MAC Address for the PFC2 or PFC3A/PFC3B/PFC3BXL

Note The MAC address can be a default or user-specified MAC address. The default MAC address is taken from a MAC address PROM on the Catalyst 6500 series switch chassis. When specifying a MAC address using the **set pbf mac** command, ensure that the MAC address is unique and is not already being used on any interfaces.

We recommend that you use the default MAC address that is provided by the MAC address PROM. When you specify your own MAC address using the **set pbf mac** command, if the MAC address is a duplicate of a MAC address that is already in use, some packets might get dropped.

To display the PBF status and the MAC address, perform this task in privileged mode:

Task	Command
Display the PBF status and the MAC address.	**show pbf**

To enable PBF, perform one of these tasks in privileged mode:

Task	Command
Enable PBF with a default MAC address.	**set pbf**
Enable PBF with a specific MAC address.	**set pbf** [**mac** *mac_address*]

This example shows how to check the PBF status and the MAC address, enable PBF with a default MAC address, and verify the change:

```
Console> (enable) show pbf
Pbf status    Mac address
-----------   ------------------
not set       00-00-00-00-00-00
Console> (enable)
Console> (enable) set pbf
PBF committed successfully.
Operation successful.
Console> (enable)
Console> (enable) show pbf
Pbf status    Mac address
-----------   ------------------
ok            00-01-64-61-39-c2
Console> (enable)
```

This example shows how to enable PBF with a specific MAC address:

```
Console> (enable) set pbf mac 00-11-11-11-11-11
PBF committed successfully.
Operation successful.
Console> (enable)

Console> (enable) show pbf
Pbf status    Mac address
-----------   ------------------
ok            00-11-11-11-11-11
Console> (enable)
```

To disable PBF and clear the PBF MAC address, perform this task in privileged mode:

Task	Command
Disable PBF and clear the PBF MAC address.	**clear pbf**

This example shows how to clear the PBF MAC address:

```
Console> (enable) clear pbf
PBF cleared.
Console> (enable)

Console> (enable) show pbf
Pbf status    Mac address
-----------   ------------------
not set       00-00-00-00-00-00
Console> (enable)
```

Specifying the PBF MAC Address on a VLAN

Note This PBF configuration step is required only on the Supervisor Engine 720 with PFC3A/PFC3B/PFC3BXL.

The **set pbf vlan** *vlan* command creates the PBF Layer 2 CAM entries on the specified VLANs. The packets that match these entries are classified as Layer 3 packets. The Layer 2 entries are created only if the PBF MAC address is set using the **set pbf mac** command before using the **set pbf vlan** command.

To specify the PBF MAC address on a VLAN, perform this task in privileged mode:

Task	Command
Specify the PBF MAC address on a VLAN.	**set pbf vlan** *vlan*

This example shows how to specify the PBF MAC address on a VLAN:

```
Console> (enable) set pbf vlan 11-12
Console> (enable) PBF enabled on vlan(s) 11-12.
Operation successful.
Console> (enable) show pbf
Pbf status    Mac address            Vlans
-----------   ------------------     ---------
ok            00-01-64-f8-39-18      11-12
Console> (enable)
```

The message “Operation successful” indicates that the PBF MAC address was saved in NVRAM.

Entering the **clear pbf** command does not clear the VLANs that are enabled for PBF. The **clear pbf** command clears the Layer 2 table entries that are associated with the VLANs (because the MAC address is no longer valid). You must explicitly clear the PBF-enabled VLANs to remove them from NVRAM by entering the **clear pbf vlan** *vlan_list* command.

Configuring VACLs for PBF

Note Enter the **set security acl adjacency** command to specify the rewrite information in the adjacency table that causes the packet header (the destination VLAN and source and destination MAC addresses) to be rewritten and forwarded to the destination VLAN.

The source MAC address is optional. If you do not specify the source MAC address, the system defaults to the PBF MAC address.

Note You can configure a maximum of 256 adjacency table entries for a VLAN. The maximum number of adjacency table entries is 1023.

Note To enable jumbo frame forwarding using PBF, enter the **mtu** keyword in the **set security acl adjacency** command.

The order of entries in a PBF VACL is important. The adjacency table entry has to be defined in the VACL before the redirect ACE because the redirect ACE uses it to redirect the traffic. You should create entries for PBF VACLs in the following order:

1. Specify the adjacency table entry.
2. Specify the redirect ACE in the PBF VACL that is using the adjacency table entry.
3. Commit the adjacency table entry.
4. Commit the PBF VACL.
5. Map the PBF VACL to a single VLAN or multiple VLANs.

Tip You can combine Steps 3 and 4 by entering the **commit security acl all** command.

Note The same adjacency table entry can be used by more than one redirect ACE.

To specify an adjacency table entry for the PFC2 or PFC3A/PFC3B/PFC3BXL, perform this task in privileged mode:

Task	Command
Specify an adjacency table entry for the PFC2 or PFC3A/PFC3B/PFC3BXL.	**set security acl adjacency** *adjacency_name dest_vlan dest_mac* [[*source_mac*] | [*source_mac* **mtu** *mtu_size*] | [**mtu** *mtu_size*]]

This example shows how to specify the adjacency table entry:

```
Console> (enable) set security acl adjacency ADJ1 11 00-00-00-00-00-0B
ADJ1 editbuffer modified. Use 'commit' command to apply changes.
Console> (enable)
```

This example shows how to create the PBF VACL for VLAN 10 (see Figure 15-10):

```
Console> (enable) set security acl adjacency ADJ1 11 00-00-00-00-00-0B
ADJ1 editbuffer modified. Use 'commit' command to apply changes.
Console> (enable) set security acl ip IPACL1 redirect ADJ1 ip host 10.0.0.1 host 11.0.0.1
IPACL1 editbuffer modified. Use 'commit' command to apply changes.
Console> (enable) set security acl ip IPACL1 permit any
IPACL1 editbuffer modified. Use 'commit' command to apply changes.
Console> (enable) commit security acl adjacency
Commit operation in progress.

Adjacency successfully committed.
Console> (enable) commit security acl IPACL1
ACL commit in progress.
```

```
ACL 'IPACL1' successfully committed.
Console> (enable) set security acl map IPACL1 10
Mapping in progress.

ACL IPACL1 successfully mapped to VLAN 10.
Console> (enable)
```

This example shows how to create the PBF VACL for VLAN 11 (see Figure 15-10):

```
Console> (enable) set security acl adjacency ADJ2 10 00-00-00-00-00-0A
ADJ2 editbuffer modified. Use 'commit' command to apply changes.
Console> (enable) set security acl ip IPACL2 redirect ADJ2 ip host 11.0.0.1 host 10.0.0.1
IPACL2 editbuffer modified. Use 'commit' command to apply changes.
Console> (enable) set security acl ip IPACL2 permit any
IPACL2 editbuffer modified. Use 'commit' command to apply changes.
Console> (enable) commit security acl adjacency
Commit operation in progress.

Adjacency successfully committed.
Console> (enable) commit security acl IPACL2
ACL commit in progress.

ACL 'IPACL2' successfully committed.
Console> (enable) set security acl map IPACL2 11
Mapping in progress.

ACL IPACL2 successfully mapped to VLAN 11.
Console> (enable)
```

Displaying PBF Information

This section describes how to display the PBF-related information.

To display the adjacency table entries, perform one of these tasks in normal mode:

Task	Command
Display the adjacency table entries.	**show security acl info** [*acl_name* | **adjacency** | **all**] [**editbuffer** [*editbuffer_index*]]
Display the PBF adjacency information for all adjacency table entries or a specific adjacency table entry.	**show pbf adjacency** [*adj_name*]
Display the PBF statistics for all adjacency table entries or a specific adjacency table entry.	**show pbf statistics** [*adj_name*]
Display the adjacency-to-VACL mappings for all adjacency table entries or a specific adjacency table entry.	**show pbf map** [*adj_name*]

This example shows how to display the adjacency table entries:

```
Console> show security acl info adjacency
set security acl adjacency ADJ1
-------------------------------------------------
1. 11 00-00-00-00-00-0b

set security acl adjacency ADJ2
-------------------------------------------------
```

```
1. 10 00-00-00-00-00-0a
Console> show pbf adjacency
Index    DstVlan   DstMac             SrcMac             Name
------------------------------------------------------------------
1        11      00-00-00-00-00-0a  00-00-00-00-00-0b     ADJ1
2        10      00-00-00-00-00-0a  00-00-00-00-00-0b     ADJ2
Console> show pbf statistics
Index    DstVlan   DstMac             SrcMac          HitCount(hex)  Name
------------------------------------------------------------------------
1        11      00-00-00-00-00-0a  00-00-00-00-00-0b  0x00000000    ADJ1
2        10      00-00-00-00-00-0a  00-00-00-00-00-0b  0x00000000    ADJ2
Console> show pbf map
Adjacency            ACL
-----------------  --------------------
ADJ1               IPACL1

ADJ2               IPACL2
Console> (enable)
```

Clearing Entries in PBF VACLs

You cannot clear the adjacency table entry before the redirect ACE. You should clear the redirect ACE and the adjacency table entry in PBF VACLs in the following order:

1. Clear the redirect ACE.
2. Commit the PBF VACL.
3. Clear the adjacency table entry.
4. Commit the adjacency table entry.

To clear a PBF adjacency table entry, perform this task in privileged mode:

Task	Command
Clear a PBF adjacency table entry.	**clear security acl adjacency** *adj name*

This example shows how to clear a PBF adjacency table entry:

```
Console> (enable) clear security acl adjacency ADJ1
Adj is in use by a VACL, clear the VACL first then clear adj.
Console> (enable) clear security acl IPACL1
IPACL1 editbuffer modified. Use 'commit' command to save changes.
Console> (enable) commit security acl IPACL1
ACL commit in progress.

ACL 'IPACL1' successfully deleted.
Console> (enable) clear security acl adjacency ADJ1
ADJ1 editbuffer modified. Use 'commit' command to apply changes.
Console> (enable) commit security acl adjacency
Console> (enable) Adjacency committed successfully
Commit operation in progress.

Console> (enable)
```

Rolling Back Adjacency Table Entries in the Edit Buffer

You can clear the adjacency table entries in the edit buffer that were made prior to the last commit by using the **rollback** command. The adjacency table entries are rolled back to their state at the last commit.

To roll back the adjacency table entries in the edit buffer, perform this task in privileged mode:

Task	Command
Roll back the adjacency table entries in the edit buffer.	**rollback security acl** {*acl_name* | **all** | **adjacency**}

This example shows how to roll back the adjacency table entries in the edit buffer:

```
Console> (enable) rollback security acl adjacency
Editbuffer for adjacency info rolled back to last commit state.
Console> (enable)
```

Configuring Hosts for PBF

This section describes the host configuration procedures for the following platforms and operating systems:

- Linux, page 15-98
- Sun Workstation, page 15-99
- MS-Windows/NT/2000 Hosts, page 15-100

Note When a router is not present in the network, you need to specify the static ARP entries on the participating hosts. The host's ARP table maps the IP address of the host device to the MAC address of the PFC2 or PFC3A/PFC3B/PFC3BXL.

Note The IP addresses in the following examples are the IP addresses that are used in Figure 15-10. These IP addresses were randomly selected; make sure that the IP addresses that you use in your network configuration are unique.

Linux

These examples show how to configure the ARP table for the hosts that run the Linux operating system.

This example shows how to configure Host A:

```
arp -s 11.0.0.1 00:11:11:11:11:11 -i eth0
route add 11.0.0.1 eth0
```

This example shows how to configure Host B:

```
arp -s 10.0.0.1 00:11:11:11:11:11 -i eth1
route add 10.0.0.1 eth1
```

Sun Workstation

When using PBF to enable forwarding between two VLANs with the Sun Workstation end hosts, note these limitations when configuring the hosts:

- PBF Limitations

  PBF does not support ARP; you must set a static ARP entry on each Sun Workstation that participates in PBF. Each static ARP entry must point to the PBF MAC address that is mapped to the destination host.

  You must also configure the Sun Workstation to have a gateway. If the Sun Workstation needs to communicate to a different network, you must define the host routes for all networks that go through PBF, and if required, you must define a default gateway.

  For example, if host 10.0.0.1 on VLAN 40 needs to communicate with host 11.0.0.1 on VLAN 50, and assuming the PBF MAC address is 00-11-11-11-11-11, the static ARP entry would be as follows:

  ```
  arp -s 11.0.0.1 00:11:11:11:11:11
  ```

  where 00-11-11-11-11-11 is the PBF MAC address, and 11.0.0.1 is the IP address of the destination host.

- Sun Workstation Limitations

  Sun Workstations do not allow you to set a static ARP entry if the destination is part of a different network (11.x.x.x in this example). This is an ARP limitation in all Sun Workstations. To overcome this problem, you need to define a dummy gateway, which is a host route, and set a static ARP entry pointing to the PBF MAC address that is mapped to the destination host.

  Using the example above, you need to first define a dummy static ARP entry for the gateway. The IP address of the gateway is one of the host addresses within that network as follows:

  ```
  (A) Kubera# arp -s 10.0.0.2  00:11:11:11:11:11
  (B) Kubera# route add host  11.0.0.1 10.0.0.2
  ```

  You need to set only one dummy ARP entry for PBF-related traffic and the host routes for each destination host.

If the number of hosts increase, you need to set the host route entries for each destination host. You can set up a startup file in /etc/rc2.d that has the host route entries for each destination host. Setting up this file prevents you from having to key in all the host route entries after the Sun Workstation is reset or rebooted.

Entries in the file should use this form:

```
Route add host <destination Host IP Address> <dummy gateway IP Address>
```

You need to use the file that contains the host route entries as one of the startup scripts. You can create the file in a directory that has full permissions for the root/superuser, set a soft link pointing to that file in /etc/rc2.d, or create the file in the /etc/rc2.d directory.

MS-Windows/NT/2000 Hosts

You must set the static ARP entries on Windows-based PCs. For Windows-based PCs, you do not need to set up any dummy gateways for switching between the VLANs with PBF.

This example shows how to configure the static ARP entries in Windows-based platforms:

```
C:\> arp -s 11.0.0.1 00-11-11-11-11-11
```

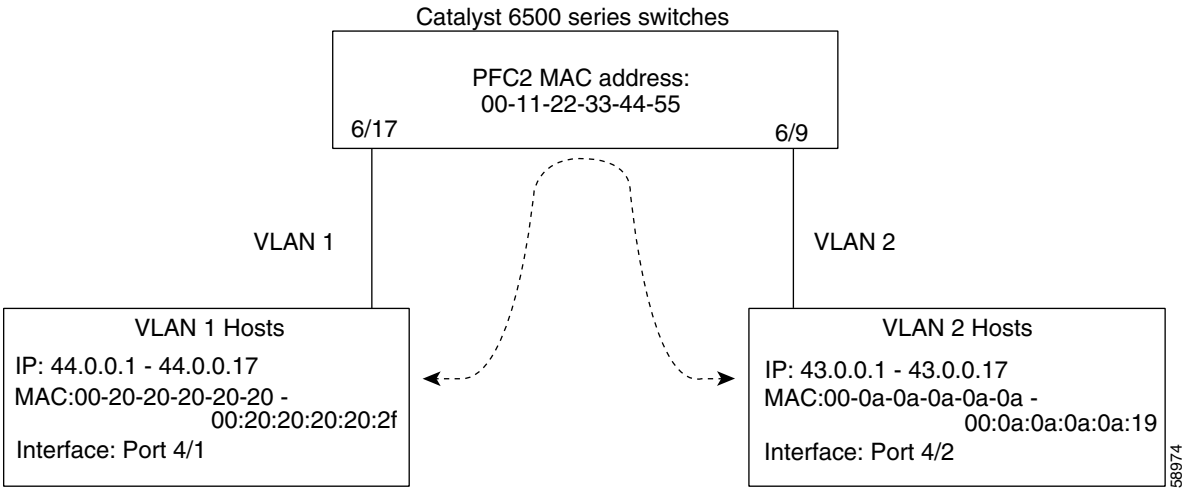
In this example, 00-11-11-11-11-11 is the PBF MAC address and 11.0.0.1 is the IP address of the destination host.

If you need to configure more hosts, you can create a batch file with the ARP entries to each destination host and specify that Windows use this file at startup.

PBF Configuration Example

This section provides the example configurations to enable PBF between the hosts on VLAN 1 and the hosts on VLAN 2 (see Figure 15-11).

Figure 15-11 Policy-Based Forwarding Configuration Example



This example shows the switch configuration file that was created to enable PBF between the hosts on VLAN 1 and VLAN 2. Only the first four hosts from each VLAN are shown in the example (44.0.0.1 through 44.0.0.4 and 43.0.0.1 through 43.0.0.4).

```
#security ACLs
clear security acl all
#adj set
set security acl adjacency a_1 2 00-0a-0a-0a-0a-0a
set security acl adjacency a_2 2 00-0a-0a-0a-0a-0b
set security acl adjacency a_3 2 00-0a-0a-0a-0a-0c
set security acl adjacency a_4 2 00-0a-0a-0a-0a-0d
set security acl adjacency b_1 1 00-20-20-20-20-20
set security acl adjacency b_2 1 00-20-20-20-20-21
set security acl adjacency b_3 1 00-20-20-20-20-22
set security acl adjacency b_4 1 00-20-20-20-20-23
#ip1
```

```
set security acl ip ip1 permit arp
set security acl ip ip1 redirect  a_1  ip host 44.0.0.1 host 43.0.0.1
set security acl ip ip1 redirect  a_2  ip host 44.0.0.2 host 43.0.0.2
set security acl ip ip1 redirect  a_3  ip host 44.0.0.3 host 43.0.0.3
set security acl ip ip1 redirect  a_4  ip host 44.0.0.4 host 43.0.0.4
set security acl ip ip1 permit ip any any
#ip2
set security acl ip ip2 permit arp
set security acl ip ip2 redirect  b_1  ip host 43.0.0.1 host 44.0.0.1
set security acl ip ip2 redirect  b_2  ip host 43.0.0.2 host 44.0.0.2
set security acl ip ip2 redirect  b_3  ip host 43.0.0.3 host 44.0.0.3
set security acl ip ip2 redirect  b_4  ip host 43.0.0.4 host 44.0.0.4
set security acl ip ip2 permit ip any any
#pbf set
set pbf mac 00-11-22-33-44-55
#
commit security acl all
set security acl map ip1 1
set security acl map ip2 2
```

This example shows how to display the MAC addresses that were learned by the switch for port 6/17 on VLAN 1:

```
Console> (enable) show cam dynamic 6/17
* = Static Entry. + = Permanent Entry. # = System Entry. R = Router Entry.
X = Port Security Entry $ = Dot1x Security Entry

VLAN  Dest MAC/Route Des    [CoS]  Destination Ports or VCs / [Protocol Type]
----  ------------------    -----  ------------------------------------------
1     00-20-20-20-20-23             6/17 [ALL]
1     00-20-20-20-20-22             6/17 [ALL]
1     00-20-20-20-20-21             6/17 [ALL]
1     00-20-20-20-20-20             6/17 [ALL]
1     00-20-20-20-20-27             6/17 [ALL]
1     00-20-20-20-20-26             6/17 [ALL]
1     00-20-20-20-20-25             6/17 [ALL]
1     00-20-20-20-20-24             6/17 [ALL]
1     00-20-20-20-20-2b             6/17 [ALL]
1     00-20-20-20-20-2a             6/17 [ALL]
1     00-20-20-20-20-29             6/17 [ALL]
1     00-20-20-20-20-28             6/17 [ALL]
1     00-20-20-20-20-2f             6/17 [ALL]
1     00-20-20-20-20-2e             6/17 [ALL]
1     00-20-20-20-20-2d             6/17 [ALL]
1     00-20-20-20-20-2c             6/17 [ALL]
Total Matching CAM Entries Displayed for 6/17 = 16 for port 6/9, vlan 2
```

This example shows how to display the MAC addresses that were learned by the switch for port 6/9 on VLAN 2:

```
Console> (enable) show cam dynamic 6/9
* = Static Entry. + = Permanent Entry. # = System Entry. R = Router Entry.
X = Port Security Entry $ = Dot1x Security Entry

VLAN  Dest MAC/Route Des    [CoS]  Destination Ports or VCs / [Protocol Type]
----  ------------------    -----  ------------------------------------------
2     00-0a-0a-0a-0a-0e             6/9 [ALL]
2     00-0a-0a-0a-0a-0f             6/9 [ALL]
2     00-0a-0a-0a-0a-0c             6/9 [ALL]
2     00-0a-0a-0a-0a-0d             6/9 [ALL]
2     00-0a-0a-0a-0a-0a             6/9 [ALL]
2     00-0a-0a-0a-0a-0b             6/9 [ALL]
2     00-0a-0a-0a-0a-19             6/9 [ALL]
2     00-0a-0a-0a-0a-18             6/9 [ALL]
2     00-0a-0a-0a-0a-17             6/9 [ALL]
2     00-0a-0a-0a-0a-16             6/9 [ALL]
2     00-0a-0a-0a-0a-15             6/9 [ALL]
2     00-0a-0a-0a-0a-14             6/9 [ALL]
2     00-0a-0a-0a-0a-13             6/9 [ALL]
2     00-0a-0a-0a-0a-12             6/9 [ALL]
2     00-0a-0a-0a-0a-11             6/9 [ALL]
2     00-0a-0a-0a-0a-10             6/9 [ALL]
Total Matching CAM Entries Displayed for 6/9 = 16
```

This example shows how to display the PBF status and the PFC2 or PFC3A/PFC3B/PFC3BXL MAC address:

```
Console> (enable) show pbf
Pbf status    Mac address
-----------   ------------------
ok            00-11-22-33-44-55
```

This example shows how to display the PBF statistics:

```
Console> (enable) show pbf statistics
Index    DstVlan   DstMac             SrcMac          HitCount(hex)  Name
-------------------------------------------------------------------------
1         2      00-0a-0a-0a-0a-0a  00-11-22-33-44-55  0x00026d7c    a_1
2         2      00-0a-0a-0a-0a-0b  00-11-22-33-44-55  0x00026d83    a_2
3         2      00-0a-0a-0a-0a-0c  00-11-22-33-44-55  0x00026d89    a_3
4         2      00-0a-0a-0a-0a-0d  00-11-22-33-44-55  0x00026d90    a_4
5         1      00-20-20-20-20-20  00-11-22-33-44-55  0x000260e3    b_1
6         1      00-20-20-20-20-21  00-11-22-33-44-55  0x000260ea    b_2
7         1      00-20-20-20-20-22  00-11-22-33-44-55  0x000260f1    b_3
8         1      00-20-20-20-20-23  00-11-22-33-44-55  0x000260f8    b_4
```

Enhancements to PBF Configuration (Software Releases 7.5(1) and Later)

This section describes how to configure PBF using the configuration commands that are available in software release 7.5(1) and later releases.

These sections describe the PBF configuration enhancements:

- PBF Configuration Enhancement Overview, page 15-103
- Specifying a PBF_MAP_ACL, page 15-104

PBF Configuration Enhancement Overview

Note The **set** command has changed in software release 8.3(1). For more information, see the "Enhancements to the PBF Configuration (Software Releases 8.3(1) and Later)" section on page 15-105.

The new **set pbf-map** command creates the security ACLs and adjacency information that is based on your input and then automatically commits the ACLs. The **set pbf-map** command involves two steps, as follows:

Step 1 Insert an entry in the adjacency table for each redirect-to-adjacency ACE that is added to the ACL.

Step 2 Create or modify an ACL. This step creates an ACE in each ACL for the redirect-to-adjacency entry, and if necessary, adds a **permit ip any any** ACE to the end of the ACL (this ACE is added only if the **permit ip any any** ACE is not already in the ACL).

The **set pbf-map** command syntax is **set pbf-map** *ip_addr_1 mac_1 vlan_1 ip_addr_2 mac_2 vlan_2*.

An example of the simplified syntax is **set pbf-map 1.1.1.1 0-0-0-0-0-1 11 2.2.2.2 0-0-0-0-0-2 12**.

The new **set pbf-map** command is equivalent to *all* of the following pre-release 7.5(1) commands:

```
set security acl adjacency PBF_MAP_ADJ_0 11 0-0-0-0-0-1
set security acl adjacency PBF_MAP_ADJ_1 12 0-0-0-0-0-2
commit security acl adjacency
set security acl ip PBF_MAP_ACL_11 redirect PBF_MAP_ADJ_1 ip host 1.1.1.1 host 2.2.2.2
set security acl ip PBF_MAP_ACL_12 redirect PBF_MAP_ADJ_0 ip host 2.2.2.2 host 1.1.1.1
```

If the **permit ip any any** ACE is missing, these two **permit ip any any** entries are added**:**

```
set security acl ip PBF_MAP_ACL_11 permit ip any any
set security acl ip PBF_MAP_ACL_12 permit ip any any
commit security acl ip PBF_MAP_ACL_11
commit security acl ip PBF_MAP_ACL_12
set security acl map PBF_MAP_ACL_11 11
set security acl map PBF_MAP_ACL_12 12
```

Each entry in the ACL that is added by the **set pbf-map** command is inserted before the default **permit ip any any** ACE.

If you want to add the entries other than the redirect ACEs to the adjacency table, enter the **set security acl ip PBF_MAP_ACL_(VLAN_ID)** command. The PBF_MAP_ACL_(VLAN_ID) ACL name is based on the following algorithm: The VLAN number of the corresponding host is added to the PBF_MAP_ACL_ string.

Enter the **clear pbf-map** command to delete the redirect-to-adjacency ACEs and adjacency information that is contained in the PBF_MAP_ACL_(VLAN_ID) ACL. Enter the **clear security acl** command to clear all other ACE types that are part of the PBF_MAP_ACL_(VLAN_ID) ACL.

Specifying a PBF_MAP_ACL

Note The ACL name that is used by the **set pbf-map** command is reserved for this command. When you enter the **set security acl** command, you cannot use any name that starts with PBF_MAP_ACL. The name that is used for the adjacency information is also reserved for the **set pbf-map** command. When you enter the **set security acl adjacency** command, you cannot use any name that starts with PBF_MAP_ADJ.

To specify a PBF_MAP_ACL, perform this task in privileged mode:

Task	Command
Specify a PBF_MAP_ACL.	**set pbf-map** *ip_addr_1 mac_1 vlan_1 ip_addr_2 mac_2 vlan_2*

This example shows how to specify a PBF_MAP_ACL:

```
Console> (enable) set pbf-map 1.1.1.1 0-0-0-0-0-1 11 2.2.2.2 0-0-0-0-0-2 22
Commit operation successful.
Commit operation successful.

ACL 'PBF_MAP_ACL_11' successfully committed.
Console> (enable)
ACL PBF_MAP_ACL_11 successfully mapped to VLAN 11.
Console> (enable)
ACL 'PBF_MAP_ACL_22' successfully committed.
Console> (enable)
ACL PBF_MAP_ACL_22 successfully mapped to VLAN 22.
Console> (enable) Operation successful.
Console> (enable)
```

Displaying the PBF_MAP_ACL Information

To display the PBF_MAP_ACL information, perform this task in normal mode:

Task	Command
Display the PBF_MAP_ACL information.	**show pbf-map** {*vlan* | **config**}

This example shows how to display the PBF-related ACEs for the specified VLAN and the statistics for each adjacency used:

```
Console> (enable) show pbf-map 11
Index    DstVlan   DstMac             SrcMac          HitCount(hex)  Name
-------------------------------------------------------------------------
1         22      00-00-00-00-00-02  00-00-00-00-00-00  0x00000000 PBF_MAP_ADJ_1
Console> (enable)
```

This example shows the PBF map configuration:

```
Console> (enable) show pbf-map config
set pbf_map 1.1.1.1 00-00-00-00-00-01 11 2.2.2.2 00-00-00-00-00-02 22
Console> (enable)
```

Clearing the PBF_MAP_ACL Configuration

To clear the PBF_MAP_ACL configuration, perform this task in normal mode:

Task	Command
Clear the PBF_MAP_ACL configuration.	**clear pbf-map all** | **vlan** *vlan* | *ip_addr_1 mac_1 vlan_1 ip_addr_2 mac_2 vlan_2*

This example shows how to clear all the ACLs and adjacency information that were created by the **set pbf-map** command:

```
Console> (enable) clear pbf-map all

ACL 'PBF_MAP_ACL_11' successfully deleted.
Console> (enable)
ACL 'PBF_MAP_ACL_22' successfully deleted.
Console> (enable)
```

This example shows how to clear the ACL with the name PBF_MAP_ACL_VLAN_# and the adjacency table that was used by that ACL:

```
Console> (enable) clear pbf-map vlan 11

ACL 'PBF_MAP_ACL_11' successfully deleted.
Console> (enable) Commit operation successful.
Console> (enable)
```

This example shows how to clear all the ACEs that were created by the **set pbf-map** command except the **permit ip any any** ACE. The command removes the entries that enable the traffic between the hosts with *ip_addr_1* and *ip_addr_2* on *vlan_1* and *vlan_2*. If the entries were already deleted using the **clear security acl** command, a message is displayed indicating that the specific entry was already cleared. The actual entries that were deleted are two ACEs (redirect-to-adjacency ACEs) and two entries in the adjacency table.

```
Console> (enable) clear pbf-map 1.1.1.1 0-0-0-0-0-1 11 2.2.2.2 0-0-0-0-0-2 22

ACL 'PBF_MAP_ACL_11' successfully committed.
Console> (enable)
ACL 'PBF_MAP_ACL_22' successfully committed.
Console> (enable)
```

Enhancements to the PBF Configuration (Software Releases 8.3(1) and Later)

This section describes how to configure PBF using two new configuration commands (**set pbf client** and **set pbf gw**) that are available in software release 8.3(1) and later releases. The PBF enhancements that are described in this section simplify the process of setting and committing the security ACLs and adjacency information. The enhanced **set pbf-map** command creates the security ACLs and adjacency information based on your input, commits them to the hardware, and maps them to the VLANs. As part of creating the necessary VACLs to redirect the traffic from one VLAN to another, the ARP packets are redirected to the software and the supervisor engine generates the ARP replies for the gateway/client requests.

These sections describe the PBF configuration enhancements:

PBF Usage Guidelines and Restrictions

This section describes the usage guidelines and restrictions for configuring PBF:

- With Supervisor Engine 720, you must specify the VLAN that you are enabling PBF on by entering the **set pbf vlan** *vlan* command. For more information, see the "Specifying the PBF MAC Address on a VLAN" section on page 15-94.
- The clients and gateways must be on different VLANs and no clients or gateways can have the same IP address. The maximum number of entries is 1024.
- The client name and gateway name must be no more than 12 characters.
- If you create a PBF map between two VLANs that already have the VACLs attached, the PBF ACLs overwrite the previous configuration. The opposite is also true. If you have created the PBF ACLs by entering the **set pbf-map** command and the PBF ACLs are attached to the VLANs, if you decide to map a new VACL to the same VLANs, the new VACL overwrites the previous configuration.

Setting and Committing Security ACLs and Adjacency Information

The new **set pbf client** command adds the new hosts to the current list. The new **set pbf gw** command is used to add a gateway to handle the interVLAN connections. The enhanced **set pbf-map** command creates two ACLs, client_name and gateway_name, commits the newly created entries to the hardware, and maps them to the VLANs.

To create a PBF map, perform these steps:

Step 1 Add the clients and gateways to their respective lists, as follows:

a. **set pbf client** *client_name ip_addr mac_addr vlan*

b. **set pbf gw** *gateway_name ip_addr ip_mask mac_addr vlan*

Step 2 Map the client list to the gateway list, as follows:

set pbf-map *client_name gateway_name*

Note The number of PBF-client groups that can be mapped to a single PBF gateway is dependent on the number of ACLs that you have already configured. The maximum number of supported ACLs is 250, so if you already have 20 ACLs defined, you can have 229 client groups mapped to a gateway.

An example is as follows:

```
Console> (enable) set pbf client cl1 21.1.1.1 00-00-00-00-40-01 101
Commit operation successful.
Console> (enable) set pbf gw gw1 21.0.0.128 255.0.0.0 00-a0-c9-81-e1-13 102
Commit operation successful.
Console> (enable) set pbf-map cl1 gw1
.ccl1 editbuffer modified. Use 'commit' command to apply changes.
.ggw1 editbuffer modified. Use 'commit' command to apply changes.
.ccl1 editbuffer modified. Use 'commit' command to apply changes.
.ggw1 editbuffer modified. Use 'commit' command to apply changes.
.ccl1 editbuffer modified. Use 'commit' command to apply changes.
.ggw1 editbuffer modified. Use 'commit' command to apply changes.
Console> (enable) ACL commit in progress.
Console> (enable) ACL commit in progress.

ACL '.ccl1' successfully committed.
Console> (enable)
ACL '.ggw1' successfully committed.
Console> (enable) Mapping in progress.
Please configure VLAN 101.

ACL .ccl1 successfully mapped to VLAN 101.
Console> (enable) Mapping in progress.
Please configure VLAN 102.

ACL .ggw1 successfully mapped to VLAN 102.
Console> (enable)
```

The new and enhanced command set is equivalent to *all* of the following commands:

```
#adj set
set security acl adjacency .c0001cl1 101 00-00-00-00-40-01 21.1.1.1
set security acl adjacency .g0002gw1 102 00-a0-c9-81-e1-13 21.0.0.128 7
#.ccl1
set security acl ip .ccl1 permit arp
set security acl ip .ccl1 permit arp-inspection any any
set security acl ip .ccl1 redirect .g0002gw1 ip host 21.1.1.1 any
set security acl ip .ccl1 permit ip any any
#.ggw1
set security acl ip .ggw1 permit arp
set security acl ip .ggw1 permit arp-inspection any any
set security acl ip .ggw1 redirect .c0001cl1 ip any host 21.1.1.1
set security acl ip .ggw1 permit ip any any
#
commit security acl all
set security acl map .ccl1 101
set security acl map .ggw1 102
```

Each entry in the ACL that is added by the **set pbf-map** command is inserted before the default **permit ip any any** ACE. If you want to add entries other then redirect to the adjacency, enter the **set security acl ip client_name** or **gateway_name** commands. The ARP-inspection entry can be replaced with a more specific one. The ARP reply is generated only after the ARP-inspection ACEs are verified. If you want to allow only some clients to get the ARP reply, the new ARP-inspection entries have to be set.

clear Commands

The **clear pbf client** command cannot be used to remove the last remaining PBF client without first removing the PBF map. To clear a single client or all clients from the list, perform this task in normal mode:

Task	Command
Clear a single client or all clients.	**clear pbf** {**client** | **gw**} *name* [*ip_addr*]

This example shows how to clear a PBF client:

```
Console> (enable) clear pbf client cl1
.c0001cl1 editbuffer modified. Use 'commit' command to apply changes.
Console> (enable) Commit operation successfull.
Console> (enable)
```

The **clear pbf gw** command cannot be used to remove the last remaining PBF gateway without first removing the PBF map. To clear a single gateway or all gateways, perform this task in normal mode:

Task	Command
Clear a single gateway or all gateways.	**clear pbf** {**client** | **gw**} *name* [*ip_addr*]

This example shows how to clear a PBF gateway:

```
Console> (enable) clear pbf gw gw1
.g0002gw1 editbuffer modified. Use 'commit' command to apply changes.
Commit operation successfull.
Console> (enable)
```

To clear the PBF mapping, perform this task in normal mode:

Task	Command
Clear the PBF mapping.	**clear pbf-map** *client_name gw_name*

This example shows how to clear the PBF mapping:

```
Console> (enable) clear pbf-map cl1 gw1
.ccl1 editbuffer modified. Use 'commit' command to save changes.
.ggw1 editbuffer modified. Use 'commit' command to save changes.
Console> (enable) ACL commit in progress.
Console> (enable) ACL commit in progress.

ACL '.ccl1' successfully deleted.
Console> (enable)
ACL '.ggw1' successfully deleted.
Console> (enable)
```

show Commands

To display all the PBF maps, perform this task in normal mode:

Task	Command
Display all the PBF maps.	**show pbf-map**

This example shows how to display all the PBF maps:

```
Console> (enable) show pbf-map
PBF MAP
Clients             Gatways
-------------------------------------------
cl1                   gw1
Console> (enable)
```

To display the PBF client configuration, perform this task in normal mode:

Task	Command
Display the PBF client configuration.	**show pbf client** [*client_name* | *ip_addr*]

This example shows how to display the PBF client configuration:

```
Console> (enable) show pbf client
Client    : cl1
Map       : gw1
VLAN      : 101
Adjacency          ip           mac
-----------------------------------------
.c0001cl1          21.1.1.1     00-00-00-00-40-01


Console> (enable)
```

To display the PBF-gateway configuration, perform this task in normal mode:

Task	Command
Display the PBF-gateway configuration.	**show pbf gw** [*gw_name* | *ip_addr*]

This example shows how to display the PBF-gateway configuration:

```
Console> (enable) show pbf gw
Client    : gw1
Map       : cl1
VLAN      : 102
Adjacency          ip           mask           mac
-----------------------------------------------------------------
.g0002gw1          21.0.0.128   255.0.0.0    00-a0-c9-81-e1-13


Console> (enable)
```

Using the sc1 Interface as a Diagnostic Interface

To temporarily place the sc1 interface in a PBF-client VLAN to test the connection between your switch and a customer's switch or router, perform these steps:

Step 1 Enter the **clear pbf arp-inspection** *list_name* command to remove the ARP-inspection ACL statement from the PBF-client VLAN in which the test is to be conducted.

To verify that an ARP-inspection ACE is set (or not set) on the ACL for a client list or a gateway, enter the **show pbf arp-inspection** command.

Step 2 Enter the **set interface sc1** command to assign the sc1 interface to the customer's VLAN and assign it an IP address in the same IP subnet as the customer's router or switch.

Step 3 Enter the **ping** command to test connectivity between the Catalyst 6500 series switch (sourced from interface sc1) and the customer's router or switch. The sc1 interface sends the ARP requests for the customer's MAC address, and the customer's router or switch responds. If and when the customer's device sends out an ARP response before sending an ICMP reply, the sc1 interface responds with the MAC address.

Step 4 After testing is completed, reconfigure the sc1 interface so that it is no longer a part of the customer's VLAN.

Step 5 Enter the **set pbf arp-inspection** *list_name* command to restore the ARP-inspection ACL statement to the PBF-client VLAN.

Enhancements to PBF Configuration (Software Releases 8.6(1) and Later)

With software release 8.6(1) and later releases, the PBF macro commands are retained in the switch configuration file. This enhancement allows you to operate and manage the switch by showing which **set** commands were used to create the security and adjacency ACLs that are associated with the PBF clients, gateways, and maps. The configuration file also retains the **clear** commands that were used to clear particular PBF clients, gateways, and maps.

The PBF configuration enhancements are described in these sections:

- Configuring the PBF Before Software Release 8.6(1), page 15-111
- Configuring PBF in Software Release 8.6(1) and Later Releases, page 15-114

Configuring the PBF Before Software Release 8.6(1)

To configure a PBF with a software release before release 8.6(1), follow these steps:

Step 1 Configure the PBF MAC address for the PFC and enable PBF.

```
Console> (enable) set pbf
PBF committed successfully.
Operation successful.
Console> (enable)
Console> (enable) show pbf
Pbf status    Mac address         Vlans
------------  ------------------  ------------------
ok            00-0d-65-36-1e-eb
Console> (enable)
```

Step 2 Configure one PBF client called CLIENT-TEST.

```
Console> (enable) set pbf client CLIENT-TEST 10.0.0.10 00-00-11-11-22-22 10
Commit operation successful.
Console> (enable)
The PBF client has been created
Console> (enable)

Console> (enable) show pbf client
Name      : CLIENT-TEST
Map       : No map
VLAN      : 10
Clients   : 1
Adjacency          ip           mac
-----------------------------------------------
.c0000CLIENT-TEST  10.0.0.10    00-00-11-11-22-22
Console> (enable)
```

The **set pbf client** command macro has created the security ACL adjacency for the client, but the macro command (**set pbf client CLIENT-TEST 10.0.0.10 00-00-11-11-22-22 10**) that created the security ACL adjacency does not appear in the following configuration:

```
Console> (enable) show run
<SNIP>  Unrelated configuration information cut out

!
#security ACLs
 clear security acl all
#pbf set
set pbf mac 00-0d-65-36-1e-eb
#adj set
set security acl adjacency .c0000CLIENT-TEST 10 00-00-11-11-22-22 10.0.0.10
#
 commit security acl all
!
<SNIP>  Unrelated configuration information cut out

Console> (enable)
```

Step 3 Configure one PBF gateway called GATEWAY-TEST.

```
Console> (enable) set pbf gw GATEWAY-TEST 10.0.0.100 255.255.255.0 11-11-22-22-33-3 3 1
Commit operation successful.
Console> (enable)
```

The following PBF gateway has been created:

```
Console> (enable) show pbf gw
Name      : GATEWAY-TEST
Map       : CLIENT-TEST,
VLAN      : 1
Gateways  : 1
Adjacency          ip           mask           mac
-----------------------------------------------------------------
.g0001GATEWAY-TEST 10.0.0.100   255.255.255.0 11-11-22-22-33-33

Console> (enable)
```

The **set pbf gateway** command macro has created the security ACL adjacency for the gateway, but the macro command (**set pbf client GATEWAY-TEST 10.0.0.100 255.255.255.0 11-11-22-22-33-33 1**) that created the security ACL adjacency does not appear in the following configuration:

```
Console> (enable) show run
<SNIP>  Unrelated configuration information cut out

!
#security ACLs
 clear security acl all
#pbf set
set pbf mac 00-0d-65-36-1e-eb
#adj set
set security acl adjacency .c0000CLIENT-TEST 10 00-00-11-11-22-22 10.0.0.10
set security acl adjacency .g0001GATEWAY-TEST 1 11-11-22-22-33-33 10.0.0.100 23
#
 commit security acl all
!

<SNIP>  Unrelated configuration information cut out

Console> (enable)
```

Step 4 Build the PBF map between the client (CLIENT-TEST) and the gateway (GATEWAY-TEST).

```
Console> (enable) set pbf-map CLIENT-TEST GATEWAY-TEST
.cCLIENT-TEST editbuffer modified. Use 'commit' command to apply changes.
.cCLIENT-TEST editbuffer modified. Use 'commit' command to apply changes.
.gGATEWAY-TEST editbuffer modified. Use 'commit' command to apply changes.
.cCLIENT-TEST editbuffer modified. Use 'commit' command to apply changes.
.gGATEWAY-TEST editbuffer modified. Use 'commit' command to apply changes.
6509> (enable) ACL commit in progress.
ACL commit in progress.

ACL '.cCLIENT-TEST' successfully committed.
Console> (enable)

ACL '.gGATEWAY-TEST' successfully committed.
Console> (enable) Mapping in progress.
Please configure VLAN 10.

ACL .cCLIENT-TEST successfully mapped to VLAN 10.
Console> (enable) Mapping in progress.

ACL .gGATEWAY-TEST successfully mapped to VLAN 1.
```

```
Console> (enable)
```

The PBF client is now mapped to the PBF gateway as follows:

```
Console> (enable) show pbf client
Name      : CLIENT-TEST
Map       : GATEWAY-TEST,
VLAN      : 10
Clients   : 1
Adjacency          ip           mac
-------------------------------------------------
.c0000CLIENT-TEST  10.0.0.10    00-00-11-11-22-22
Console> (enable)
```

The PBF gateway is now mapped to the PBF client as follows:

```
Console> (enable) show pbf gw
Name      : GATEWAY-TEST
Map       : CLIENT-TEST,
VLAN      : 1
Gateways  : 1
Adjacency          ip           mask           mac
-------------------------------------------------------------------
.g0001GATEWAY-TEST 10.0.0.100   255.255.255.0 11-11-22-22-33-33
Console> (enable)
```

The PBF map has been built as follows:

```
Console> (enable) show pbf-map
PBF MAP
Clients             Gateways
-----------------------------------------
CLIENT-TEST           GATEWAY-TEST
Console> (enable)
```

The **set pbf-map macro** command has created security ACL IP lists and security ACL map lists for the PBF client and PBF gateway, but the macro command (**set pbf-map CLIENT-TEST GATEWAY-TEST**) that created these security ACLs does not appear in the following configuration:

```
Console> (enable) show run
<SNIP>  Unrelated configuration information cut out

!
#security ACLs
 clear security acl all
#pbf set
set pbf mac 00-0d-65-36-1e-eb
#adj set
set security acl adjacency .c0000CLIENT-TEST 10 00-00-11-11-22-22 10.0.0.10
set security acl adjacency .g0001GATEWAY-TEST 1 11-11-22-22-33-33 10.0.0.100 23
#.cCLIENT-TEST
set security acl ip .cCLIENT-TEST permit arp
set security acl ip .cCLIENT-TEST permit arp-inspection any any
set security acl ip .cCLIENT-TEST redirect .g0001GATEWAY-TEST ip host 10.0.0.10
any
set security acl ip .cCLIENT-TEST permit ip any any
#.gGATEWAY-TEST
set security acl ip .gGATEWAY-TEST permit arp
set security acl ip .gGATEWAY-TEST redirect .c0000CLIENT-TEST ip any host 10.0.0
.10
set security acl ip .gGATEWAY-TEST permit ip any any
#
 commit security acl all
set security acl map .cCLIENT-TEST 10
```

```
set security acl map .gGATEWAY-TEST 1
!

<SNIP>  Unrelated configuration information cut out

Console> (enable)
```

Looking at the above configuration, it is not obvious that the following three commands were used to create the adjacency and security ACLs on the switch:

- **set pbf client CLIENT-TEST 10.0.0.10 00-00-11-11-22-22 10**
- **set pbf gateway GATEWAY-TEST 10.0.0.100 255.255.255.0 11-11-22-22-33-33 1**
- **set pbf-map CLIENT-TEST GATEWAY-TEST**

This problem is further complicated as more and more PBF clients and PBF maps are added to the configuration. It is also not obvious which commands were used to clear the PBF configurations. In the PBF map for the above configuration, the **clear** commands are as follows:

- **clear pbf client CLIENT-TEST 10.0.0.10**
- **clear pbf gateway GATEWAY-TEST 10.0.0.100**
- **clear pbf-map CLIENT-TEST GATEWAY-TEST**

Listing the **set** and **clear** commands for the PBF clients, PBF gateways, and PBF maps, allows you to manage, configure, and use the PBF map feature.

Configuring PBF in Software Release 8.6(1) and Later Releases

To configure PBF with a software release before release 8.6(1), follow these steps:

Step 1 Configure the PBF MAC address for the PFC and enable PBF.

```
Console> (enable) set pbf
Operation successful.

Console> (enable) show pbf
Pbf status    Mac address         Vlans
------------  ------------------  ------------------
ok            00-0d-65-35-ed-83
```

Step 2 Configure one PBF client called CLIENT-TEST.

```
Console> (enable) set pbf client CLIENT-TEST 10.0.0.10 00-00-11-11-22-22 10
Commit operation successful.

Console> (enable) show pbf client
Name      : CLIENT-TEST
Map       : No map
VLAN      : 10
Clients   : 1
Adjacency          ip           mac
-------------------------------------------------
.c0000CLIENT-TEST  10.0.0.10    00-00-11-11-22-22
```

Step 3 Configure one PBF gateway called GATEWAY-TEST.

```
Console> (enable) set pbf gw GATEWAY-TEST 10.0.0.100 255.255.255.0 11-11-22-22-3 3-3 3 1
Commit operation successful.

Console> (enable) show pbf gw
Name      : GATEWAY-TEST
Map       : No map
VLAN      : 3
Gateways  : 1
Adjacency          ip           mask           mac
-------------------------------------------------------------
.g0001GATEWAY-TEST 10.0.0.100   255.255.255.0 11-11-22-22-33-03
```

Step 4 Build the PBF map between the client (CLIENT-TEST) and the gateway (GATEWAY-TEST).

```
Console> (enable) set pbf-map CLIENT-TEST GATEWAY-TEST
.cCLIENT-TEST editbuffer modified. Use 'commit' command to apply changes.
.cCLIENT-TEST editbuffer modified. Use 'commit' command to apply changes.
.gGATEWAY-TEST editbuffer modified. Use 'commit' command to apply changes.
.cCLIENT-TEST editbuffer modified. Use 'commit' command to apply changes.
.gGATEWAY-TEST editbuffer modified. Use 'commit' command to apply changes.
Console> (enable) ACL commit in progress.
ACL commit in progress.
Console> (enable) Console> (enable) Mapping in progress.
Please configure VLAN 10.

ACL .cCLIENT-TEST successfully mapped to VLAN 10.
Console> (enable) Mapping in progress.
Please configure VLAN 3.

ACL .gGATEWAY-TEST successfully mapped to VLAN 3.

Console> (enable) show pbf client
Name      : CLIENT-TEST
Map       : GATEWAY-TEST,
VLAN      : 10
Clients   : 1
Adjacency          ip           mac
-------------------------------------------------
.c0000CLIENT-TEST  10.0.0.10    00-00-11-11-22-22

Console> (enable) show pbf gw
Name      : GATEWAY-TEST
Map       : CLIENT-TEST,
VLAN      : 3
Gateways  : 1
Adjacency          ip           mask           mac
-------------------------------------------------------------
.g0001GATEWAY-TEST 10.0.0.100   255.255.255.0 11-11-22-22-33-03

Console> (enable) show pbf map
Adjacency            ACL
------------------  --------------------
.c0000CLIENT-TEST    .gGATEWAY-TEST

.g0001GATEWAY-TEST   .cCLIENT-TEST
```

Step 5 Display the PBF configuration commands.

```
Console> (enable) show run
<SNIP>  Unrelated configuration information cut out

!
#security ACLs
clear security acl all
#pbf set
set pbf mac 00-0d-65-35-ed-83
#set pbf client
set pbf client CLIENT-TEST 10.0.0.10 00-00-11-11-22-22 10
#set pbf gw
set pbf gw GATEWAY-TEST 10.0.0.100 255.255.255.0 11-11-22-22-33-03 3
#set pbf-map
set pbf-map CLIENT-TEST GATEWAY-TEST
#
commit security acl all
!
<SNIP>  Unrelated configuration information cut out

Console> (enable)
```

Downloadable ACLs

Downloadable ACLs are a set of ACEs that are configured on a RADIUS server. Downloadable ACLs are downloaded during authentication of a NAC feature such as Dot1x, mac-auth, LPIP, or web-auth.

Downloadable ACLs are a port-based feature. You will need to configure the security ACL so that it is port based and map an ACL with an include keyword to the port. Do not reconfigure the security ACL with the include keyword once it has been mapped to the port. Make sure to clear the security ACL with the include keyword if you make any modifications.

Once authentication is successful, a downloaded ACL is initiated with DHCP snooping, ARP inspection, or static DHCP bindings. The set of ACEs that were downloaded get recommitted as system-generated ACLs along with ACLs that were mapped to the port. For example, an ACL that was mapped to a port and a downloaded ACL are remapped to the port at runtime. The downloaded ACLs are placed in the **include downloaded-acl** *feature* ACE.

The following sections describe how to configure and display information about downloaded ACLs. Downloadable ACLs can only be mapped to ports with a port-based security ACL mode.

Note Downloadable ACLs are only supported on switches that feature a Supervisor Engine 720 or Supervisor Engine 32.

Note DNS hostnames are supported in the ACEs of downloadable ACLs from RADIUS servers. Make sure to enable DNS.

Note If your downloaded ACL is larger than 4 KB, enable IP reassembly by using the **set ip reassembly enable** command.

Configuring a Downloaded ACL for dot1x

To configure a downloaded ACL for dot1x without an IP phone, perform these steps:

Step 1 Create a base ACL with an include dot1x keyword.

```
Console> (enable) set security acl ip dacl1x permit arp-inspection any any
dacl1x editbuffer modified. Use 'commit' command to apply changes.
Console> (enable) set security acl ip dacl1x permit dhcp-snooping
Successfully configured DHCP Snooping for ACL dacl1x. Use 'commit' command to save
changes.
Console> (enable) set security acl ip dacl1x include downloaded-acl dot1x
Successfully configured placeholder download ACL dacl1x. Use 'commit' command to save
changes.
Console> (enable) commit security acl all
Commit operation in progress.
```

Step 2 Set the security-acl mode on the port used for authentication to port-based mode.

```
Console> (enable) set port security-acl 5/35 port-based
Warning: Vlan-based ACL features will be disabled on ports 5/35
ACL interface is set to port-based mode for port(s) 5/35.
```

Step 3 Map the base ACL (with the include keyword) to that port.

```
Console> (enable) set security acl map dacl1x 5/35
Mapping in progress.
ACL dacl1x successfully mapped to port(s) 5/35
```

Step 4 Enable dot1x globally and on that port.

```
Console> (enable) set dot1x system-auth-control enable
Dot1x is globally enabled.
Configured RADIUS servers will be used for dot1x authentication.
Console> (enable) set port dot1x 5/35 port-control auto
Port 5/5 dot1x port-control is set to auto.
Trunking disabled for port 5/35 due to Dot1x feature.
Spantree port fast start option enabled for port 5/35.
```

Step 5 Display the port security settings for the configured port.

```
Console> (enable) show port security-acl 5/35
Port  Interface Type Interface Type Interface Merge Status
      config         runtime        runtime
----- -------------- -------------- ----------------------
 5/35     port-based     port-based         not applicable

Config:
Port  ACL name                         Type
----- -------------------------------- ----
 5/35 dacl1x                           IP

Runtime:
Port  ACL name                         Type
----- -------------------------------- ----
 5/35 dacl1x                           IP

dhcp-snooping:
Port     Trust        Source-Guard    Source-Guarded IP Addresses
-----    -----------  ------------    ---------------------------
 5/35    untrusted    disabled

Port     Binding Limit         No. of Existing Bindings
```

```
-----    ------------------    ------------------------
 5/35    32                    0
```

Step 6 Authenticate the dot1x port and that the downloadable ACL is downloaded and the child ACL is generated. Check the authentication status.

```
Console> (enable) show port dot1x 5/35
Port  Auth-State          BEnd-State Port-Control        Port-Status
----- ------------------- ---------- ------------------- -------------
 5/35 authenticated       idle       auto                authorized

Port  Port-Mode     Re-authentication   Shutdown-timeout   Control-Mode
                                                           admin   oper
----- ------------ -----------------   ----------------   ---------------
 5/35 SingleAuth    disabled            disabled           Both    Both

Port  Posture-Token Critical-Status Termination action Session-timeout
----- ------------ --------------- ------------------ ---------------
 5/35 -            no              NoReAuth           -

Port  Session-Timeout-Override Url-Redirect
----- ------------------------ ---------------------------------
 5/35 disabled                 -

Port  Critical  Port-Name
----- --------  ---------
 5/35 disabled  -

Port  Downloaded ACL
----- ----------------------------------------
 5/35 ACSACL#-IP-test-44bb6f49
```

Note If the dot1x Auth-state is in the ipawaiting state, add IP to the host (through DHCP or ARP or the addition of static DHCP snooping bindings). A downloadable ACL will be downloaded and a child ACL will be created.

If an MSFC is the router, to obtain DHCP-snooping bindings, map the DHCP-snooping ACL to the authenticated host VLAN. If an external router configuration is used, map the DHCP-snooping ACL to the host and DHCP-server port.

Sample Output of show Commands

The following sample outputs of **show** commands that are used for displaying the child ACL and downloaded ACL after authentication:

- Displays the system-generated ACL information:

```
Console> (enable) show security acl info dacl1x_5_35
set security acl ip dacl1x_5_35
---------------------------------------------------
arp permit
1. permit arp-inspection any any
2. permit dhcp-snooping
3. permit ip host 9.6.6.104 10.76.255.85 255.255.255.0
4. deny ip host 9.6.6.104 64.104.129.189 255.255.0.0
5. permit tcp host 9.6.6.104 eq 21 host 10.76.255.25
6. deny ip host 9.6.6.104 6.104.129.189 255.255.0.0
```

```
7. deny ip host 9.6.6.104 67.104.129.189 255.255.0.0
8. include downloaded-acl dot1x
```

- Displays the dot1x user all O/P:

```
Console> (enable) show dot1x user all
Username                         Mod/Port   UserIP      VLAN
---------------------            --------   ------      ------
host                              5/35      9.6.6.104   16

Downloaded ACL
-----------------------------------------------------------
ACSACL#-IP-test-44bb6f49

Derived ACL
--------------------------------
dacl1x_5_35
```

- Checks the DACL name:

```
Console> (enable) show security acl downloaded-acl all
Downloaded ACL Summary:
    ACL Name                                  Date/Time
-------------------------------------------------------------------------
  1.#ACSACL#-IP-test-44bb6f49                 Fri Jul 21 2006, 05:05:58
```

Displays the user-mapped IP, port, and the feature:

```
Console> (enable) show security acl downloaded-acl user-map
Downloaded ACL User Map:
ACL Name : #ACSACL#-IP-test-44bb6f49
User Count : 1
Num of Aces : 5
     Ip Address                          mNo/pNo     Feature
-------------------------------------------------------------
  1. 9.6.6.104                             5/35      dot1x
```

- Displays the DACL information specific to the port:

```
Console> (enable) show security acl downloaded-acl port 5/35
Port  IP Address       Feature    Downloaded ACL
----- ---------------- ---------- ------------------------------------
 5/35 9.6.6.104         dot1x       #ACSACL#-IP-test-44bb6f49
```

- Displays the ACEs that were downloaded from the RADIUS server:

```
Console (enable) show security acl downloaded-acl #ACSACL#-IP-test-44bb6f49
Downloaded ACE's for #ACSACL#-IP-test-44bb6f49:
permit ip any 10.76.255.85 255.255.255.0
deny ip any 64.104.129.189 255.255.0.0
permit tcp any eq 21 host 10.76.255.25
deny ip any 6.104.129.189 255.255.0.0
deny ip any 67.104.129.189 255.255.0.0
```

Configuring a Downloaded ACL for Dot1x for an IP Phone

To configure a downloaded ACL for dot1x with an IP phone, perform these steps:

Step 1 Grant permission for the IP phone by configuring the base-ACL.

```
Console> (enable) set security acl ip dacl1x permit arp-inspection any any
```

```
dacl1x editbuffer modified. Use 'commit' command to apply changes.
Console> (enable) set security acl ip dacl1x permit dhcp-snooping
Successfully configured DHCP Snooping for ACL dacl1x. Use 'commit' command to save
changes.
Console> (enable) set security acl ip dacl1x include downloaded-acl dot1x
Successfully configured placeholder download ACL dacl1x. Use 'commit' command to save
changes.
Console> (enable) set security acl ip dacl1x include ip-phone
Successfully configured placeholder download ACL dacl1x. Use 'commit' command to save
changes.
Console> (enable) commit security acl all
Commit operation in progress.
```

Step 2 Display the child ACL with an IP phone configured.

```
Console> (enable) show security acl downloaded-acl ipphone-map
Port  IP Address
----- ------------------------------
 4/1   9.6.6.135
Console> (enable) show security acl tcam interface 4/1
Input
IP
0. redirect arp (matches 0)
1. redirect udp any any (matches 0)
2. redirect udp any 21862 host 9.6.6.3 53000 (matches 0)
3. redirect tcp any any 80 (matches 0)
4. permit ip host 9.6.6.135 any (matches 10)
5. deny ip any any (matches 0)

Console> (enable) show security acl info dacl_4_1
set security acl ip dacl_4_1
---------------------------------------------------
arp permit
1. permit arp-inspection host 9.2.2.2 any
2. permit dhcp-snooping
3. permit eapoudp
4. include downloaded-acl web-auth
5. permit url-redirect
6. permit ip host 9.6.6.135 any
7. include ip-phone
8. include downloaded-acl dot1x
9. include downloaded-acl macauth-bypass
10. include downloaded-acl eou
```

Note In the above sample outputs, the child ACL has only an IP phone ACE expanded. No feature (dot1x, Mac-auth, LPIP, Webauth) is enabled on the port.

Creating a Placeholder for a Downloaded ACL

To create a placeholder for a downloaded ACL, perform this task in enable mode:

Task	Command
Create a placeholder for a downloaded ACL.	**set security acl ip test include downloaded-acl** *feature*

The feature variable can be one of the following:

- dot1x
- webauth
- macauth-bypass
- eou

This example shows how to create a placeholder for a downloaded ACL:

```
Console> set security acl ip test include downloaded-acl dot1x
Console> Successfully configured placeholder download ACL test. Use
  'commit' command to save changes.
Console> show security acl info test
set security acl ip test
------------------------------------------------
1. permit arp-inspection
2. permit eapoudp
3. include downloaded-acl dot1x
4. permit url-redirect
5. deny ip any any
```

Creating a Placeholder for an IP Phone

To create a placeholder for an IP phone, perform this task in enable mode:

Task	Command
Create a placeholder for an IP phone.	**set security acl ip test include ip-phone**

This example shows how to create a placeholder for an IP phone:

```
Console> (enable) set security acl ip test include ip-phone
Successfully configured placeholder download ACL test. Use 'commit' command to save
changes.
```

Displaying Downloaded ACL Information

To display downloaded ACL information, perform this task in enable mode:

Task	Command
Display downloaded ACL information.	**show security acl downloaded-acl** *all*

This example shows how to display downloaded ACL information:

```
Console> (enable) show security acl downloaded-acl all
Downloaded ACL Summary:
    ACL Name                          Date/Time
-------------------------------------------------------------
  1.#ACSACL#-IP-test_acl2-44cf4bcd    Tue Aug 1 2006, 03:14:54
  2.#ACSACL#-IP-lpipacl-44a100c7      Tue Aug 1 2006, 03:04:56
```

To display detailed information about a downloaded ACL, perform this task in enable mode:

Task	Command
Display detailed information about a downloaded ACL.	**show security acl downloaded-acl** *ACL name*

This example shows detailed information about a downloaded ACL:

```
Console> (enable) show security acl downloaded-acl #ACSACL#-IP-test_acl2-44cf4bcd
Downloaded ACE's for #ACSACL#-IP-test_acl2-44cf4bcd :
  1. permit ip any host 10.1.1.1
  2. permit tcp any host 100.1.1.3
  3. permit udp any host 10.76.88.34
  4. deny ip any host 9.6.5.7
  5. deny tcp any host 2.3.4.5
  6. deny udp any host 3.4.5.5
  7. permit icmp any host 100.1.1.5
```

To display detailed the mapping between the user and an ACL, perform this task in enable mode:

Task	Command
Display the mapping between a user and an ACL.	**show security-acl downloaded-acl user-map**

This example shows how to display mapping information about a downloaded ACL:

```
Console> (enable) show security acl downloaded-acl user-map
Downloaded ACL User Map:
ACL Name : #ACSACL#-IP-test_acl2-44cf4bcd
User Count : 1
Num of Aces : 7
     Ip Address                          mNo/pNo     Feature
-------------------------------------------------------------
  1. 10.1.1.5                              3/13      dot1x
```

To display the host information on a port, perform this task in enable mode:

Task	Command
Display the host information on a port.	**show security acl downloaded-acl port** *mod/port*

This example shows how to display host information on a port:

```
Console> (enable) show security acl downloaded-acl port 3/45
Port  IP Address       Feature    Downloaded ACL
----- ---------------- ---------- ---------------------------
3/45  9.6.2.233        dot1x      #ACSACL#-IP-testacl-44c7197a
```

To display the IP phone information that has been detected on a port on which downloaded ACLs are present, perform this task in enable mode:

Task	Command
Display the IP phone information on a port.	**show security acl downloaded-acl ipphone-map**

This example shows how to display host information on a port:

```
Port  IP Address
----- ------------------------------
3/45  10.1.1.5
```

To display the ACL name downloaded for a feature in addition to the port-specific information, perform this task in enable mode:

Task	Command
Display the ACL name downloaded for a feature and port-specific information.	**show port [dot1x| web-auth | eou | mac-auth-bypass]** *mod/port*

This example shows how to display the ACL name downloaded for a feature and port-specific information:

```
Console> (enable) show port dot1x 3/45
Port  Auth-State          BEnd-State Port-Control        Port-Status
----- ------------------- ---------- ------------------- -------------
 3/45 authenticated       idle       auto                authorized

Port  Port-Mode     Re-authentication   Shutdown-timeout   Control-Mode
                                                           admin   oper
----- ------------- -----------------   ----------------   ---------------
 3/45 SingleAuth    disabled            disabled           Both    Both

Port  Posture-Token Critical-Status Termination action Session-timeout
----- ------------- --------------- ------------------ ---------------
 3/45 -             no              NoReAuth           -

Port  Session-Timeout-Override Url-Redirect
----- ------------------------ ------------------------------
 3/45 disabled                 -

Port  Critical  Port-Name
----- --------  ---------
 3/45 disabled  -

Port  Downloaded ACL
----- ------------------------------------------
 3/45 #ACSACL#-IP-testacl-44c7197a
```

CHAPTER 16

Configuring NDE

This chapter describes how to configure NetFlow Data Export (NDE) on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Understanding How NDE Works

These sections describe how NDE works:

Overview of NDE and Integrated Layer 3 Switching Management

Catalyst 6500 series switches provide Layer 3 switching with Cisco Express Forwarding (CEF) for Supervisor Engine 2, Supervisor Engine 720, and Supervisor Engine 32. For Supervisor Engine 1 with the PFC, Layer 3 switching is provided with Multilayer Switching (MLS). You can use NDE to monitor all Layer 3-switched traffic through the Multilayer Switch Feature Card (MSFC). NDE complements the embedded Remote Monitoring (RMON) capabilities on the switch that allow you to see all port traffic.

Note NDE is not supported for the IP multicast or Internetwork Packet Exchange (IPX) traffic.

Note NDE version 7 and NDE version 8 are not supported for the MSFC.

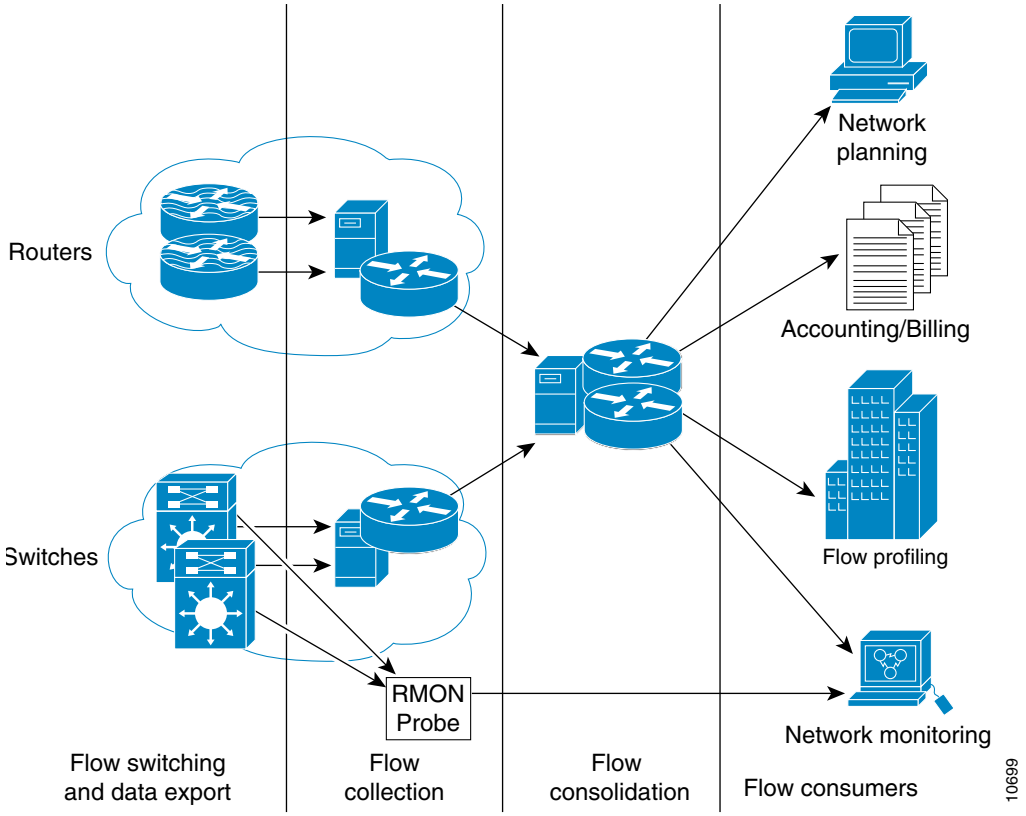
Note For information on configuring CEF for PFC2 and PFC3A, see Chapter 13, "Configuring CEF for PFC2 and PFC3A." For information on configuring MLS, see Chapter 14, "Configuring MLS."

Integrated Layer 3-switching management includes the products, management utilities, and partner applications that are designed to gather the flow statistics, export the statistics, collect and perform data reduction on the exported statistics, and forward them to the applications for traffic monitoring, planning, and accounting. The flow collectors, such as the Cisco SwitchProbe and NetFlow FlowCollector, gather and classify the flows. This flow information is then aggregated and fed to applications such as TrafficDirector, NetSys, or NetFlow Analyzer.

Traffic Statistics Data Collection

An external data collector gathers the flow entries from the statistics cache of one or more switches or Cisco routers. The switch or router transmits the data to the flow collector by grouping the flow entries for the expired flows from its statistics cache into a User Datagram Protocol (UDP) datagram, which consists of a header and a series of flow entries. See Figure 16-1.

Figure 16-1 Integrated Layer 3 Switching Management



Using NDE Filters

By default, *all* the expired flows are exported until you specify a filter. After specifying a filter, only the expired and purged flows matching the specified filter criteria are exported. The filter values are stored in NVRAM and are not cleared when NDE is disabled.

If the flow mask is in destination-ip mode and the NDE filter contains a filter on both source and destination, only the destination filter is effective. If the flow mask is in destination-ip mode (as shown in the following display), all the flows with destination address 9.1.2.15 are exported. The source filter for host 10.1.2.15 is not effective (it is ignored).

```
Console> (enable) set mls nde flow destination 9.1.2.15/32 source 10.1.2.15/32
Netflow data export: destination filter set to 9.1.2.15/32
Netflow data export: source filter set to 10.1.2.15/32
Console> (enable)
```

Using Bridged-Flow Statistics

Note The bridged-flow statistics are not supported on Supervisor Engine 720 or Supervisor Engine 32.

You can set the bridged-flow statistics reporting per VLAN. The bridged flows are exported through NDE when you enable the bridged-flow statistics.

Caution Use this feature carefully. As the NetFlow entries increase in the NetFlow table, the NDE performance may degrade. See the "NDE Configuration Guidelines" section on page 16-7 for information on configuring the bridged-flow statistics.

Note You can also enable NetFlow table entry creation on a per-VLAN basis. However, because the bridged-flow statistics and per-VLAN entry creation use the same mechanism for collecting the statistics, the VLAN entries may overlap. See the "Specifying NetFlow Table Entry Creation on a Per-Interface Basis" section on page 13-28.

NDE Versions

NDE on the PFC supports the following NDE versions to export the statistics that are captured on the PFC for the Layer 3-switched traffic:

- Supervisor Engine 1 and PFC
  - NDE version 5 with software release 7.5 and later releases
  - NDE version 7 with software release 6.1 and later releases
- Supervisor Engine 2 and PFC2
  - NDE version 5 with software release 7.5 and later releases
  - NDE version 7 with software release 6.1 and later releases
- Supervisor Engine 720 and PFC3A/PFC3B/PFC3BXL—NDE versions 5 and 7 (Supervisor Engine 720 was initially supported in software release 8.1[1]).

- Supervisor Engine 32 and PFC3B/PFC3BXL—NDE versions 5 and 7 (Supervisor Engine 32 was initially supported in software release 8.4[1]).

Depending on the current flow mask, some fields in the flow records might not have values. When the PFC exports the cached entries, the unsupported fields are filled with a zero (0).

The following tables list the supported NDE fields:

- Table 16-1—Version 5 header format
- Table 16-2—Version 5 flow record format
- Table 16-3—Version 7 header format
- Table 16-4—Version 7 flow record format

Table 16-1 NDE Version 5 Header Format

Bytes	Content	Description
0–1	version	NetFlow export format version number
2–3	count	Number of flows exported in this packet (1–30)
4–7	SysUptime	Current time in milliseconds since router booted
8–11	unix_secs	Current seconds since 0000 UTC 1970
12–15	unix_nsecs	Residual nanoseconds since 0000 UTC 1970
16–19	flow_sequence	Sequence counter of total flows seen
20–21	engine_type	Type of flow switching engine (VS_ENGINE_TYPE_CATALYST_SWITCH)
21–23	engine_id	0

Table 16-2 NDE Version 5 Flow Record Format

Bytes	Content	Description	Flow masks: X=Populated			
			Destination	Destination Source	Full	Full VLAN[1]
0–3	srcaddr	Source IP address	0	X	X	X
4–7	dstaddr	Destination IP address	X	X	X	X
8–11	nexthop	Next-hop router's IP address	X	X	X	X
12–13	input	Ingress interface SNMP ifIndex[2]	0	X	X	X
14–15	output	Egress interface SNMP ifIndex	X	X	X	X
16–19	dPkts	Packets in the flow	X	X	X	X
20–23	dOctets	Octets (bytes) in the flow	X	X	X	X
24–27	first	SysUptime at start of the flow (milliseconds)	X	X	X	X
28–31	last	SysUptime at the time the last packet of the flow was received (milliseconds)	X	X	X	X
32–33	srcport	Layer 4 source port number or equivalent	0	0	X	X

Table 16-2 NDE Version 5 Flow Record Format (continued)

Bytes	Content	Description	Flow masks: X=Populated			
			Destination	Destination Source	Full	Full VLAN[1]
34–35	dstport	Layer 4 destination port number or equivalent	0	0	X	X
36	pad1	Unused (zero) byte				
37	tcp_flags	Cumulative OR of TCP flags	0	0	0	0
38	prot	Layer 4 protocol (for example, 6=TCP, 17=UDP)	0	0	X	X
39	tos	IP type-of-service byte	X	X	X	X
40–41	src_as	Autonomous system number of the source, either origin or peer	0	0	0	0
42–43	dst_as	Autonomous system number of the destination, either origin or peer	0	0	0	0
44–45	src_mask	Source address prefix mask bits	0	0	0	0
46–47	dst_mask	Destination address prefix mask bits	0	0	0	0
48	pad2	Pad 2 is unused (zero) bytes				

1. This flow mask is not configurable from the CLI. It is only turned on if certain features, such as reflexive ACLs, are set up.
2. This feature is not supported on Supervisor Engine 1 or 1A.

Table 16-3 NDE Version 7 Header Format

Bytes	Content	Description
0–1	version	NetFlow export format version number
2–3	count	Number of flows exported in this packet (1–30)
4–7	SysUptime	Current time in milliseconds since router booted
8–11	unix_secs	Current seconds since 0000 UTC 1970
12–15	unix_nsecs	Residual nanoseconds since 0000 UTC 1970
16–19	flow_sequence	Sequence counter of total flows seen
20–24	reserved	Unused (zero) bytes

Table 16-4 NDE Version 7 Flow Record Format

Bytes	Content	Description	Flow masks: X=Populated			
			Destination	Destination Source	Full	Full VLAN[1]
0–3	srcaddr	Source IP address	0	X	X	X
4–7	dstaddr	Destination IP address	X	X	X	X
8–11	nexthop	Next-hop router's IP address	X	X	X	X
12–13	input	Ingress interface SNMP ifIndex[2]	0	X	X	X

Table 16-4 NDE Version 7 Flow Record Format (continued)

Bytes	Content	Description	Flow masks: X=Populated			
			Destination	Destination Source	Full	Full VLAN[1]
14–15	output	Egress interface SNMP ifIndex	X	X	X	X
16–19	dPkts	Packets in the flow	X	X	X	X
20–23	dOctets	Octets (bytes) in the flow	X	X	X	X
24–27	First	SysUptime at start of the flow (milliseconds)	X	X	X	X
28–31	Last	SysUptime at the time the last packet of the flow was received (milliseconds)	X	X	X	X
32–33	srcport	Layer 4 source port number or equivalent	0	0	X	X
34–35	dstport	Layer 4 destination port number or equivalent	0	0	X	X
36	flags	Flow mask in use	X	X	X	X
37	tcp_flags	Cumulative OR of TCP flags	0	0	0	0
38	prot	Layer 4 protocol (for example, 6=TCP, 17=UDP)	0	0	X	X
39	tos	IP type-of-service byte	X	X	X	X
40–41	src_as	Autonomous system number of the source, either origin or peer	0	0	0	0
42–43	dst_as	Autonomous system number of the destination, either origin or peer	0	0	0	0
44	src_mask	Source address prefix mask bits	0	0	0	0
45	dst_mask	Destination address prefix mask bits	0	0	0	0
46–47	pad2	Pad 2 uses two bytes				
48–51	MLS RP	IP address of MLS router	X[3]	X[2]	X[2]	X[2]

1. This flow mask is not configurable from the CLI. It is only turned on if certain features, such as reflexive ACLs, are set up.
2. This feature is not supported on Supervisor Engine 1 or 1A.
3. For switched entries.

Default NDE Configuration

Table 16-5 shows the default NDE configuration.

Table 16-5 Default NDE Configuration

Feature	Default Value
NDE	Disabled
NDE data collector address and UDP port	None specified
NDE filters	None configured

Configuring NDE on the Switch

These sections describe how to configure NDE:

NDE Configuration Guidelines

This section describes the configuration guidelines if the NetFlow table has too many entries:

- With software release 8.5(1) and later releases, the multiple flow mask feature is supported on Supervisor Engine 720. This feature results in some changes to the NDE functionality. For detailed information on using the multiple flow mask feature with NDE, see the "Flow Mask Modes—Software Release 8.5(1) and Later Releases" section on page 14-7.
- Reduce the MLS aging time. For PFC2, set the aging time high enough to keep the number of entries within the 32,000 flow range of the PFC2. For PFC3A, set the aging time high enough to keep the number of entries within the 64,000 flow range of the PFC3A.

  When using the bridged-flow statistics with a Supervisor Engine 2, set the aging time to 1 second. For information on how to change the MLS aging time, see the "Specifying the MLS Aging-Time Value" section on page 14-19 in Chapter 14, "Configuring MLS."

Note The bridged-flow statistics are not supported on Supervisor Engine 720 or Supervisor Engine 32.

- If there are protocols with fewer packets per flow running, reduce the MLS fast aging time. For information on how to change the MLS fast aging time, see the "Specifying IP MLS Long-Duration Aging Time, Fast Aging Time, and Packet Threshold Values" section on page 14-20 in Chapter 14, "Configuring MLS."
- Use the flow mask that is required to extract the kind of information that you want. A full flow mask gives more information but as the number of flows increase, the load on the Layer 3 aging also increases. Try to use a flow mask with the minimum granularity that is required to get the data that you need. With a full flow mask, you might need to decrease the MLS aging time because a full flow mask increases the number of flows per second. For information on setting the flow mask, see the "Setting the Minimum IP MLS Flow Mask" section on page 14-21 in Chapter 14, "Configuring MLS."
- Exclude the entries with fewer packets per flow. Some query protocols, like the Domain Name System (DNS), generate fewer packets per flow and can be excluded from the NetFlow table with the **set mls exclude protocol** command. You can specify up to four protocol filters, but the packets from the filtered protocols will go to the MSFC.
- Keep the specific flows from being added to the NetFlow table with the **set mls nde flow exclude** command.
- Enable the bridged-flow statistics on a VLAN to increase the number of flows in the NetFlow table with the bridged flows for VLANs appearing with the Layer 3 flows. As the NetFlow entries increase in the NetFlow table, the performance degrades.

  On the Supervisor Engine 1, if there is no space in the hardware NetFlow table to report the VLAN flows, the packets are sent to the MSFC for software forwarding and the NetFlow Full Errors register is incremented.

  On the Supervisor Engine 2, if a flow entry is not found in the NetFlow table, the packets are forwarded and the NetFlow Full Errors register is incremented resulting in a loss of statistics.

  To prevent the NetFlow table from overflowing, you can do the following:

  - Keep the flow mask at the least granular value. For example, if the protocol and Layer 4 port information is not required, set the flow mask to the destination-source or to the destination instead of to full flow.
  - Set the aging time to the least possible value (1 second), depending on the traffic profile.
  - Enable the bridged-flow statistics only on the VLANs on which the intraVLAN statistics are required. The interVLAN statistics are reported by default.
- You can enable NetFlow table entry creation on a per-VLAN basis. However, because the bridged-flow statistics and per-VLAN entry creation use the same mechanism for collecting the statistics, the VLAN entries may overlap. See the "Specifying NetFlow Table Entry Creation on a Per-Interface Basis" section on page 13-28.

Specifying an NDE Collector

Before enabling NDE for the first time, you must specify an NDE collector and UDP port to receive the exported statistics. The collector address and UDP port number are saved in NVRAM and are preserved if NDE is disabled and reenabled or if the switch is power cycled.

Note If you are using the NetFlow FlowCollector application for data collection, verify that the UDP port number that you specify is the same port number that is shown in the FlowCollector's nfconfig.file. This file is located at /opt/csconfc/config/nfconfig.file in the FlowCollector application.

With software release 8.3(1) and later releases, the dual destination feature allows NetFlow export data to be sent to two destinations simultaneously. With this enhancement, you can set up two unique collectors. The same NetFlow data is exported to both destinations. However, the count of the packets to the two collectors may differ depending on the time that the two destinations were created. The count of the packets sent to the individual collectors is maintained separately. The other NetFlow parameters for both the destinations are the same.

NDE cannot be enabled unless a collector is set up. You should set up both the primary and secondary destinations before enabling NDE.

The secondary destination IP address and port number cannot be identical to the primary destination IP address and port number.

To specify an NDE collector, perform this task in privileged mode:

Task	Command
Specify an NDE collector and UDP port for data export of hardware-switched packets.	**set mls nde** {*collector_ip* | *collector_name*} {*udp_port_number*}

This example shows how to specify an NDE collector when no other collectors have been configured:

```
Console> (enable) set mls nde 10.6.1.10 7772
Number of collectors configured is 1
Netflow export configured for port 7772 on host 10.6.1.10
Netflow export is not enabled. Please enable it now.
Console> (enable)
```

This example shows how to specify an NDE collector when one collector has already been configured:

```
Console> (enable) set mls nde 10.6.1.10 7775
Number of collectors configured is 2
Netflow export configured for port 7775 on host 10.6.1.10
Netflow export is not enabled. Please enable it now.
Console> (enable)
```

Clearing an NDE Collector

You can enter the **clear mls nde** command to clear both the primary and secondary collectors and disable NDE. To clear a specific collector destination, specify the collector IP address and port number.

To clear an NDE collector, perform this task in privileged mode:

Task	Command
Clear all NDE collectors or a specific NDE collector.	**clear mls nde** {*ip_address port*}

This example shows how to clear both the primary and secondary collectors:

```
Console> (enable) clear mls nde
Collector's IP address cleared.
Secondary Collector IP address cleared.
Console> (enable)
```

This example shows how to clear a specific collector destination:

```
Console> (enable) clear mls nde 10.6.1.10 9939
Cleared Collector IP 10.6.1.10 port 9939
Console> (enable)
```

Configuring NetFlow on the MSFC

Note If the MSFC is not present you can only collect (and export) bridged-flow statistics (if the bridged-flow statistics feature is enabled). You must enable NetFlow on the MSFC Layer 3 interfaces to support NDE for routed and Layer 3-switched traffic.

Refer to these publications for more information about configuring NetFlow on the MSFC:

- http://www.cisco.com/en/US/docs/ios/12_1/switch/configuration/guide/switch_c.html, "NetFlow," at this URL:

  http://www.cisco.com/en/US/docs/ios/12_1/switch/configuration/guide/xcdnfov.html
- *Cisco IOS Switching Services Command Reference*, Release 12.1, at this URL:

  http://www.cisco.com/en/US/docs/ios/12_1/switch/command/reference/switch_r.html

These sections describe how to configure NetFlow on the MSFC:

- Enabling NetFlow, page 16-11
- Configuring the MSFC NDE Source Interface, page 16-11
- Configuring the NDE Destination, page 16-11

Enabling NetFlow

To enable NetFlow, perform this task on each Layer 3 interface:

	Task	Command
Step 1	Select a VLAN interface to configure.	Router(config)# **interface vlan** *vlan_ID*
Step 2	Enable NetFlow.	Router(config-if)# **ip route-cache flow**

Configuring the MSFC NDE Source Interface

To configure the interface that is used as the source of the NDE packets containing the statistics from the MSFC, perform this task:

Task	Command
Configure the interface that is used as the source of the NDE packets containing the statistics from the MSFC: • Select an interface that is configured with an IP address. • Use a loopback interface.	Router(config)# **ip flow-export source** {**vlan** | **loopback**} *number*

This example shows how to configure a loopback interface as the NDE flow source:

```
Router(config)# ip flow-export source loopback 0
Router(config)#
```

Configuring the NDE Destination

To configure the NDE flow destination IP address and UDP port, perform this task:

Task	Command
Configure the NDE destination IP address and UDP port.	Router(config)# **ip flow-export destination** *ip_address udp_port_number*

This example shows how to configure the NDE flow destination IP address and UDP port:

```
Router(config)# ip flow-export destination 172.20.52.37 200
Router(config)#
```

Enabling NDE

To enable NDE, perform this task in privileged mode:

Task	Command
Enable NDE on the switch.	**set mls nde enable**

This example shows how to enable NDE on the switch:

```
Console> (enable) set mls nde enable
Netflow data export enabled.
Netflow data export to port 9996 on 172.20.15.1 (Stargate)
Console> (enable)
```

If you attempt to enable NDE without first specifying a collector, you see this display:

```
Console> (enable) set mls nde enable
Please set host name and UDP port number with 'set mls nde <collector_ip>
<udp_port_number>'.
Console> (enable)
```

Enabling and Disabling Bridged-Flow Statistics on VLANs

Note This feature is supported on the Supervisor Engine 1 or 1A/PFC, Supervisor Engine 2/PFC2 and no MSFC/MSFC2 is required. This feature is not supported on the Supervisor Engine 720 or Supervisor Engine 32.

Use the **set mls bridged-flow-statistics** command to enable or disable the bridged-flow statistics for the specified VLANs. You can enter one or multiple VLANs.

Note You can enable NetFlow table entry creation on a per-VLAN basis. However, because the bridged-flow statistics and per-VLAN entry creation use the same mechanism for collecting the statistics, the VLAN entries may overlap. See the "Specifying NetFlow Table Entry Creation on a Per-Interface Basis" section on page 13-28.

To enable or disable the bridged-flow statistics for a VLAN or for a range of VLANs, perform this task in privileged mode:

Task	Command
Enable or disable the bridged-flow statistics for a VLAN or for a range of VLANs.	**set mls bridged-flow-statistics** {**enable** | **disable**} {*vlanlist*}

This example shows how to enable the bridged-flow statistics on the specified VLANs:

```
Console> (enable) set mls bridged-flow-statistics enable 1,20-21
Netflow statistics is enabled for bridged packets on vlan(s) 1,20-21.
Console> show mls nde
Netflow Data Export version: 7
Netflow Data Export enabled
Netflow Data Export configured for port 9991 on host 21.0.0.1
Total packets exported = 0
Bridged flow statistics is enabled on vlan(s) 1,20-21.
Console>
```

Specifying a Destination Host Filter

To specify a destination host filter, perform this task in privileged mode:

Task	Command
Specify a destination host filter for an NDE flow.	**set mls nde flow destination** [*ip_addr_spec*]

This example shows how to specify a destination host filter so that only the expired flows to host 171.69.194.140 are exported:

```
Console> (enable) set mls nde flow destination 171.69.194.140
Netflow Data Export successfully set
Destination filter is 171.69.194.140/255.255.255.255
Filter type: include
Console> (enable)
```

Specifying a Destination and Source Subnet Filter

To specify a destination and source subnet filter, perform this task in privileged mode:

Task	Command
Specify a destination and source subnet filter for an NDE flow.	**set mls nde flow destination** [*ip_addr_spec*] **source** [*ip_addr_spec*]

This example shows how to specify a destination and source subnet filter so that only the expired flows to subnet 171.69.194.0 from subnet 171.69.173.0 are exported (assuming that the flow mask is set to source-destination-ip):

```
Console> (enable) set mls nde flow destination 171.69.194.140/24 source 171.69.173.5/24
Netflow Data Export successfully set
Source filter is 171.69.173.0/24
Destination filter is 171.69.194.0/24
Filter type: include
Console> (enable)
```

Specifying a Destination TCP/UDP Port Filter

To specify a destination TCP/UDP port filter, perform this task in privileged mode:

Task	Command
Specify a destination TCP/UDP port filter for an NDE flow.	**set mls nde flow dst-prt** [*port_number*]

This example shows how to specify a destination TCP/UDP port filter so that only the expired flows to destination port 23 are exported (the flow mask is set to full):

```
Console> (enable) set mls nde flow dst-port 23
Netflow Data Export successfully set
Destination port filter is 23
Filter type: include
Console> (enable)
```

Specifying a Source Host and Destination TCP/UDP Port Filter

To specify a source host and destination TCP/UDP port filter, perform this task in privileged mode:

Task	Command
Specify a source host and destination TCP/UDP port filter for an NDE flow.	**set mls nde flow source** [*ip_addr_spec*] **dst-prt** [*port_number*]

This example shows how to specify a source host and destination TCP/UDP port filter so that only the expired flows from host 171.69.194.140 to destination port 23 are exported (the flow mask is set to full):

```
Console> (enable) set mls nde flow source 171.69.194.140 dst-port 23
Netflow Data Export successfully set
Source filter is 171.69.194.140/255.255.255.255
Destination port filter is 23
Filter type: include
Console> (enable)
```

Specifying a Protocol Filter

To specify a protocol filter, perform this task in privileged mode:

Task	Command
Specify a protocol filter for an NDE flow.	**set mls nde flow protocol** *protocol*

This example shows how to specify a protocol filter so that only the expired flows from protocol 17 are exported:

```
Console> (enable) set mls nde flow protocol 17
Netflow Data Export filter successfully set.
Protocol filter is 17
Filter type: include
Console> (enable)
```

Specifying Protocols for Statistics Collection

You can enter the **set mls statistics protocol** *protocol port* command to specify up to 64 different protocols for which to collect statistics to be exported using NDE. The *protocol* argument can be **ip**, **ipinip**, **icmp**, **igmp**, **tcp**, and **udp**, or a decimal number for the other protocol families. The *port* argument specifies the protocol port.

To specify the protocols for statistics collection, perform this task in privileged mode:

Task	Command
Specify the protocols for statistics collection.	**set mls statistics protocol** *protocol port*

This example shows how to specify a protocol for statistics collection:

```
Console> (enable) set mls statistics protocol 17 1934
Protocol 17 port 1934 is added to protocol statistics list.
Console> (enable)
```

Removing Protocols for Statistics Collection

You can enter the **clear mls statistics protocol** {*protocol port* | **all**} command to specify up to 64 different protocols for which to collect statistics to be exported using NDE. The *protocol* argument can be **tcp**, **udp**, **icmp**, or a decimal number for the other protocol families. The *port* argument specifies the protocol port. Use the **all** keyword to remove all the protocols for statistics collection.

To remove the protocols for statistics collection, perform this task in privileged mode:

Task	Command
Remove the protocols for statistics collection.	**clear mls statistics protocol** {*protocol port*	**all**}

This example shows how to remove a protocol for statistics collection:

```
Console> (enable) clear mls statistics protocol 17 1934
Protocol 17 port 1934 cleared from protocol statistics list.
Console> (enable)
```

Clearing the NDE Flow Filter

To clear the NDE flow filter and reset the filter to the default (all flows exported), perform this task in privileged mode:

Task	Command
Clear the NDE flow filter.	**clear mls nde flow**

This example shows how to clear the NDE flow filter so that all the flows are exported:

```
Console> (enable) clear mls nde flow
Netflow data export filter cleared.
Console> (enable)
```

Disabling NDE

Note With Supervisor Engine 1 and a PFC, if NDE is enabled and you disable MLS, you lose the statistics for existing cache entries because the statistics are not exported.

To disable NDE on the switch, perform this task in privileged mode:

Task	Command
Disable NDE on the switch.	**set mls nde disable**

This example shows how to disable NDE on the switch:

```
Console> (enable) set mls nde disable
Netflow data export disabled.
Console> (enable)
```

Removing the NDE IP Address

To remove the NDE IP address from the MSFC, perform this task in global configuration mode:

Task	Command
Remove the NDE IP address from the MSFC.	Router(config)# **no mls nde-address** [*ip_addr*]

This example shows how to remove the NDE IP address from the MSFC:

```
Router(config)# no mls nde-address 170.170.2.1
Router(config)#
```

Displaying the NDE Configuration

To display the NDE configuration on the switch, perform this task in privileged mode:

Task	Command
Display the NDE configuration on the switch.	**show mls nde**

This example shows how to display the NDE configuration on the switch:

```
Console> (enable) show mls nde
Netflow Data Export enabled
Netflow Data Export configured for port 7772 on host 10.6.1.10
Secondary Data Export configured for port 7775 on host 10.6.1.10
Source filter is 171.69.194.140/255.255.255.0
Destination port filter is 23
Total packets exported = 26784
Console> (enable)
```

This example shows how to display the NDE configuration when the bridged-flow statistics are enabled on the switch:

```
Console> (enable) show mls nde
Netflow Data Export version:7
Netflow Data Export enabled
Netflow Data Export configured for port 7772 on host 10.6.1.10
Secondary Data Export configured for port 7775 on host 10.6.1.10
Total packets exported = 0
Bridged flow statistics is enabled on vlan(s) 1,20-21.
```

CHAPTER 17

Configuring GVRP

This chapter describes how to configure the Generic Attribute Registration Protocol (GARP) VLAN Registration Protocol (GVRP) on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Note GVRP requires supervisor engine software release 5.2 or later releases. With a Supervisor Engine 720, the minimum required software release is 8.3(1). With a Supervisor Engine 32, the minimum required software release is 8.4(1).

Understanding How GVRP Works

GVRP is a GARP application that provides IEEE 802.1Q-compliant VLAN pruning and dynamic VLAN creation on 802.1Q trunk ports.

With GVRP, the switch can exchange the VLAN configuration information with the other GVRP switches, prune the unnecessary broadcast and unknown unicast traffic, and dynamically create and manage the VLANs on the switches that are connected through the 802.1Q trunk ports.

Note GARP and GVRP are industry-standard protocols that are described in IEEE 802.1p.

Default GVRP Configuration

Table 17-1 shows the default GVRP configuration.

Table 17-1 GVRP Default Configuration

Feature	Default Value
GVRP global enable state	Disabled
GVRP per-trunk enable state	Disabled on all ports
GVRP dynamic creation of VLANs	Disabled
GVRP registration mode	**normal**, with VLAN 1 set to **fixed**, for all ports
GVRP applicant state	**normal** (ports do not declare VLANs when in the STP[1] blocking state)
GARP timers	• Join time: 200 ms • Leave time: 600 ms • Leaveall time: 10,000 ms

1. STP = Spanning Tree Protocol

GVRP Configuration Guidelines

This section describes the guidelines for configuring GVRP:

- You can configure the per-port GVRP state only on the 802.1Q-capable ports.
- You must enable GVRP on both ends of an 802.1Q trunk link.
- The GVRP registration mode for VLAN 1 is always **fixed** and is not configurable.
- When VTP pruning is enabled, it runs on all the GVRP-disabled 802.1Q trunk ports.

Configuring GVRP on the Switch

These sections describe how to configure GVRP:

Enabling GVRP Globally

You must enable GVRP globally before any GVRP processing occurs on the switch. Enabling GVRP globally enables GVRP to perform the VLAN pruning on the 802.1Q trunk links. The pruning occurs only on the GVRP-enabled trunks. For information on setting the per-trunk port GVRP enable state, see the "Enabling GVRP on Individual 802.1Q Trunk Ports" section on page 17-3.

To enable dynamic VLAN creation, you must explicitly enable dynamic VLAN creation globally on the switch. For information on enabling dynamic VLAN creation, see the "Enabling GVRP Dynamic VLAN Creation" section on page 17-4.

To enable GVRP globally on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Enable GVRP on the switch.	**set gvrp enable**
Step 2	Verify the configuration.	**show gvrp configuration**

This example shows how to enable GVRP and verify the configuration:

```
Console> (enable) set gvrp enable
GVRP enabled
Console> (enable) show gvrp configuration
Global GVRP Configuration:
GVRP Feature is currently enabled on the switch.
GVRP dynamic VLAN creation is disabled.
GVRP Timers(milliseconds)
Join = 200
Leave = 600
LeaveAll = 10000

Port based GVRP Configuration:
Port                                                    GVRP Status Registration
------------------------------------------------------- ----------- ------------
2/1-2,3/1-8,7/1-24,8/1-24                               Enabled     Normal

GVRP Participants running on 3/7-8.
Console>
```

Enabling GVRP on Individual 802.1Q Trunk Ports

Note You can change the per-trunk GVRP configuration regardless of whether GVRP is enabled globally. However, GVRP does not function on any ports until you enable it globally. For information on configuring GVRP globally on the switch, see the "Enabling GVRP Globally" section on page 17-3.

There are two per-port GVRP states:

- The static GVRP state that is configured in the command-line interface (CLI) and stored in NVRAM
- The actual GVRP state of the ports (active GVRP participants)

You can configure the static GVRP port-state on any of the 802.1Q-capable switch ports, regardless of the global GVRP enable state or whether the port is an 802.1Q trunk. However, in order for the port to become an active GVRP participant, you must enable GVRP globally and the port must be an 802.1Q trunk port, either through CLI configuration or Dynamic Trunking Protocol (DTP) negotiation.

To enable GVRP on the individual 802.1Q-capable ports, perform this task in privileged mode:

	Task	Command
Step 1	Enable GVRP on an individual 802.1Q-capable port.	**set port gvrp** *mod/port* **enable**
Step 2	Verify the configuration.	**show gvrp configuration**

This example shows how to enable GVRP on 802.1Q-capable port 1/1:

```
Console> (enable) set port gvrp 1/1 enable
GVRP enabled on 1/1.
Console> (enable)
```

Enabling GVRP Dynamic VLAN Creation

You can enable GVRP dynamic VLAN creation only if these conditions are met:

- The switch is in VTP transparent mode
- All the trunk ports on the switch are 802.1Q trunks (the trunk connection to an MSFC is exempt from this restriction)
- GVRP is enabled on all the trunk ports

If you enable dynamic VLAN creation, these configuration restrictions are imposed:

- You cannot change the switch to VTP server or client mode
- You cannot disable GVRP on a trunk port running GVRP

If any port on the switch becomes an Inter-Switch Link (ISL) trunk (either by CLI configuration or negotiated using DTP) while dynamic VLAN creation is enabled, dynamic VLAN creation is disabled automatically until the conditions for enabling dynamic VLAN creation are restored.

Note The VLANs can only be created dynamically on 802.1Q trunks in the **normal** registration mode.

Note Dynamic VLAN creation supports all VLAN types.

To enable GVRP dynamic VLAN creation on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Enable dynamic VLAN creation on the switch.	**set gvrp dynamic-vlan-creation enable**
Step 2	Verify the configuration.	**show gvrp configuration**

This example shows how to enable dynamic VLAN creation on the switch:

```
Console> (enable) set gvrp dynamic-vlan-creation enable
Dynamic VLAN creation enabled.
Console> (enable)
```

Configuring GVRP Registration

These sections describe how to configure GVRP registration modes on switch ports:

Configuring GVRP Normal Registration

Configuring an 802.1Q trunk port in **normal** registration mode allows dynamic creation (if dynamic VLAN creation is enabled), registration, and deregistration of the VLANs on the trunk port. Normal mode is the default.

To configure GVRP normal registration on an 802.1Q trunk port, perform this task in privileged mode:

	Task	Command
Step 1	Configure normal registration on an 802.1Q trunk port.	**set gvrp registration normal** *mod/port*
Step 2	Verify the configuration.	**show gvrp configuration**

This example shows how to configure normal registration on an 802.1Q trunk port:

```
Console> (enable) set gvrp registration normal 1/1
Registrar Administrative Control set to normal on port 1/1.
Console> (enable)
```

Configuring GVRP Fixed Registration

Configuring an 802.1Q trunk port in **fixed** registration mode allows manual creation and registration of VLANs, prevents VLAN deregistration, and registers all the VLANs that are known on other ports on the trunk port.

To configure GVRP fixed registration on an 802.1Q trunk port, perform this task in privileged mode:

	Task	Command
Step 1	Configure fixed registration on an 802.1Q trunk port.	**set gvrp registration fixed** *mod/port*
Step 2	Verify the configuration.	**show gvrp configuration**

This example shows how to configure fixed registration on an 802.1Q trunk port:

```
Console> (enable) set gvrp registration fixed 1/1
Registrar Administrative Control set to fixed on port 1/1.
Console> (enable)
```

Configuring GVRP Forbidden Registration

Configuring an 802.1Q trunk port in **forbidden** registration mode deregisters all the VLANs (except VLAN 1) and prevents any further VLAN creation or registration on the trunk port.

To configure GVRP forbidden registration on an 802.1Q trunk port, perform this task in privileged mode:

	Task	Command
Step 1	Configure forbidden registration on an 802.1Q trunk port.	**set gvrp registration forbidden** *mod/port*
Step 2	Verify the configuration.	**show gvrp configuration**

This example shows how to configure forbidden registration on an 802.1Q trunk port:

```
Console> (enable) set gvrp registration forbidden 1/1
Registrar Administrative Control set to forbidden on port 1/1.
Console> (enable)
```

Configuring GVRP VLAN Declarations from Blocking Ports

To prevent the undesirable Spanning Tree Protocol (STP) topology reconfiguration on a port that is connected to a device that does not support Per-VLAN STP+ (PVST+), configure the GVRP active applicant state on the port. The ports in the GVRP active applicant state send GVRP VLAN declarations when they are in the STP blocking state, which prevents the STP bridge protocol data units (BPDUs) from being pruned from the other port.

Note Configuring fixed registration on the other device's port also prevents undesirable STP topology reconfiguration.

To configure an 802.1Q trunk port to send VLAN declarations when it is in the blocking state, perform this task in privileged mode:

Task	Command
Configure an 802.1Q trunk port to send VLAN declarations when it is in the blocking state.	**set gvrp applicant state** {**normal** | **active**} *mod/port*

This example shows how to configure a group of 802.1Q trunk ports to send VLAN declarations when it is in the blocking state:

```
Console> (enable) set gvrp applicant state active 4/2-3,4/9-10,4/12-24
Applicant was set to active on port(s) 4/2-3,4/9-10,4/12-24.
Console> (enable)
```

Use the **normal** keyword to return to the default state (active mode disabled).

Setting the GARP Timers

Note The **set gvrp timer** and **show gvrp timer** commands are aliases for the **set garp timer** and **show garp timer** commands. The aliases may be used if desired.

Note Modifying the GARP timer values affects the behavior of *all* the GARP applications running on the switch, not just GVRP. (For example, GMRP uses the same timers.)

You can modify the default GARP timer values on the switch.

When setting the timer values, the value for **leave** must be greater than three times the **join** value (**leave >= join** * 3). The value for **leaveall** must be greater than the value for **leave** (**leaveall > leave**).

If you attempt to set a timer value that does not adhere to these rules, an error is returned. For example, if you set the **leave** timer to 600 ms and you attempt to configure the **join** timer to 350 ms, an error is returned. Set the **leave** timer to at least 1050 ms and then set the **join** timer to 350 ms.

Caution Set the same GARP timer values on all the Layer 2-connected devices. If the GARP timers are set differently on the Layer 2-connected devices, the GARP applications (for example, GMRP and GVRP) do not operate successfully.

To set the GARP timer values, perform this task in privileged mode:

	Task	Command
Step 1	Set the GARP timer values.	**set garp timer** {**join** | **leave** | **leaveall**} *timer_value*
Step 2	Verify the configuration.	**show garp timer**

This example shows how to set the GARP timers and verify the configuration:

```
Console> (enable) set garp timer leaveall 10000
GMRP/GARP leaveAll timer value is set to 10000 milliseconds.
Console> (enable) set garp timer leave 600
GMRP/GARP leave timer value is set to 600 milliseconds.
Console> (enable) set garp timer join 200
GMRP/GARP join timer value is set to 200 milliseconds.
Console> (enable) show garp timer
Timer     Timer Value (milliseconds)
--------  --------------------------
Join      200
Leave     600
LeaveAll  10000
Console> (enable)
```

Displaying GVRP Statistics

To display the GVRP statistics on the switch, perform this task:

Task	Command
Display the GVRP statistics.	**show gvrp statistics** [*mod/port*]

This example shows how to display the GVRP statistics for port 1/1:

```
Console> (enable) show gvrp statistics 1/1
Join Empty Received:     0
Join In Received:        0
Empty Received:          0
LeaveIn Received:        0
Leave Empty Received:    0
Leave All Received:      40
Join Empty Transmitted:  156
Join In Transmitted:     0
Empty Transmitted:       0
Leave In Transmitted:    0
Leave Empty Transmitted: 0
Leave All Transmitted:   41
VTP Message Received:    0
Console> (enable)
```

Clearing GVRP Statistics

To clear all the GVRP statistics on the switch, perform this task in privileged mode:

Task	Command
Clear the GVRP statistics.	**clear gvrp statistics** {*mod/port* | **all**}

This example shows how to clear all the GVRP statistics on the switch:

```
Console> (enable) clear gvrp statistics all
GVRP Statistics cleared for all ports.
Console> (enable)
```

Disabling GVRP on Individual 802.1Q Trunk Ports

To disable GVRP on the individual 802.1Q trunk ports, perform this task in privileged mode:

	Task	Command
Step 1	Disable GVRP on an individual 802.1Q trunk port.	**set port gvrp disable** *mod/port*
Step 2	Verify the configuration.	**show gvrp configuration**

This example shows how to disable GVRP on 802.1Q trunk port 1/1:

```
Console> (enable) set gvrp disable 1/1
GVRP disabled on 1/1.
Console> (enable)
```

Disabling GVRP Globally

To disable GVRP globally on the switch, perform this task in privileged mode:

Task	Command
Disable GVRP on the switch.	**set gvrp disable**

This example shows how to disable GVRP globally on the switch:

```
Console> (enable) set gvrp disable
GVRP disabled
Console> (enable)
```

CHAPTER 18

Configuring MVRP

This chapter describes how to configure the IEEE 802.1ak Multiple VLAN Registration Protocol (MVRP) on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Note MVRP requires Catalyst 6500 series switch software release 8.7(1) or later.

Understanding How MVRP Works

MVRP is an MRP application that provides IEEE 802.1ak-compliant VLAN pruning and dynamic VLAN creation on trunk ports.

With MVRP, the switch can exchange VLAN configuration information with other MVRP switches, prune the unnecessary broadcast and unknown unicast traffic, and dynamically create and manage the VLANs on the switches that are connected through trunk ports.

Note MRP and MVRP are industry-standard protocols that are described in IEEE 802.1ak.

Note MVRP replaces GVRP with faster and smaller transmissions, and extends support to larger networks and 4k (4094) VLANs.

Default MVRP Configuration

Table 18-1 shows the default MVRP configuration.

Table 18-1 MVRP Default Configuration

Feature	Default Value
MVRP global enable state	Disabled
MVRP per-trunk enable state	Disabled on all ports
MVRP dynamic creation of VLANs	Disabled
MVRP registration mode	**normal**, with VLAN 1 set to **fixed**, for all ports
MVRP applicant state	**normal** (ports do not declare VLANs when in STP[1] blocking state)
MVRP timers	• Join timer: 20 cs[2] • Leave timer: 60 cs • Leaveall timer: 1000 cs
MVRP periodic timer	Disabled

1. STP = Spanning Tree Protocol
2. cs = Centiseconds

MVRP Configuration Guidelines

When configuring MVRP, follow these guidelines:

- GVRP must be disabled globally and on a per-port basis.
- You can configure MVRP only on the trunk ports.
- You must enable MVRP on both ends of a trunk link.
- The switch should not have any PVLAN configuration.
- The MVRP registration mode for VLAN 1 is always *fixed* and is not configurable.
- When VTP pruning is enabled, it interoperates with MVRP on all the MVRP-enabled trunk ports.

Scalability Data for MVRP

- On a Catalyst 6500 series switch with Supervisor Engine 720, that runs software release 8.7(3), the switch supports the following within an optimal operational CPU utilization:
  - MVRP pruning for 4094 VLANs on 6 trunks/channels.
  - MVRP pruning for 2500 VLANs on 11 trunks/channels

  The default timer values are Join- 20 Centi-seconds, Leave-150 Centi-seconds, and Leave All-6000 Centi-seconds.

Caution Increase in number of VLANs and decreased timer values from the set default, results in high CPU utilization

Configuring MVRP on the Switch

These sections describe how to configure MVRP:

Enabling MVRP Globally

You must enable MVRP globally before any MVRP processing occurs on the switch. Enabling MVRP globally enables MVRP to perform the VLAN pruning on the trunk links. The pruning occurs only on the MVRP-enabled trunks. For information on setting the per-trunk port MVRP enable state, refer to the "Enabling MVRP on Individual Trunk Ports" section on page 18-4.

Note MVRP cannot be enabled globally and on individual trunk ports if PVLANs exist.

To enable dynamic VLAN creation, you must enable dynamic VLAN creation globally on the switch. For information on enabling dynamic VLAN creation, refer to the "Enabling MVRP Dynamic VLAN Creation" section on page 18-5.

To enable MVRP globally on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Enable MVRP on the switch.	**set mvrp enable**
Step 2	Verify the configuration.	**show mvrp configuration**

This example shows how to enable MVRP and verify the configuration:

```
Console> (enable) set mvrp enable
MVRP enabled
```

```
Console> (enable) show mvrp configuration

Global MVRP Configuration:
MVRP Feature is currently enabled on the switch.
MVRP dynamic VLAN creation is disabled.

Port based MVRP Configuration:
MVRP-Status Registration Applicant Port(s)
----------- ------------ --------- ------------
Enabled     Normal       Normal    3/1-10,3/14,3/24
Disabled    Normal       Normal    2/2-3,3/11-13,3/15-23,3/25-48
Disabled    Fixed        Normal    2/1

Trunk based MVRP Configuration:
MVRP-Status Registration Applicant Trunk(s)
----------- ------------ --------- ------------
Enabled     Normal       Normal    3
Disabled    Normal       Normal    2

MVRP Timers(centiseconds):
-------------------------
JoinTimer  LvTimer  LvAllTimer  Port(s)
---------  -------  ----------  --------
30         600      1000        3/1
200        600      1000        3/2
200        600      1000        3/3
200        600      1000        3/4
200        600      1000        3/14
200        600      1000        3/24
Console>
```

Enabling MVRP on Individual Trunk Ports

Note You can change the per-trunk MVRP configuration even if MVRP is enabled globally. However, MVRP does not function on any ports until you enable it globally. For information on configuring MVRP globally on the switch, see the "Enabling MVRP Globally" section on page 18-3.

You can enable MVRP on any of the individual trunk ports, regardless of the global MVRP enable state. However, in order for the port to transmit and receive MVRP data, and become an active MVRP participant, you must enable MVRP globally.

To enable MVRP on the individual trunk ports, perform this task in privileged mode:

	Task	Command
Step 1	Enable MVRP on an individual trunk port.	**set port mvrp** *mod/port* **enable**
Step 2	Verify the configuration.	**show mvrp configuration**

This example shows how to enable MVRP on individual trunk port 3/1:

```
Console> (enable) set port mvrp 3/1 enable
MVRP enabled on 3/1.
Console> (enable)
```

Enabling MVRP Dynamic VLAN Creation

You can enable MVRP dynamic VLAN creation only if these conditions are met:

- The switch is in VTP transparent or off mode.
- The switch does not have any PVLAN configuration.

If you enable dynamic VLAN creation, these configuration restrictions are imposed:

- You cannot change the switch to VTP server or client mode.
- When PVLAN information (primary and secondary) from VTPv3 is received, MVRP creates it as normal VLAN.

Note VLANs can only be created dynamically on 802.1ak trunks in the **normal** registration mode.

To enable MVRP dynamic VLAN creation on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Enable dynamic VLAN creation on the switch.	**set mvrp dynamic-vlan-creation enable**
Step 2	Verify the configuration.	**show mvrp configuration**

This example shows how to enable dynamic VLAN creation on the switch:

```
Console> (enable) set mvrp dynamic-vlan-creation enable
MVRP Dynamic VLAN creation is enabled.
Console> (enable)
```

Configuring MVRP Registration

These sections describe how to configure MVRP registration modes on switch ports:

- Configuring MVRP Normal Registration, page 18-5
- Configuring MVRP Fixed Registration, page 18-6
- Configuring MVRP Forbidden Registration, page 18-6

Configuring MVRP Normal Registration

Configuring an 802.1ak trunk port in **normal** registration mode responds to all MVRP requests and messages, while retaining all registrations and deregistrations on the trunk port. Normal mode is the default.

To configure MVRP normal registration on a trunk port, perform this task in privileged mode:

	Task	Command
Step 1	Configure normal registration on an 802.1ak trunk port.	**set port mvrp** *mod/port* **registration normal**
Step 2	Verify the configuration.	**show mvrp configuration**

This example shows how to configure normal registration on an 802.1ak trunk port:

```
Console> (enable) set port mvrp 3/1 registration normal
Registrar Administrative Control set to normal on port(s) 3/1.
Console> (enable)
```

Configuring MVRP Fixed Registration

Configuring an 802.1ak trunk port in **fixed** registration mode ignores further MVRP requests and messages while retaining all existing registrations on the trunk port.

Note The registration is fixed for all configured VLANs on the port.

To configure MVRP fixed registration on an 802.1ak trunk port, perform this task in privileged mode:

	Task	Command
Step 1	Configure fixed registration on an 802.1ak trunk port.	**set port mvrp** *mod/port* **registration fixed**
Step 2	Verify the configuration.	**show mvrp configuration**

This example shows how to configure fixed registration on an 802.1ak trunk port:

```
Console> (enable) set port mvrp 3/1 registration fixed
Registrar Administrative Control set to fixed on port(s) 3/1.
Console> (enable)
```

Configuring MVRP Forbidden Registration

Configuring an 802.1ak trunk port in **forbidden** registration mode deregisters all the VLANs (except VLAN 1) on the trunk port.

To configure MVRP forbidden registration on an 802.1ak trunk port, perform this task in privileged mode:

	Task	Command
Step 1	Configure forbidden registration on an 802.1ak trunk port.	**set port mvrp** *mod/port* **registration forbidden**
Step 2	Verify the configuration.	**show mvrp configuration**

This example shows how to configure forbidden registration on an 802.1ak trunk port:

```
Console> (enable) set port mvrp 3/1 registration forbidden
Registrar Administrative Control set to forbidden on port(s) 3/1.
Console> (enable)
```

Configuring MVRP on Ports with STP Blocking State

To prevent Spanning Tree Protocol (STP) topology reconfiguration on a port that is connected to a device that does not support Per-VLAN STP+ (PVST+), configure the MVRP active applicant state on the port. The ports in the MVRP active applicant state send MVRP VLAN declarations when they are in the STP blocking state, which prevents the STP bridge protocol data units (BPDUs) from being pruned from the other port.

Note Configuring fixed registration on the other device's port also prevents STP topology reconfiguration.

To configure an 802.1ak trunk port to send VLAN declarations when it is in the blocking state, perform this task in privileged mode:

Task	Command
Configure a trunk port to send VLAN declarations when it is in the blocking state.	**set port mvrp** *mod/port* **applicant** {**normal** | **active**}

This example shows how to configure a group of trunk ports to send VLAN declarations when it is in the blocking state:

```
Console> (enable) set port mvrp 4/2-3,4/9-10,4/12-24 applicant active
MVRP Applicant was set to active on port(s) 4/2-3,4/9-10,4/12-24.
Console> (enable)
```

Use the **normal** keyword to return to the default state (active mode disabled).

Configuring the MVRP Timers

An MVRP-enabled port uses the timers listed in Table 18-2 to transmit, receive, and respond to requests.

Table 18-2 MVRP Timers

Timer	Description
join	The join timer defines the interval between transmit opportunities. The value can range from 20 to 10000000, in centiseconds. The default is 20.
leave	The leave timer defines the waiting time before transiting to an empty state. The value can range from 60 to 10000000, in centiseconds. The default is 60.
leaveall	The leavall timer defines the frequency in which the leaveall message is generated. The value can range from 1000 to 10000000, in centiseconds. The default is 1000.

The leave time should be at least twice the join time to allow reregistration after a leave or leaveall message, even if a message is lost. To minimize the volume of rejoining traffic generated following a leaveall message, the leaveall time should be larger than the leave time.

Caution Set the same MVRP timer values on all the Layer 2-connected devices.

To set the MVRP timer values, perform this task in privileged mode:

	Task	Command
Step 1	Set the MVRP timer values.	**set port mvrp** *mod/port* **timer** {**join** | **leave** | **leaveall**} *timer_value*
Step 2	Verify the configuration.	**show mvrp configuration**

This example shows how to set the MVRP timers and verify the configuration:

```
Console> (enable) set port mvrp 4/2-3,4/9-10 timer leaveall 10000
MVRP leaveAll timer value is set to 10000 centiseconds for port(s) 4/2-3,4/9-10.
Console> (enable) set port mvrp 4/2-3,4/9-10 timer leave 600
MVRP leave timer value is set to 600 centiseconds for port(s) 4/2-3,4/9-10.
Console> (enable) set port mvrp 4/2-3,4/9-10 timer join 200
MVRP join timer value is set to 200 centiseconds for port(s) 4/2-3,4/9-10.
Console> (enable)
```

Enabling the Periodic Timer

The periodic timer defines the frequency during which the periodic events are generated. The value is preset to 1 second. The periodic timer value cannot be modified but can either be enabled or disabled. The defaut is disabled.

To enable the MVRP periodic timer, perform this task in privileged mode:

	Task	Command
Step 1	Enable the MVRP periodic timer.	**set port mvrp** *mod/port* **periodictimer** {**enable** | **disable**}

This example shows how to enable the MVRP periodic timer on a range of ports:

```
Console> (enable) set port mvrp 4/2-3,4/9-10,4/12-24 periodictimer enable
MVRP periodic timer is enabled on port(s) 4/2-3,4/9-10,4/12-24.
console>
```

Displaying MVRP Configuration Summary

To display the summary of MVRP configuration on the switch, perform this task in enabled mode:

Task	Command
Display summary of MVRP configuration.	**show mvrp configuration**

This example shows how to display the summary of MVRP configuration on the switch:

```
Console> (enable) show mvrp configuration
Global MVRP Configuration:
MVRP Feature is currently disabled on the switch.
MVRP dynamic VLAN creation is disabled.

Port based MVRP Configuration:
```

```
MVRP-Status Registration Applicant Port(s)
----------- ------------ --------- ------------
Enabled     Normal       Normal    3/1-10,3/14,3/24
Disabled    Normal       Normal    2/2-3,3/11-13,3/15-23,3/25-48
Disabled    Fixed        Normal    2/1

MVRP Participants running on no ports.

MVRP Timers(centiseconds):
-------------------------
JoinTimer  LvTimer  LvAllTimer  Port(s)
---------  -------  ----------  --------
30         600      1000        2/1
200        600      1000        3/1
200        600      1000        3/2
200        600      1000        3/3
200        600      1000        3/4
200        600      1000        3/5
Console> (enable)
```

Displaying MVRP Statistics

To display the MVRP statistics on the switch, perform this task:

Task	Command
Display the MVRP statistics.	**show mvrp statistics** [*mod/port*]

This example shows how to display the MVRP statistics for port 3/1:

```
Console> (enable) show mvrp statistics 3/1
Valid packets Received:    186
Invalid Packets Received:  0
New Received:              0
Join In Received:          1167
In Received:               0
Join Empty Received:       22387
Empty Received:            31
Leave Received:            210
Leave All Received:        63
Packets Transmitted:       176
New Transmitted:           0
Join In Transmitted:       311
In Transmitted:            0
Join Empty Transmitted:    873
Empty Transmitted:         11065
Leave Transmitted:         167
Leave All Transmitted:     4
Packets Received:          249
Packets Dropped:           0
Y76> (enable)
Console> (enable)
```

Displaying MVRP State Machines

To display the MVRP state machines on the switch, perform this task:

Task	Command
Display the MVRP states on the port.	**show mvrp machines** [*vlan/mod/port*]

This example shows how to display the MVRP state machines for port 3/14:

```
Console> (enable) show mvrp machines 3/14
MAD machine state:
Mod/Port VLAN   Applicant: state & mgmt       Registrar: state & mgmt
-------- ----   ------------------------------ ----------------------
  3/14   0001   Aa:  Anxious      Normal,      IN  Registration_fixed
  3/14   0142   Aa:  Anxious      Normal,      MT  Normal_registration
  3/14   1002   Vo:  Very_anxious Normal,      MT  Normal_registration
  3/14   1003   Vo:  Very_anxious Normal,      MT  Normal_registration
  3/14   1004   Vo:  Very_anxious Normal,      MT  Normal_registration
  3/14   1005   Vo:  Very_anxious Normal,      MT  Normal_registration
  3/14   1006   Vo:  Very_anxious Normal,      MT  Normal_registration
  3/14   1007   Vo:  Very_anxious Normal,      MT  Normal_registration
  3/14   1008   Vo:  Very_anxious Normal,      MT  Normal_registration
  3/14   1009   Vo:  Very_anxious Normal,      MT  Normal_registration
  3/14   1010   Vo:  Very_anxious Normal,      MT  Normal_registration
  3/14   1011   Vo:  Very_anxious Normal,      MT  Normal_registration
  3/14   1012   Vo:  Very_anxious Normal,      MT  Normal_registration
  3/14   1016   Vo:  Very_anxious Normal,      MT  Normal_registration
Console> (enable)
```

Displaying MVRP Trunks

To display the MVRP trunk information for the port, perform this task:

Task	Command
Display MVRP trunk information and VLANs pruned.	**show mvrp trunk** [*mod/port*]

This example shows how to display the MVRP trunk information for port 3/14:

```
Console> (enable) show mvrp trunk 3/14
Port            MVRP Pruned Vlans
-----           ----------------------------------------
3/14             2-4094
Console> (enable)
```

Disabling MVRP on Individual Trunk Ports

To disable MVRP on the individual trunk ports, perform this task in privileged mode:

	Task	Command
Step 1	Disable MVRP on an individual trunk port.	**set port mvrp** *mod/port* **disable**
Step 2	Verify the configuration.	**show mvrp configuration**

This example shows how to disable MVRP on a port 3/14:

```
Console> (enable) set port mvrp 3/14 disable
MVRP is disabled on port(s) 3/14.
Console> (enable)
```

Disabling MVRP Globally

To disable MVRP globally on the switch, perform this task in privileged mode:

Task	Command
Disable MVRP on the switch.	**set mvrp disable**

This example shows how to disable MVRP globally on the switch:

```
Console> (enable) set mvrp disable
MVRP is disabled on the switch.
Console> (enable)
```

Clearing MVRP Configuration

To clear all MVRP configuration on the switch, perform this task in privileged mode

:

Task	Command
Clear MVRP configuration.	**clear mvrp configuration** {**all** | *mod/port*}

This example shows how to clear all MVRP configuration on the switch:

```
Console> (enable) clear mvrp configuration all
Warning:MVRP configuration will be cleared.
Do you want to continue (y/n) [y]? y
MVRP configuration is cleared for all ports on the switch.
Console> (enable)
```

Clearing MVRP Counters

To clear all MVRP counters on the switch, perform this task in privileged mode:

Task	Command
Clear all MVRP counters.	**clear mvrp counters**

This example shows how to clear all MVRP counters on the switch:

```
Console> (enable) clear mvrp counters
Warning:MVRP counters will be cleared.
Do you want to continue (y/n) [y]? y
MVRP counters cleared for all ports on the swtich.
Console> (enable)
```

Clearing MVRP Statistics

To clear all the MVRP statistics on the switch, perform this task in privileged mode:

Task	Command
Clear the MVRP statistics.	**clear mvrp statistics** {**all** | *mod/port*}

This example shows how to clear all MVRP statistics on the switch:

```
Console> (enable) clear mvrp statistics all
Warning:MVRP statistics will be cleared.
Do you want to continue (y/n) [y]? y
MVRP Statistics for all ports are cleared.
Console> (enable)
```

CHAPTER 19

Configuring Dynamic Port VLAN Membership with VMPS

This chapter describes how to configure dynamic port VLAN membership using the VLAN Management Policy Server (VMPS) on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Understanding How VMPS Works

With VMPS, you can assign the switch ports to the VLANs dynamically, based on the source Media Access Control (MAC) address of the device that is connected to the port. When you move a host from a port on one switch in the network to a port on another switch in the network, the switch assigns the new port to the proper VLAN for that host dynamically.

When you enable VMPS, a MAC address-to-VLAN mapping database downloads from a Trivial File Transfer Protocol (TFTP) server and VMPS begins to accept the client requests. If you reset or power cycle the switch, the VMPS database downloads from the TFTP server automatically and VMPS is reenabled.

VMPS opens a User Datagram Protocol (UDP) socket to communicate and listen to the client requests. When the VMPS server receives a valid request from a client, it searches its database for a MAC address-to-VLAN mapping.

If the assigned VLAN is restricted to a group of ports, VMPS verifies the requesting port against this group. If the VLAN is allowed on the port, the VLAN name is returned to the client. If the VLAN is not allowed on the port and VMPS is not in secure mode, the host receives an "access denied" response. If VMPS is in secure mode, the port is shut down.

If a VLAN in the database does not match the current VLAN on the port and active hosts are on the port, VMPS sends an access denied or a port shutdown response that is based on the VMPS secure mode.

You can configure a fallback VLAN name. If you connect a device with a MAC address that is not in the database, VMPS sends the fallback VLAN name to the client. If you do not configure a fallback VLAN and the MAC address does not exist in the database, VMPS sends an access denied response. If VMPS is in secure mode, it sends a port shutdown response.

You can also make an explicit entry in the configuration table to deny access to specific MAC addresses for security reasons by specifying a **--NONE--** keyword for the VLAN name. In this case, VMPS sends an access denied or port shutdown response.

A dynamic port can belong to only one *native* VLAN in software releases prior to release 6.2(1)—with software release 6.2(1), a port can belong to a native VLAN and an auxiliary VLAN. See the "Dynamic Port VLAN Membership with Auxiliary VLANs" section on page 19-14 for complete details.

When the link comes up, a dynamic port is isolated from its static VLAN. The source MAC address from the first packet of a new host on the dynamic port is sent to VMPS, which attempts to match the MAC address to a VLAN in the VMPS database. If there is a match, VMPS provides the VLAN number to assign to the port. If there is no match, VMPS either denies the request or shuts down the port (depending on the VMPS secure mode setting).

Multiple hosts (MAC addresses) can be active on a dynamic port if they are all in the same VLAN. If the link goes down on a dynamic port, the port returns to an isolated state. Any hosts that come online through the port are checked again with VMPS before the port is assigned to a VLAN.

Default VMPS and Dynamic Port Configuration

Table 19-1 shows the default VMPS and dynamic port configuration.

Table 19-1 Default VMPS and Dynamic Port Configuration

Feature	Default Configuration
VMPS server	
VMPS enable state	Disabled
VMPS management domain	Null
VMPS TFTP server	None
VMPS database configuration filename	*vmps-config-database.1*
VMPS fallback VLAN	Null
VMPS secure mode	Open
VMPS no domain requests	Allow

Table 19-1 Default VMPS and Dynamic Port Configuration (continued)

Feature	Default Configuration
VMPS Client	
VMPS domain server	None
VMPS reconfirm interval	60 minutes
VMPS server retry count	3
Dynamic ports	No dynamic ports configured

Dynamic Port VLAN Membership and VMPS Configuration Guidelines

This section describes the guidelines for dynamic port VLAN membership:

- You must configure VMPS before you configure the ports as dynamic.
- When you configure a port as dynamic, spanning-tree PortFast is enabled automatically for that port. Automatic enabling of spanning-tree PortFast prevents the applications on the host from timing out and entering loops that are caused by incorrect configurations. You can disable spanning-tree PortFast mode on a dynamic port.
- If you reconfigure a port from a static port to a dynamic port on the same VLAN, the port connects immediately to that VLAN. However, VMPS checks the legality of the specific host on the dynamic port after a certain period.
- The static secure ports cannot become dynamic ports. You must turn off security on the static secure port before it can become dynamic.
- The static ports that are trunking cannot become dynamic ports. You must turn off trunking on the trunk port before changing it from static to dynamic.

Note The VLAN Trunking Protocol (VTP) management domain and the management VLAN of the VMPS clients and the VMPS server must be the same. For more information, see Chapter 10, "Configuring VTP," and Chapter 11, "Configuring VLANs."

Configuring VMPS and Dynamic Port VLAN Membership on the Switch

These sections describe how to configure VMPS and define the dynamic ports on the clients:

Creating the VMPS Database

To use VMPS, you must create a VMPS database and store it on a TFTP server. The VMPS parser is line based. Start each entry in the file on a new line. Ranges are not allowed for the port numbers.

Note For an example ASCII text VMPS database configuration file, see the "VMPS Database Configuration File Example" section on page 19-10.

Follow these guidelines for creating the VMPS database file:

- Begin the configuration file with the word "VMPS," to prevent other types of configuration files from incorrectly being read by the VMPS server.
- Define the VMPS domain—The VMPS domain should correspond to the VTP domain name that is configured on the switch.
- Define the security mode—VMPS can operate in open or secure mode.
- (Optional) Define a fallback VLAN—The fallback VLAN is assigned if the MAC addresses of the connected host is not defined in the database.
- Define the MAC address-to-VLAN name mappings—Enter the MAC address of each host and the VLAN to which each should belong. Use the **--NONE--** keyword as the VLAN name to deny the specified host network connectivity. A port is identified by the IP address of the switch and the module/port number of the port in the form *mod/port*.
- Define port groups—A port group is a logical group of ports. You can apply VMPS policies to individual ports or to port groups. The keyword **all-ports** specifies all the ports in the specified switch.
- Define VLAN groups—A VLAN group defines a logical group of VLANs. These logical groups define the VLAN port policies.
- Define VLAN port policies—The VLAN port policies define the ports that are associated with a restricted VLAN. You can configure a restricted VLAN by defining the set of dynamic ports on which it can exist.

To create a VMPS database, perform this task:

	Task	Command
Step 1	Determine the MAC addresses of the hosts that you want to be assigned to VLANs dynamically.	**show cam**
Step 2	Create an ASCII text file on your workstation or PC that contains the MAC address-to-VLAN mappings.	–
Step 3	Move the ASCII text file to a TFTP server so that it can be downloaded to the switch.	–

Configuring VMPS

When you enable VMPS, the switch downloads the VMPS database from the TFTP or rcp server and begins accepting VMPS requests.

To configure VMPS, perform this task in privileged mode:

	Task	Command
Step 1	Specify the download method.	**set vmps downloadmethod rcp** | **tftp** [*username*]
Step 2	Configure the IP address of the TFTP or rcp server on which the ASCII text VMPS database configuration file resides.	**set vmps downloadserver** *ip_addr* [*filename*]
Step 3	Enable VMPS.	**set vmps state enable**
Step 4	Verify the VMPS configuration.	**show vmps**

This example shows how to enable VMPS on the switch:

```
Console> (enable) set vmps state enable
Vlan Membership Policy Server enable is in progress.
Console> (enable)
```

To disable VMPS, perform this task in privileged mode:

	Task	Command
Step 1	Disable VMPS.	**set vmps state disable**
Step 2	Verify that VMPS is disabled.	**show vmps**

This example shows how to disable VMPS on the switch:

```
Console> (enable) set vmps state disable
All the VMPS configuration information will be lost and the resources released on disable.
Do you want to continue (y/n[n]): y
Vlan Membership Policy Server disabled.
Console> (enable)
```

Configuring Dynamic Ports on VMPS Clients

To configure dynamic ports on VMPS client switches, perform this task in privileged mode:

	Task	Command
Step 1	Specify the IP address of the VMPS server (the switch with VMPS enabled).	**set vmps server** *ip_addr* [**primary**]
Step 2	Verify the VMPS server specification.	**show vmps server**
Step 3	Configure the dynamic port VLAN membership assignments to a port.	**set port membership** *mod/port* **dynamic**
Step 4	Verify the dynamic port assignments.	**show port** [*mod*[*/port*]]

This example shows how to specify the VMPS server, verify the VMPS server specification, assign the dynamic ports, and verify the configuration:

```
Console> (enable) show vmps server
VMPS domain server VMPS Status
--------------------------------------
192.0.0.6
192.0.0.1       primary
192.0.0.9
Console> (enable) set port membership 3/1-3 dynamic
Ports 3/1-3 vlan assignment set to dynamic.
Spantree port fast start option enabled for ports 3/1-3.
Console> (enable) set port membership 1/2 dynamic
Trunking port 1/2 vlan assignment cannot be set to dynamic.
Console> (enable) set port membership 2/1 dynamic
ATM LANE port 2/1 vlan assignment can not be set to dynamic.
Console> show port
Port   Name     Status   Vlan    Level     Duplex       Speed       Type
1/1             connect  dyn-3   normal    full         100         100 BASE-TX
1/2             connect  trunk   normal    half         100         100 BASE-TX
2/1             connect  trunk   normal    full         155         OC3 MMF ATM
3/1             connect  dyn-5   normal    half         10          10 BASE-T
3/2             connect  dyn-5   normal    half         10          10 BASE-T
3/3             connect  dyn-5   normal    half         10          10 BASE-T
Console> (enable)
```

Note The **show port** command displays *dyn-* under the Vlan column of the display when it has not yet been assigned a VLAN for a port.

Administering and Monitoring VMPS

To show information about the MAC address-to-VLAN mappings, perform one of these tasks in privileged mode:

Task	Command
Show the VLAN to which a MAC address is mapped in the database.	**show vmps mac** [*mac_address*]
Show the MAC addresses that are mapped to a VLAN in the database.	**show vmps vlan** [*vlan_name*]
Show the ports belonging to a restricted VLAN.	**show vmps vlanports** [*vlan_name*]

To show the VMPS statistics, perform this task in privileged mode:

Task	Command
Show the VMPS statistics.	**show vmps statistics**

To clear the VMPS statistics, perform this task in privileged mode:

Task	Command
Clear the VMPS statistics.	**clear vmps statistics**

To clear a VMPS server entry, perform this task in privileged mode:

Task	Command
Clear a VMPS server entry.	**clear vmps server** *ip_addr*

To reconfirm the dynamic port VLAN membership assignments, perform this task in privileged mode:

	Task	Command
Step 1	Reconfirm the dynamic port VLAN membership assignments.	**reconfirm vmps**
Step 2	Verify the dynamic VLAN reconfirmation status.	**show dvlan statistics**

This example shows how to reconfirm dynamic port VLAN membership assignments:

```
Console> (enable) reconfirm vmps
reconfirm process started
Use 'show dvlan statistics' to see reconfirm status
Console> (enable)
```

To download the VMPS database manually (to download a changed database configuration file or retry after a failed download attempt), perform this task in privileged mode:

	Task	Command
Step 1	Download the VMPS database from the TFTP server, or specify a different VMPS database configuration file.	**download vmps**
Step 2	Verify the VMPS database configuration file.	**show vmps**

Configuring Static VLAN Port Membership

To return a port to static VLAN port membership, perform this task in privileged mode:

	Task	Command
Step 1	Configure the static port VLAN membership assignments to a port.	**set port membership** *mod*/*port* **static**
Step 2	Verify the static port assignments.	**show port** [*mod*[/*port*]]

This example shows how to return a port to static VLAN port membership:

```
Console> (enable) set port membership 3/1 static
Port 3/1 vlan assignment set to static.
Console> (enable)
```

Backing up the VMPS Configuration File

Use the VMPS configuration file backup feature to prevent delays after a power shutdown when the VMPS clients and servers are coming back online. After a power shutdown, the VMPS requests that are sent by the clients are queued by the TFTP server until the VMPS server downloads the VMPS configuration file from the VMPS server. To ensure that the client access is not delayed during a system reboot, you can configure the switch to back up the VMPS configuration file locally and use this file until it is has downloaded the current VMPS configuration file from the remote TFTP server.

To configure the switch to back up the VMPS configuration file, perform this task in privileged mode:

	Task	Command
Step 1	Manually back up the VMPS configuration file. **Note** If you do not specify the filename, the system will save the VMPS configuration file as **vmps-backup-config-database.1**.	**set vmps config-file device:** [**filename**]
Step 2	Enable an automatic backup of the VMPS configuration file.	**set vmps config-file auto-save enable | disable**
Step 3	Verify the configuration.	**show vmps**

This example shows how to manually back up the VMPS configuration file:

```
Console> (enable) set vmps config-file disk0:
Vlan Membership Policy Server back-up file name is set to disk0:vmps-backup-conf
ig-database.1.
Console> (enable)
```

This example shows how to configure the system to automatically back up the VMPS configuration file:

```
Console> (enable) set vmps config-file auto-save enable
Auto save to store Vlan Membership Policy Server configuration file is enabled.
Console> (enable)
```

This example shows how to verify the configuration:

```
Console> show vmps
VMPS Server Status:
-------------------
Management Domain        (null)
State                    disabled
Operational Status       inactive
TFTP Server              default
TFTP File                vmps-config-database.1
Fallback VLAN            (null)
Secure Mode              open
VMPS No Domain Req       allow
VMPS Backup file name    disk0:vmps-backup-config-database.1
VMPS Auto-Save state     enabled
```

```
VMPS Client Status:
---------------------
VMPS VQP Version:     1
Reconfirm Interval:   60 min
Server Retry Count:   3
VMPS domain server:

No dynamic ports configured.
Console>
```

Troubleshooting VMPS and Dynamic Port VLAN Membership

These sections describe how to troubleshoot VMPS and dynamic port VLAN membership:

- Troubleshooting VMPS, page 19-9
- Troubleshooting Dynamic Port VLAN Membership, page 19-10

Troubleshooting VMPS

Table 19-2 shows the VMPS error messages that you might see when you enter the **set vmps state enable** or the **download vmps** command.

Table 19-2 VMPS Error Messages

VMPS Error Message	Recommended Action
`TFTP server IP address is not configured.`	Specify the TFTP server address using the **set vmps tftpserver** *ip_addr* [*filename*] command.
`Unable to contact the TFTP server 172.16.254.222.`	Enter a static route (using the **set ip route** command) to the TFTP server.
`File "vmps_configuration.db" not found on the TFTP server 172.16.254.222.`	Check the filename of the VMPS database configuration file on the TFTP server. Make sure that the permissions are set correctly.
`Enable failed due to insufficient resources.`	The switch does not have sufficient resources to run the database. You can fix this problem by increasing the dynamic random-access memory (DRAM).

After VMPS successfully downloads the VMPS database configuration file, it parses the file and builds a database. When the parsing is complete, VMPS outputs statistics about the total number of lines that are parsed and the number of parsing errors.

To obtain more information on the VMPS parsing errors, set the syslog level for VMPS to 3 using the **set logging level vmps 3** command.

Troubleshooting Dynamic Port VLAN Membership

A dynamic port might shut down under these circumstances:

- VMPS is in secure mode and it is illegal for the host to connect to the port. The port shuts down to prevent the host from connecting to the network.
- More than 50 active hosts reside on a dynamic port.

To reenable a shut-down dynamic port, enter the **set port enable** *mod/port* command.

Dynamic Port VLAN Membership with VMPS Configuration Examples

These sections provide examples of how to configure VMPS and dynamic ports:

- VMPS Database Configuration File Example, page 19-10
- Dynamic Port VLAN Membership Configuration Example, page 19-12

VMPS Database Configuration File Example

This example shows a sample VMPS database configuration file. A VMPS database configuration file is an ASCII text file that is stored on a TFTP server accessible to the switch configured as the VMPS server. A summary of the configuration example follows:

- The security mode is open.
- The default is used for the fallback VLAN.
- MAC address-to-VLAN name mappings—The MAC address of each host and the VLAN to which each host belongs is defined.
- Port groups are defined.
- VLAN groups are defined.
- VLAN port policies are defined for the ports that are associated with restricted VLANs.

```
!VMPS File Format, version 1.1
! Always begin the configuration file with
! the word "VMPS"
!
!vmps domain <domain-name>
! The VMPS domain must be defined.
!vmps mode {open | secure}
! The default mode is open.
!vmps fallback <vlan-name>
!vmps no-domain-req { allow | deny }
!
! The default value is allow.
vmps domain WBU
vmps mode open
vmps fallback default
vmps no-domain-req deny
!
!
!MAC Addresses
!
```

```
vmps-mac-addrs
!
! address <addr> vlan-name <vlan_name>
!
address 0012.2233.4455 vlan-name hardware
address 0000.6509.a080 vlan-name hardware
address aabb.ccdd.eeff vlan-name Green
address 1223.5678.9abc vlan-name ExecStaff
address fedc.ba98.7654 vlan-name --NONE--
address fedc.ba23.1245 vlan-name Purple
!
!Port Groups
!
!vmps-port-group <group-name>
! device <device-id> { port <port-name> | all-ports }
!
vmps-port-group WiringCloset1
 device 198.92.30.32 port 3/2
 device 172.20.26.141 port 2/8
vmps-port-group "Executive Row"
 device 198.4.254.222 port 1/2
 device 198.4.254.222 port 1/3
 device 198.4.254.223 all-ports
!
!
!VLAN groups
!
!vmps-vlan-group <group-name>
! vlan-name <vlan-name>
!
vmps-vlan-group Engineering
vlan-name hardware
vlan-name software
!
!
!VLAN port Policies
!
!vmps-port-policies {vlan-name <vlan_name> | vlan-group <group-name> }
! { port-group <group-name> | device <device-id> port <port-name> }
!
vmps-port-policies vlan-group Engineering
 port-group WiringCloset1
vmps-port-policies vlan-name Green
 device 198.92.30.32 port 4/8
vmps-port-policies vlan-name Purple
 device 198.4.254.22 port 1/2
 port-group "Executive Row"
```

Dynamic Port VLAN Membership Configuration Example

Figure 19-1 shows a network with a VMPS server switch and VMPS client switches with dynamic ports. In this example, these assumptions apply:

- The VMPS server and the VMPS client are separate switches.
- Switch 1 is the primary VMPS server.
- Switch 3 and Switch 10 are secondary VMPS servers.
- End stations are connected to these clients:
  - Switch 2
  - Switch 9
- The database configuration file is called Bldg-G.db and is stored on a TFTP server with IP address 172.20.22.7.

To configure VMPS and dynamic ports, perform these steps:

Step 1 Configure Switch 1 as the primary VMPS server.

a. Configure the IP address of the TFTP server on which the ASCII file resides:

```
Console> (enable) set vmps tftpserver 172.20.22.7 Bldg-G.db
```

b. Enable VMPS:

```
Console> (enable) set vmps state enable
```

After entering these commands, the file Bldg-G.db is downloaded to Switch 1. Switch 1 becomes the VMPS server.

Step 2 Configure the VMPS server addresses on each VMPS client.

a. Configure the primary VMPS server IP address:

```
Console> (enable) set vmps server 172.20.26.150 primary
```

b. Configure the secondary VMPS server IP addresses:

```
Console> (enable) set vmps server 172.20.26.152

Console> (enable) set vmps server 172.20.26.159
```

c. Verify the VMPS server addresses:

```
Console> (enable) show vmps server
```

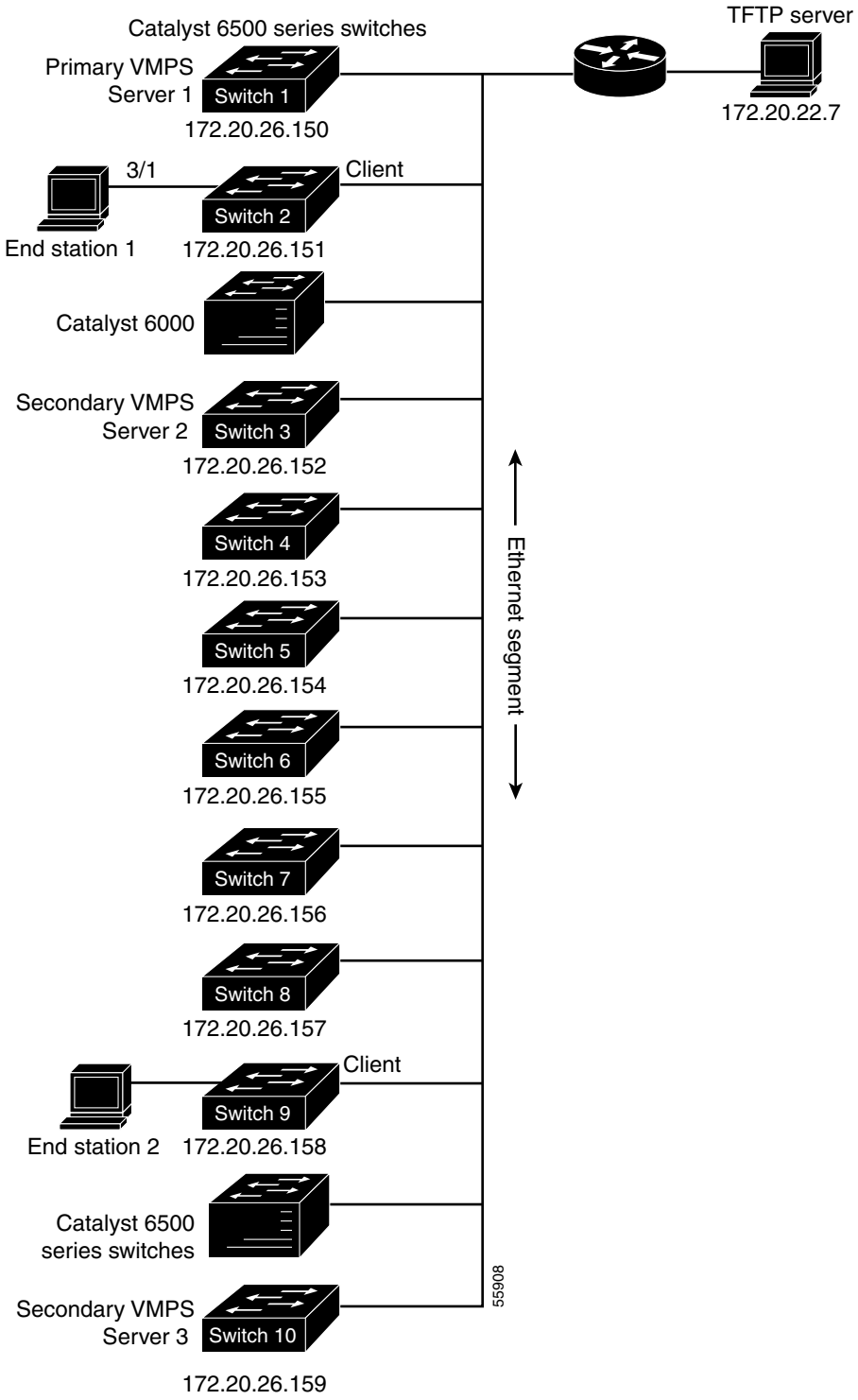
Step 3 Configure port 3/1 on Switch 2 as dynamic.

```
Console> (enable) set port membership 3/1 dynamic
```

Step 4 Connect End Station 2 on port 3/1. When End Station 2 sends a packet, Switch 2 sends a query to the primary VMPS server, Switch 1. Switch 1 responds with the VLAN to assign to port 3/1. Because spanning-tree PortFast mode is enabled by default on dynamic ports, port 3/1 connects immediately and enters forwarding mode.

Step 5 Repeat Steps 2 and 3 to configure the VMPS server addresses and assign dynamic ports on each VMPS client switch.

Figure 19-1 Dynamic Port VLAN Membership Configuration



Dynamic Port VLAN Membership with Auxiliary VLANs

Note This feature requires software release 6.2(1) or later releases.

This section describes how to configure a dynamic port to belong to two VLANs—a native VLAN and an auxiliary VLAN. This section uses the following terminology:

- Auxiliary VLAN—Separate VLAN for IP phones
- Native VLAN—Traditional VLAN for data
- Auxiliary VLAN ID—VLAN ID of an auxiliary VLAN
- Native VLAN ID—VLAN ID of a native VLAN

Prior to software release 6.2(1), the dynamic ports could only belong to one VLAN. You could not enable the dynamic port VLAN feature on the ports that carried a native VLAN and an auxiliary VLAN.

With software release 6.2(1) and later releases, the dynamic ports can belong to two VLANs. The switch port that is configured for connecting an IP phone can have separate VLANs that are configured for carrying the following:

- Voice traffic to and from the IP phone (auxiliary VLAN)
- Data traffic to and from the PC that is connected to the switch through the *access port* of the IP phone (native VLAN)

These sections include configuration guidelines and examples:

- Dynamic Port VLAN Membership with Auxiliary VLANs Guidelines, page 19-14
- Configuring Dynamic Port VLAN Membership with Auxiliary VLANs, page 19-15

Note For detailed information on the auxiliary VLANs and Cisco voice-over-IP networks, see Chapter 55, "Configuring a VoIP Network."

Dynamic Port VLAN Membership with Auxiliary VLANs Guidelines

This section describes the guidelines and restrictions for configuring dynamic port VLAN membership for the auxiliary VLANs:

- Configuration of the native VLAN ID is dynamic for the PC that is connected to the access port of the IP phone. Configuration of the auxiliary VLAN ID is not dynamic; you need to configure it manually. As the auxiliary VLAN ID is manually configured, the VMPS server is queried for packets coming from the PC, not for the packets coming from the IP phone.
- All the packets except the Cisco Discovery Protocol (CDP) packets from the IP phone are tagged with the auxiliary VLAN ID. All the packets that are tagged with the auxiliary VLAN ID are considered to be the packets from the phone, and all the other packets are considered to be the packets from the PC.

- When configuring the auxiliary VLAN ID with 802.1p or untagged frames, you need to configure the VMPS server with the IP phone's MAC address (see the "Dynamic Port VLAN Membership with VMPS Configuration Examples" section on page 19-10 for information on configuring VMPS).
- For the dynamic ports, the auxiliary VLAN ID cannot be the same as the native VLAN ID that is assigned by VMPS for the dynamic port.
- See the "Dynamic Port VLAN Membership and VMPS Configuration Guidelines" section on page 19-3 prior to configuring any port.

Configuring Dynamic Port VLAN Membership with Auxiliary VLANs

To configure dynamic port VLAN membership with auxiliary VLANs, perform this task in privileged mode:

Task	Command
Configure dynamic port VLAN membership with the auxiliary VLANs.	**set port auxiliaryvlan** *mod*[*/port*] {*vlan* | **untagged** | **dot1p** | **none**} [**cdpverify** {**enable** | **disable**}]

This example shows how to add voice ports to the auxiliary VLANs and specify an encapsulation type:

```
Console> (enable) set port auxiliaryvlan 5/9 222
Auxiliaryvlan 222 configuration successful.
AuxiliaryVlan AuxVlanStatus Mod/Ports
------------ ------------ ------------------------
222          active       5/9
Console> (enable)

Console> (enable) set port auxiliaryvlan 5/9 dot1p
Port 5/9 allows the connected device send and receive packets with 802.1p priority.
Console> (enable)
```

This example shows how to specify port 5/9 as a dynamic port:

```
Console> (enable) set port membership 5/9 dynamic
Warning: Auxiliary Vlan set to dot1p|untagged on dynamic port. VMPS will be queried for IP
phones.
Port 5/9 vlan assignment set to dynamic.
Spantree port fast start option enabled for ports 5/9.
Console> (enable)
```

This example shows that the auxiliary VLAN ID specified cannot be the same as the native VLAN ID:

```
Console> (enable) set port auxiliaryvlan 5/10 223
Auxiliary vlan cannot be set to 223 as PVID=223.
Console> (enable)
```

CHAPTER 20

Checking Status and Connectivity

This chapter describes how to check the status and connectivity on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Checking the Module Status

Catalyst 6500 series switches are multimodule systems. You can see what modules are installed and the MAC address ranges and version numbers for each module using the **show module** [*mod*] command. Specify a particular module number to see detailed information on that module.

To check the module status, perform this task in normal mode:

Task	Command
Check the module status.	**show module** [*mod*]

This example shows how to check the module status. The output shows that there is one supervisor engine and four additional modules that are installed in the chassis.

```
Console> (enable) show module
Mod Slot Ports Module-Type                Model                Status
--- ---- ----- ------------------------- -------------------- --------
1   1    2     1000BaseX Supervisor      WS-X6K-SUP1-2GE      ok
2   2    24    100BaseFX MM Ethernet     WS-X6224-100FX-MT    ok
3   3    8     1000BaseX Ethernet        WS-X6408-GBIC        ok
4   4    48    10/100BaseTX (Telco)      WS-X6248-TEL         ok
5   5    48    10/100BaseTX (RJ-45)      WS-X6248-RJ-45       ok

Mod Module-Name         Serial-Num
--- ------------------- -----------
1                       SAD03040546
2                       SAD03110020
3                       SAD03070194
4                       SAD03140787
5                       SAD03181291

Mod MAC-Address(es)                         Hw     Fw          Sw
--- --------------------------------------- ------ ----------- ------------------
1   00-50-f0-a8-26-b2 to 00-50-f0-a8-26-b3 1.4    5.1(1)      5.2(1)CSX
    00-50-f0-a8-26-b0 to 00-50-f0-a8-26-b1
    00-50-3e-8d-64-00 to 00-50-3e-8d-67-ff
2   00-50-54-6c-e9-a8 to 00-50-54-6c-e9-bf 1.3    4.2(0.24)V 5.2(1)CSX
3   00-50-54-6c-93-6c to 00-50-54-6c-93-73 1.4    4.2(0.24)V 5.2(1)CSX
4   00-50-54-bf-59-64 to 00-50-54-bf-59-93 0.103  4.2(0.24)V 5.2(1)CSX
5   00-50-f0-ac-30-54 to 00-50-f0-ac-30-83 1.0    4.2(0.24)V 5.2(1)CSX

Mod Sub-Type                Sub-Model           Sub-Serial  Sub-Hw
--- ----------------------- ------------------- ----------- ------
1   L2 Switching Engine I   WS-F6020            SAD03040312 1.0
Console> (enable)
```

This example shows how to check the module status on a specific module:

```
Console> (enable) show module 4
Mod Slot Ports Module-Type                Model                Status
--- ---- ----- ------------------------- -------------------- --------
4   4    48    10/100BaseTX (Telco)      WS-X6248-TEL         ok

Mod Module-Name         Serial-Num
--- ------------------- -----------
4                       SAD03140787

Mod MAC-Address(es)                         Hw     Fw          Sw
--- --------------------------------------- ------ ----------- ------------------
4   00-50-54-bf-59-64 to 00-50-54-bf-59-93 0.103  4.2(0.24)V 5.2(1)CSX
Console> (enable)
```

Checking the Port Status

You can see summary or detailed information on the switch ports using the **show port** [*mod*[**/***port*]] command. To see summary information on all of the ports on the switch, enter the **show port** command with no arguments. Specify a particular module number to see information on the ports on that module only. Enter both the module number and the port number to see detailed information about the specified port.

To apply configuration commands to a particular port, you must specify the appropriate logical module. For more information, see the "Checking the Module Status" section on page 20-2.

To check the port status, perform this task in normal mode:

Task	Command
Check the port status.	**show port** [*mod*[**/***port*]]

This example shows how to see information on the ports on a specific module only:

```
Console> (enable) show port 1
Port  Name               Status     Vlan       Duplex Speed Type
----- ------------------ ---------- ---------- ------ ----- ------------
 1/1                     connected  1            full  1000 1000BaseSX
 1/2                     notconnect 1            full  1000 1000BaseSX

Port  Security Secure-Src-Addr   Last-Src-Addr     Shutdown Trap     IfIndex
----- -------- ----------------- ----------------- -------- -------- -------
 1/1  disabled                                     No       disabled 3
 1/2  disabled                                     No       disabled 4

Port     Broadcast-Limit Broadcast-Drop
-------- --------------- --------------------
 1/1                   -                    0
 1/2                   -                    0

Port   Send FlowControl    Receive FlowControl   RxPause    TxPause
       admin    oper       admin    oper
-----  -------- --------   -------- ---------     ---------- ----------
 1/1   desired  off        off      off           0          0
 1/2   desired  off        off      off           0          0

Port  Status     Channel   Admin Ch    Neighbor                          Neighbor
                 Mode      Group Id    Device                               Port
----- ---------- --------- ----- ----- ---------------------------------- -----
 1/1  connected  auto      65    0
 1/2  notconnect auto      65    0

Port  Align-Err  FCS-Err    Xmit-Err   Rcv-Err    UnderSize
----- ---------- ---------- ---------- ---------- ---------
 1/1           0          0          0          0         0
 1/2           0          0          0          0         0

Port  Single-Col Multi-Coll Late-Coll  Excess-Col Carri-Sen Runts     Giants
----- ---------- ---------- ---------- ---------- --------- --------- ---------
 1/1           0          0          0          0         0         0         0
 1/2           0          0          0          0         0         0         0

Last-Time-Cleared
--------------------------
Tue Jun 8 1999, 10:01:35
Console> (enable)
```

This example shows how to see information on an individual port:

```
Console> (enable) show port 1/1
Port  Name               Status     Vlan       Duplex Speed Type
----- ------------------ ---------- ---------- ------ ----- ------------
 1/1                     connected  1            full  1000 1000BaseSX

Port  Security Secure-Src-Addr   Last-Src-Addr     Shutdown Trap     IfIndex
----- -------- ----------------- ----------------- -------- -------- -------
 1/1  disabled                                      No       disabled 3

Port     Broadcast-Limit Broadcast-Drop
-------- --------------- --------------------
 1/1                   -                    0
Port   Send FlowControl    Receive FlowControl   RxPause    TxPause
       admin    oper       admin    oper
-----  -------- --------   -------- --------     ---------- ----------
 1/1   desired  off        off      off          0          0

Port  Status     Channel   Admin Ch    Neighbor                          Neighbor
                 Mode      Group Id    Device                               Port
----- ---------- --------- ----- ----- ---------------------------------- -----
 1/1  connected  auto      65    0

Port  Align-Err  FCS-Err    Xmit-Err   Rcv-Err    UnderSize
----- ---------- ---------- ---------- ---------- ---------
 1/1           0          0          0          0         0

Port  Single-Col Multi-Coll Late-Coll  Excess-Col Carri-Sen Runts     Giants
----- ---------- ---------- ---------- ---------- --------- --------- ---------
 1/1           0          0          0          0         0         0         0

Last-Time-Cleared
--------------------------
Tue Jun 8 1999, 10:01:35
Console> (enable)
```

Displaying the Port MAC Address

In addition to displaying the MAC address range for a module using the **show module** command, you can display the MAC address of a specific port in the switch using the **show port mac-address** [*mod*[**/***port*]] command.

To display the MAC address of a specific port, perform this task in normal mode:

Task	Command
Display the MAC address of a specific port.	**show port mac-address** [*mod*[**/***port*]]

This example shows how to display the MAC address of a specific port:

```
Console> show port mac-address 4/1
Port  Mac address
----- -----------------------
 4/1  00-50-54-bf-59-64
```

This example shows how to display the MAC addresses of all ports on a module:

```
Console> show port mac-address 4
Port  Mac address
----- ---------------------
 4/1  00-50-54-bf-59-64
 4/2  00-50-54-bf-59-65
 4/3  00-50-54-bf-59-66
 4/4  00-50-54-bf-59-67
 4/47 00-50-54-bf-59-92
 4/48 00-50-54-bf-59-93
```

Displaying the Duplicate MAC Entries in the CAM Table

You can track multiple E-LAN VLANs and VLAN loops using the MAC duplication indicator (&) displayed next to the MAC entries that appear more than once in the CAM table.

To display the duplicate MAC entries in the CAM table, perform these tasks in enabled mode:

Task	Command
Display all duplicate MAC addresses in the CAM table.	**show cam duplicate**
Display only the dynamic MAC addresses with the duplicate indicator (&)	**show cam dynamic** [*mod*[**/***port*]]

The **show cam static | permanent** commands also display MAC entries with the duplicate indicator (&).

This example shows how to display all duplicate MAC entries in the CAM table:

```
Console> (enable) show cam duplicate
* = Static Entry. + = Permanent Entry. # = System Entry. R = Router Entry. X = Port
Security Entry $ = Dot1x Security Entry M = Mac-Auth-Bypass Entry & = Duplicate MAC entry

                                                 Destination Ports or
   VLAN  Dest MAC/Route Des    [CoS]     Age     VCs / [Protocol Type]
   ----  ------------------    -----  ----------  ---------------------
42    00-d0-02-83-eb-89  &                    3/3
   142    00-d0-02-83-eb-89  &                     5/3
    42    d8-d9-02-83-ef-ff  &                     2/3
     3    d8-d9-02-83-ef-ff  &                     3/4

Total Matching CAM Entries Displayed = 2

=========================================================================
```

Note If the **show cam duplicate** command delays the printing of duplicate entries, some of the entries might age out before the print operation is complete.

This example shows how to display only the dynamic MAC addresses with the duplicate indicator (&):

```
Console> (enable) show cam dynamic
* = Static Entry. + = Permanent Entry. # = System Entry. R = Router Entry. X = Port
Security Entry $ = Dot1x Security Entry M = Mac-Auth-Bypass Entry & = Duplicate MAC entry

                                                 Destination Ports or
   VLAN  Dest MAC/Route Des    [CoS]     Age     VCs / [Protocol Type]
   ----  ------------------    -----  ----------  ---------------------
```

```
   142   00-d0-02-94-4f-ff                       5/4
   42    00-d0-02-83-eb-fc                       2/1
   142   00-d0-02-83-eb-ff  &                    5/3
   Total Matching CAM Entries Displayed = 3

=======================================================================
```

Displaying Port Capabilities

You can display the capabilities of any port in a switch using the **show port capabilities** [[*mod*][*/port*]] command.

To display the capabilities of a specific port, perform this task in normal mode:

Task	Command
Display the capabilities of a specific port.	**show port capabilities** [*mod*[*/port*]]

This example shows how to display the port capabilities for switch ports:

```
Console> (enable) show port capabilities 1/1
Model                   WS-X6K-SUP1A-2GE
Port                    1/1
Type                    No Connector
Speed                   1000
Duplex                  full
Trunk encap type        802.1Q,ISL
Trunk mode              on,off,desirable,auto,nonegotiate
Channel                 yes
Broadcast suppression   percentage(0-100)
Flow control            receive-(off,on,desired),send-(off,on,desired)
Security                yes
Membership              static,dynamic
Fast start              yes
QOS scheduling          rx-(1p1q4t),tx-(1p2q2t)
CoS rewrite             yes
ToS rewrite             DSCP
UDLD                    yes
Inline power            no
AuxiliaryVlan           no
SPAN                    source,destination
COPS port group         1/1-2
Console> (enable)
```

Configuring the MAC Utilization Load Interval

These sections describe how to configure the MAC utilization load interval:

Understanding How the MAC Utilization Load Interval Works

The **show mac utilization** command displays the packet rate, bit rate, and octet rate per port, per module, and per VLAN, based on the load interval. You can set the load interval to either 30 or 300 seconds. You can also clear the MAC utilization counters on a port, range of ports, or for all ports in a module.

Setting the MAC Utilization Load Interval

You can set the MAC utilization load interval to 30 or 300 seconds. The default is 300 seconds.

To set the MAC utilization load interval, perform this task in enabled mode:

Task	Command
Set the MAC utilization load interval.	**set mac utilization load-interval** *seconds*

This example shows how to set the MAC utilization load interval to 30 seconds:

```
Console> (enable) set mac utilization load-interval 30
Load interval set to 30 seconds.
```

Displaying MAC Utilization Statistics

To display MAC utilization statistics, perform this task in enabled mode:

Task	Command
Display the MAC utilization statistics.	**show mac utilization** [**vlan** *number*]|[*mod[/port*]

This example shows how to display the MAC utilization statistics globally:

```
Console> (enable) show mac utilization

30 seconds input/output port rates:
Port  Xmit-Packet-Rate     Xmit-Octet-Rate      Xmit-Bit-Rate
----- -------------------- -------------------- --------------------
 2/1                555351             71088003            568704024
 2/2                555351             71088110            568704880
 2/3                555350             71088002            568704016
 2/14               555351             71088050            568704400
 2/15               555350             71088001            568704008
 2/16               555351             71088042            568704336
 3/1                     0                   12                   96
 3/2                     0                    0                    0
 3/3                     0                    0                    0
 3/43                    0                    7                   56
 3/44                    0                    4                   32
 3/45                    0                   46                  368
 3/46                    0                   46                  368
 3/47                    0                   40                  320
 3/48                    0                   40                  320
 4/1                     0                    0                    0
 4/2                     0                   18                  144
 4/3                     0                   18                  144
 4/4                     0                   18                  144
```

```
12/1                   369               23658              189264
12/2                     0                  12                  96
12/3                614539           921816483          7374531864
12/4                     0                   0                   0
13/1                 33960            50941147           407529176
13/2                 33960            50941151           407529208
13/3                 33960            50941190           407529520

Port  Rcv-Packet-Rate      Rcv-Octet-Rate       Rcv-Bit-Rate
----- ------------------- ------------------- -------------------
 2/1                845671           108247607           865980856
 2/2                555384            71090299           568722392
 2/3                555384            71090397           568723176
 2/4                555384            71090295           568722360
 2/5                555384            71090401           568723208
 2/6                555384            71090296           568722368
 2/16               845671           108247597           865980776
 3/1                     1                 129                1032
 3/2                     0                   0                   0
 3/3                     0                   0                   0
 3/4                     0                   0                   0
 3/29                    0                  73                 584
 3/43                    0                  73                 584
 3/44                    0                  73                 584
 3/45                    0                  14                 112
 3/46                    0                   9                  72
 3/47                    0                  12                  96
 3/48                    0                  12                  96
 4/1                     0                   0                   0
 4/2                     0                  18                 144
 4/3                     0                  18                 144
 4/25                    0                  18                 144
 4/26                    0                  18                 144
 4/27                    0                  18                 144
 4/28                    0                  18                 144
 4/29                    0                  18                 144
 8/1                     0                   0                   0
 8/2                     0                   0                   0
12/1                614201           921296589          7370372712
12/2                614198           921301441          7370411528
12/3                     0                  12                  96
12/4                     0                   0                   0
13/1                 82362           123544992           988359936
13/21                33960            50941535           407532280
13/22                33960            50940833           407526664
13/23                33960            50941552           407532416
Console> (enable)
```

This example shows how to display the MAC utilization statistics for a VLAN:

```
Console> (enable) show mac utilization vlan 100

300 seconds input/output port rates:
Port  Xmit-Packet-Rate     Xmit-Octet-Rate      Xmit-Bit-Rate
----- ------------------- ------------------- -------------------
13/1                 33925            50886135           407089080
13/26                33924            50885801           407086408

Port  Rcv-Packet-Rate      Rcv-Octet-Rate       Rcv-Bit-Rate
----- ------------------- ------------------- -------------------
13/1                 82278           123414184           987313472
13/26                33927            50887092           407096736
Console> (enable)
```

This example shows how to display MAC utilization statistics for a module:

```
Console> (enable) show mac utilization 12

30 seconds input/output port rates:
Port  Xmit-Packet-Rate     Xmit-Octet-Rate      Xmit-Bit-Rate
----- -------------------- -------------------- --------------------
12/1                396702            594010991           4752087928
12/2                395978            593964837           4751718696
12/3                412889            619338738           4954709904
12/4                396693            418773370           3350186960

Port  Rcv-Packet-Rate      Rcv-Octet-Rate       Rcv-Bit-Rate
----- -------------------- -------------------- --------------------
12/1                412891            619344814           4954758512
12/2                412891            619340051           4954720408
12/3                395978            593964450           4751715600
12/4                405223            425521134           3404169072
Console> (enable)
```

This example shows how to display MAC utilization statistics for a port:

```
Console> (enable) show mac utilization 12/1

30 seconds input/output port rates:
Port  Xmit-Packet-Rate     Xmit-Octet-Rate      Xmit-Bit-Rate
----- -------------------- -------------------- --------------------
12/1                405825            607683712           4861469696

Port  Rcv-Packet-Rate      Rcv-Octet-Rate       Rcv-Bit-Rate
----- -------------------- -------------------- --------------------
12/1                408276            612401845           4899214760
Console> (enable)
```

Clearing MAC Utilization Counters

To clear the MAC utilization counters, perform this task in enabled mode:

Task	Command
Clear the MAC utilization counters.	**clear mac utilization** [*mod*/*port*][1]

1. There is no option for this command.

This example shows how to clear the MAC utilization counters for a port:

```
Console> (enable) clear mac utilization 1/1
Mac utilization counters are cleared for the port 1/1.
```

This example shows how to clear the MAC utilization counters for a module:

```
Console> (enable) clear mac utilization 1
Module  1 mac utilization counters are cleared.
```

This example shows how to clear the MAC utilization counters globally:

```
Console> (enable) clear mac utilization
Mac utilization counters are cleared.
```

Checking the 10-Gigabit Ethernet Link Status

Cable diagnostics allow you to activate the pseudorandom binary sequence (PRBS) test on the 10-Gigabit Ethernet links.

Note The PRBS test is currently available on the 1-port 10GBASE-E serial 10-Gigabit Ethernet module (WS-X6502-10GE).

To run the PRBS test between two devices, you must start it on both ends of the cable. If the cable is looped back, a single end can generate the test sequence (on the Tx), verify the test sequence, and count the errors (at the Rx).

Before the PRBS test starts, the port is automatically put in the errdisable state. The errdisable timeout is disabled for the port so that the port is not automatically reenabled after the timeout interval ends. The errdisable timeout is automatically reenabled on the port after the PRBS test finishes.

When the PRBS test is running, the system does not allow you to enter the **set port enable** and **set port disable** commands.

The PRBS error counter measures the reliability of the cable. The error counter range is from 0–255. A value of 0 signifies a perfect link connection; a value of 255 signifies that the port is faulty, not connected, or that there is no communication through the link. If the counter does not remain at 0 for a predetermined length of time, the link is faulty. For example, for a baud error rate (BER) of 10^-12, the counter should remain at 0 for 100 seconds.

Each time that you access the PRBS counter by entering the **show port prbs** command, the PRBS error counter value is reset to 0, and the counter begins to accumulate errors again.

Note The PRBS counter is a “read and clear” register. The first reading in a sequence is usually unreliable and serves primarily to purge the counter; successive readings are accurate.

To start or stop the PRBS test, perform this task in privileged mode:

	Task	Command
Step 1	Start or stop the PRBS test.	**test cable-diagnostics prbs** {**start** | **stop**} *mod/port*
Step 2	Display the PRBS test counter information.	**show port prbs**

This example shows how to start the PRBS test on port 1 on module 5:

```
Console> (enable) test cable-diagnostics prbs start 5/1
PRBS cable-diagnostic test started on port 5/1.
Console> (enable)
```

This example shows how to stop the PRBS test on port 1 on module 5:

```
Console> (enable) test cable-diagnostics prbs stop 5/1
PRBS cable-diagnostic test stopped on port 5/1.
Console> (enable)
```

This example shows the message that displays when the PRBS test is not supported on a module:

```
Console> (enable) test cable-diagnostics prbs start 6/1
Feature not supported on module 6.
```

```
Console> (enable)
```

This example shows how to display the PRBS counter values and the ports that are running the PRBS test:

```
Console> (enable) show port prbs
Port PRBS state Error Counters
6/1 start 30
7/1 stop -
Console> (enable)
```

Checking the Cable Status Using TDR

You can check the status of the copper cables by using the time domain reflectometer (TDR). TDR is supported on the following modules: WS-X6148-GE-TX, WS-X6148V-GE-TX, WS-X6548-GE-TX, WS-X6548V-GE-TX, WS-X6548-GE-45AF, WS-X6748-GE-TX, WS-X6148A-GE-TX, WS-X6148-GE-45AF, WS-X6148A-GE-45AF, WS-X6148A-RJ-45, and WS-X6148A-45AF. The TDR detects a cable fault by sending a signal through the cable and reading the signal that is reflected back to it. All or part of the signal can be reflected back by any number of cable defects or by the end of the cable itself.

Note TDR can test cables up to a maximum length of 115 meters.

Use TDR to determine if the cabling is at fault if you cannot establish a link. This test is especially important when replacing an existing switch, upgrading to Gigabit Ethernet, or installing new cable plants.

To start or stop the TDR test, perform this task in privileged mode:

	Task	Command
Step 1	Start or stop the TDR test.	**test cable-diagnostics tdr** {**start** | **stop**} *mod/port*
Step 2	Display the TDR test counter information.	**show port tdr**

This example shows how to start the TDR test on port 1 on module 2:

```
Console> (enable) test cable-diagnostics tdr start 2/1
TDR test started on port 2/1. Use show port tdr <m/p> to see the results
Console> (enable)
```

This example shows how to stop the TDR test on port 1 on module 2:

```
Console> (enable) test cable-diagnostics tdr stop 2/1
tdr cable-diagnostic test stopped on port 2/1.
Console> (enable)
```

This example shows the message that displays when the TDR test is not supported on a module:

```
Console> (enable) test cable-diagnostics tdr start 2/1
Feature not supported on module 2.
Console> (enable)
```

This example shows how to display the TDR test results for a port:

```
Console> (enable) show port tdr 2/1
TDR test last run on Mon, March 10 2003 at 1:35:00 pm
```

```
Port  Speed  Local pair  Pair length         Remote pair  Pair status
----- ------ ---------- ------------------- ------------ ------------
2/1   1000   Pair A      12  +/- 3 meters    Pair A       Terminated
             Pair B      12  +/- 3 meters    Pair B       Terminated
             Pair C      12  +/- 3 meters    Pair C       Terminated
             Pair D      12  +/- 3 meters    Pair D       Terminated
```

Using Telnet

You can access the switch command-line interface (CLI) using Telnet. In addition, you can use Telnet from the switch to access the other devices in the network. Up to eight simultaneous Telnet sessions are possible.

To Telnet to another device on the network from the switch, perform this task in privileged mode:

Task	Command
Open a Telnet session with a remote host.	**telnet** *host* [*port*]

This example shows how to Telnet from the switch to a remote host:

```
Console> (enable) telnet labsparc
Trying 172.16.10.3...
Connected to labsparc.
Escape character is '^]'.

UNIX(r) System V Release 4.0 (labsparc)

login:
```

Using Secure Shell Encryption for Telnet Sessions

Note To use Secure Shell encryption commands, you must be running an encryption image. See Chapter 27, "Working with System Software Images" for the software image naming conventions that are used for the encryption images.

Note The Secure Shell encryption feature includes cryptographic software written by Eric Young (eay@cryptsoft.com).

Secure Shell encryption provides security for Telnet sessions and other remote connections to the switch. Secure Shell encryption is supported for remote logins to the switch only. Telnet sessions that are initiated from the switch cannot be encrypted. To use this feature, you must install the application on the client accessing the switch, and you must configure Secure Shell encryption on the switch.

The current implementation of Secure Shell encryption supports SSH version 1 and version 2. SSH version 1 supports DES and 3DES encryption methods, and SSH version 2 supports the 3DES and AES encryption methods. Secure shell encryption can be used with RADIUS and TACACS+ authentication. To configure authentication with Secure Shell encryption, enter the **telnet** keyword in the **set authentication** commands.

Note If you are using Kerberos to authenticate connections to the switch, you will not be able to use Secure Shell encryption.

Note Catalyst 6500 series software release 8.7(1) supports SSH keyboard interactive authentication methods such as S/KEY, one-time-pads, hardware tokens that print a number or string, and other legacy authentication methods with RADIUS and TACACS servers. For SSH keyboard interactive authentication to work, ensure that the **Apply password change rule** checkbox is checked on the Authentication Server Group Setup page on the RADIUS/TACACS server. The keyboard interactive authentication method works only with SSH V2 and the blank password mechanism is supported only with TACACS authentication.

To enable Secure Shell encryption on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Create the RSA host key.	**set crypto key rsa** *nbits* [**force**]
Step 2	Set the SSH version. **Note** If you do not specify the **v1** or the **v2** keyword, SSH operates in compatibility mode.	**set ssh mode** {**v1** | **v2**}
Step 3	Clear the SSH mode configuration.	**clear ssh mode**
Step 4	Display the SSH configuration information.	**show ssh**

This example shows how to create the RSA host key:

```
Console> (enable) set crypto key rsa 1024
Generating RSA keys.... [OK]
Console> (enable) set ssh mode v2
SSH protocol mode set to SSHv2 Only.
Console> (enable) show ssh
Session      Protocol       Cipher     State           PID     Userid       Host
-------      --------       ------     -----           ---     --------     -----
0            V2             3DES       SESSION_OPEN    146     dkoya        171.69.66.45
1            V1             3DES       SESSION_OPEN    147     -            dove.cisco.com
SSH server mode : V1 and V2
Console> (enable)
```

The *nbits* value specifies the RSA key size. The valid key size range is from 512–2048 bits. For SSH version 2, the minimum recommended key size is 768 bits. A key size with a larger number provides higher security but takes longer to generate.

You can enter the optional **force** keyword to regenerate the keys and suppress the warning prompt of overwriting existing keys.

Monitoring User Sessions

You can display the currently active user sessions on the switch using the **show users** command. The command output displays all the active console port and Telnet sessions on the switch.

To display the active user sessions on the switch, perform this task in privileged mode:

Task	Command
Display the currently active user sessions on the switch.	**show users** [**noalias**]

This example shows the output of the **show users** command when local authentication is enabled for console and Telnet sessions (the asterisk [*] indicates the current session):

```
Console> (enable) show users
  Session  User             Location
  -------- ---------------- -------------------------
  console
  telnet                    sam-pc.bigcorp.com
* telnet                    jake-mac.bigcorp.com
Console> (enable)
```

This example shows the output of the **show users** command when TACACS+ authentication is enabled for console and Telnet sessions:

```
Console> (enable) show users
  Session  User             Location
  -------- ---------------- -------------------------
  console  sam
  telnet   jake             jake-mac.bigcorp.com
  telnet   tim              tim-nt.bigcorp.com
* telnet   suzy             suzy-pc.bigcorp.com
Console> (enable)
```

This example shows how to display information about a user session using the **noalias** keyword to display the IP addresses of the connected hosts:

```
Console> (enable) show users noalias
  Session  User             Location
  -------- ---------------- -------------------------
  console
  telnet                    10.10.10.12
* telnet                    10.10.20.46
Console> (enable)
```

To disconnect an active user session, perform this task in privileged mode:

Task	Command
Disconnect an active user session on the switch.	**disconnect** {**console** | *ip_addr*}

This example shows how to disconnect an active console port session and an active Telnet session:

```
Console> (enable) show users
  Session  User             Location
  -------- ---------------- -------------------------
  console  sam
  telnet   jake             jake-mac.bigcorp.com
  telnet   tim              tim-nt.bigcorp.com
* telnet   suzy             suzy-pc.bigcorp.com
Console> (enable) disconnect console
Console session disconnected.
Console> (enable) disconnect tim-nt.bigcorp.com
Telnet session from tim-nt.bigcorp.com disconnected. (1)
Console> (enable) show users
  Session  User             Location
  -------- ---------------- -------------------------
  telnet   jake             jake-mac.bigcorp.com
* telnet   suzy             suzy-pc.bigcorp.com
Console> (enable)
```

Using Ping

These sections describe how to use IP ping:

Understanding How Ping Works

You can use IP ping to test connectivity to remote hosts. If you attempt to ping a host in a different IP subnetwork, you must define a static route to the network or configure a router to route between those subnets.

The **ping** command is configurable from normal EXEC mode and privileged EXEC mode. In normal EXEC mode, the **ping** command supports the **-s** parameter, which allows you to specify the packet size and packet count. In privileged EXEC mode, the **ping** command lets you specify the packet size, packet count, and the wait time.

Table 20-1 shows the default values that apply to the **ping**-**s** command.

Table 20-1 Ping Default Values

Description	Ping	Ping-s
Number of Packets	5	0=continuous ping
Packet Size	56	56
Wait Time	2	2
Source Address	Host IP Address	N/A

To stop a ping in progress, press **Ctrl-C**.

Ping returns one of the following responses:

- Normal response—The normal response (*hostname* is alive) occurs in 1 to 10 seconds depending on the network traffic.
- Destination does not respond—If the host does not respond, a no answer message is returned.
- Unknown host—If the host does not exist, an unknown host message is returned.
- Destination unreachable—If the default gateway cannot reach the specified network, a destination unreachable message is returned.
- Network or host unreachable—If there is no entry in the route table for the host or network, a network or host unreachable message is returned.

Executing Ping

To ping another device on the network from the switch, perform one of these tasks in normal or privileged mode:

Task	Command
Ping a remote host.	**ping** *host*
Ping a remote host using ping options.	**ping -s** *host* [*packet_size*] [*packet_count*]

This example shows how to ping a remote host from normal EXEC mode:

```
Console> ping labsparc
labsparc is alive
Console> ping 72.16.10.3
12.16.10.3 is alive
Console>
```

This example shows how to ping a remote host using the ping **-s** option:

```
Console> ping -s 12.20.5.3 800 10
PING 12.20.2.3: 800 data bytes
808 bytes from 12.20.2.3: icmp_seq=0. time=2 ms
808 bytes from 12.20.2.3: icmp_seq=1. time=3 ms
808 bytes from 12.20.2.3: icmp_seq=2. time=2 ms
808 bytes from 12.20.2.3: icmp_seq=3. time=2 ms
808 bytes from 12.20.2.3: icmp_seq=4. time=2 ms
808 bytes from 12.20.2.3: icmp_seq=5. time=2 ms
808 bytes from 12.20.2.3: icmp_seq=6. time=2 ms
808 bytes from 12.20.2.3: icmp_seq=7. time=2 ms
808 bytes from 12.20.2.3: icmp_seq=8. time=2 ms
808 bytes from 12.20.2.3: icmp_seq=9. time=3 ms

----17.20.2.3 PING Statistics----
10 packets transmitted, 10 packets received, 0% packet loss
round-trip (ms)  min/avg/max = 2/2/3
Console>
```

This example shows how to enter a **ping** command in privileged mode specifying the number of packets, the packet size, and the timeout period:

```
Console> (enable) ping
Target IP Address []: 12.20.5.19
Number of Packets [5]: 10
Datagram Size [56]: 100
Timeout in seconds [2]: 10
Source IP Address [12.20.2.18]: 12.20.2.18
!!!!!!!!!!

----12.20.2.19 PING Statistics----
10 packets transmitted, 10 packets received, 0% packet loss
round-trip (ms)  min/avg/max = 1/1/1
Console> (enable)
```

Using Layer 2 Traceroute

The Layer 2 Traceroute utility allows you to identify the physical path that a packet will take when going from a source to a destination. The Layer 2 Traceroute utility determines the path by looking at the forwarding engine tables of the switches in the path.

Information is displayed about all Catalyst 6500 series switches that are in the path from the source to the destination.

These sections describe how to use Layer 2 Traceroute:

- Layer 2 Traceroute Usage Guidelines, page 20-17
- Identifying a Layer 2 Path, page 20-18

Layer 2 Traceroute Usage Guidelines

This section describes the guidelines for using the Layer 2 Traceroute utility:

- The Layer 2 Traceroute utility works for unicast traffic only.
- You must enable Cisco Discovery Protocol (CDP) on all of the Catalyst 5000 and 6500 series switches in the network. (See Chapter 31, "Configuring CDP" for information about enabling CDP.) If any devices in the path are transparent to CDP, **l2trace** will not be able to trace the Layer 2 path through those devices.
- You can use this utility from a switch that is not in the Layer 2 path between the source and the destination; however, all of the switches in the path, including the source and destination, must be reachable from the switch.
- All switches in the path must be reachable from each other.
- You can trace a Layer 2 path by specifying the source and destination IP addresses (or IP aliases) or the MAC addresses. If the source and destination belong to multiple VLANs and you specify MAC addresses, you can also specify a VLAN.
- The source and destination switches must belong in the same VLAN.
- The maximum number of hops that an **l2trace** query will try is 10; this includes the hops that are involved in source tracing.
- The Layer 2 Traceroute utility does not work with Token Ring VLANs, when multiple devices are attached to one port through hubs, or when multiple neighbors are on a port.

Identifying a Layer 2 Path

To identify a Layer 2 path, perform one of these tasks in privileged mode:

Task	Command
(Optional) Trace a Layer 2 path using MAC addresses.	**l2trace** {*src-mac-addr*} {*dest-mac-addr*} [*vlan*] [**detail**]
(Optional) Trace a Layer 2 path using IP addresses or IP aliases.	**l2trace** {*src-ip-addr*} {*dest-ip-addr*} [**detail**]

This example shows the source and destination MAC addresses specified, with no VLAN specified, and the detail option specified. For each Catalyst 5000 and 6500 series switch found in the path, the output shows the device type, device name, device IP address, in port name, in port speed, in port duplex mode, out port name, out port speed, and out port duplex mode.

```
Console> (enable) l2trace 00-01-22-33-44-55 10-22-33-44-55-66 detail

l2trace vlan number is 10.

00-01-22-33-44-55 found in C5500 named wiring-1 on port 4/1 10Mb half duplex
C5500:wiring-1:192.168.242.10:4/1 10Mb half duplex -> 5/2 100MB full duplex
C5000:backup-wiring-1:192.168.242.20:1/1 100Mb full duplex -> 3/1 100MB full duplex
C5000:backup-core-1:192.168.242.30:4/1 100 MB full duplex -> 1/1 100MB full duplex
C6000:core-1:192.168.242.40:1/1 100MB full duplex -> 2/1 10MB half duplex.
10-22-33-44-55-66 found in C6000 named core-1 on port 2/1 10MB half duplex.
```

Using IP Traceroute

The IP Traceroute utility allows you to identify the path that packets take through the network at Layer 3 on a hop-by-hop basis. The command output displays all network layer (Layer 3) devices, such as the routers, that the traffic passes through on the way to the destination.

These sections describe how to use IP Traceroute:

- Understanding How IP Traceroute Works, page 20-18
- Executing IP Traceroute, page 20-19

Understanding How IP Traceroute Works

The **traceroute** command uses the Time To Live (TTL) field in the IP header to cause the routers and the servers to generate specific return messages. Traceroute starts by sending a User Datagram Protocol (UDP) datagram to the destination host with the TTL field set to 1. If a router finds a TTL value of 1 or 0, it drops the datagram and sends back an Internet Control Message Protocol (ICMP) time-exceeded message to the sender. The traceroute facility determines the address of the first hop by examining the source address field of the ICMP time-exceeded message.

To identify the next hop, traceroute sends a UDP packet with a TTL value of 2. The first router decrements the TTL field by 1 and sends the datagram to the next router. The second router sees a TTL value of 1, discards the datagram, and returns the time-exceeded message to the source. This process continues until the TTL is incremented to a value that is large enough for the datagram to reach the destination host (or until the maximum TTL is reached).

To determine when a datagram reaches its destination, traceroute sets the UDP destination port in the datagram to a very large value that the destination host is unlikely to be using. When a host receives a datagram with an unrecognized port number, it sends an ICMP port unreachable error to the source. This message indicates to the traceroute facility that it has reached the destination.

The switches can participate as the source or destination of the **traceroute** command but will not appear as a hop in the **traceroute** command output.

Executing IP Traceroute

To trace the path that the packets take through the network, perform this task in privileged mode:

Task	Command
Execute IP traceroute to trace the Layer 3 path that the packets take through the network.	**traceroute** [**-n**] [**-w** *wait_time*] [**-i** *initial_ttl*] [**-m** *max_ttl*] [**-p** *dest_port*] [**-q** *nqueries*] [**-t** *tos*] *host* [*data_size*]

This example shows how to use the **traceroute** command:

```
Console> (enable) traceroute 10.1.1.100
traceroute to 10.1.1.100 (10.1.1.100), 30 hops max, 40 byte packets
 1 10.1.1.1 (10.1.1.1)  1 ms  2 ms  1 ms
 2 10.1.1.100 (10.1.1.100)  2 ms  2 ms  2 ms
Console> (enable)
```

This example shows how to perform a **traceroute** with six queries to each hop with packets of 1400 bytes each:

```
Console> (enable) traceroute -q 6 10.1.1.100 1400
traceroute to 10.1.1.100 (10.1.1.100), 30 hops max, 1440 byte packets
 1 10.1.1.1 (10.1.1.1)  2 ms  2 ms  2 ms  1 ms  2 ms  2 ms
 2 10.1.1.100 (10.1.1.100)  2 ms  4 ms  3 ms  3 ms  3 ms  3 ms
Console> (enable)
```

Using System Warnings on Port Counters

You can monitor and troubleshoot the Catalyst 6500 series switches by polling the selected error counters on the ports and logging the system error messages. The messages are logged for the system, hardware, and spanning-tree ports for these conditions:

- Backplane traffic levels that exceed configurable thresholds
- Low remaining memory
- Detected memory corruption
- NVRAM logs
- Inband errors
- User Datagram Protocol (UDP) and TCP errors

The hardware error information is logged to provide information for the debug port counters at 30-minute intervals. The messages are logged if the counter values increase.

Spanning-tree error information is provided for the following:

- Ports that go from the blocking to the forwarding state
- Bridge protocol data unit (BPDU) skewing that exceeds a fixed threshold

These sections describe how to use the system warning feature on the Catalyst 6500 series switches:

- Executing System Warnings on Port Counters, page 20-20
- Executing Hardware Level Warnings on Port Counters, page 20-23
- Executing Spanning-Tree Warnings on Port Counters, page 20-23

Executing System Warnings on Port Counters

These sections describe how to execute the system warnings on the port counters:

- Backplane Traffic, page 20-20
- Low Remaining Memory, page 20-21
- Detected Memory Corruption, page 20-21
- NVRAM Logs, page 20-22
- Inband Errors, page 20-22
- UDP Errors, page 20-22

Backplane Traffic

You can configure backplane threshold detection by using a high threshold as a percentage. When backplane traffic goes over the specified threshold, compared with the previous traffic poll, a syslog message is generated. However, if you specify a 100-percent threshold (the default), no syslog message is generated.

For switches with three switching buses, you can configure a threshold and syslog throttling (to control the syslog event polling and message generation) for each switching bus instead of configuring the average traffic of all three buses. The throttle interval is 5 minutes.

This example shows how to set a threshold:

```
Console> (enable) set traffic monitor help
Usage: set traffic monitor <threshold>
       (threshold = 0..100 in percentage)
Console> (enable) set traffic monitor 60
Traffic monitoring threshold set to 60%.
Console> (enable) show traffic
Threshold: 60%

Backplane-Traffic Peak Peak-Time
----------------- ---- -------------------------
  0%                0% Tue Apr 16 2002, 08:01:53

Fab Chan Input Output
-------- ----- ------
       0    0%     0%
       1    0%     0%
       2    0%     0%
       3    0%     0%
       4    0%     0%
       5    0%     0%
```

```
        6    0%     0%
        7    0%     0%
        8    0%     0%
        9    0%     0%
       10    0%     0%
       11    0%     0%
       12    0%     0%
       13    0%     0%
       14    0%     0%
       15    0%     0%
       16    0%     0%
       17    0%     0%
Console> (enable)
```

Some sample syslog messages are as follows:

```
2000 Jan 11 06:00:27 PST -07:00 %SYS-4-SYS_HITRFC: 62% traffic detected on switching bus
(A)
2000 Feb 21 12:00:27 PST -07:00 %SYS-4-SYS_HITRFC: 65% traffic detected on switching bus
```

Low Remaining Memory

When memory allocation of clusters and buffers on the Catalyst 6500 series switch goes above a high watermark of 90 percent, the syslog messages are generated. These actions generate the syslog messages:

- When cluster allocation usage goes above a high watermark of 90 percent, the throttle interval is 1 hour.
- When mbufs allocation usage goes above a high watermark of 90 percent, the throttle interval is 1 hour.
- When malloc allocation usage goes above a high watermark of 90 percent, the throttle interval is 1 hour.

A sample syslog message is as follows:

```
1999 Sep 9 00:00:00 PDT -07:00 %SYS-3-SYS_MEMLOW: Memory cluster usage exceeded 90%
1999 Sep 9 00:00:00 PDT -07:00 %SYS-3-SYS_MEMLOW: Mbuf usage exceeded 90%
1999 Sep 9 00:00:00 PDT -07:00 %SYS-3-SYS_MEMLOW: Malloc usage exceeded 90%
```

Detected Memory Corruption

By default, memory corruption that is detected by the Memory Management Module (MMU) is enabled. This example shows how to enable memory corruption detection:

```
Console> (enable) set errordetection memory
Usage: set errordetection memory <enable|disable>
Console> (enable) set errordetection memory enable
Memory error detection enabled.
Console> (enable) show errordetection
Inband error detection:             disabled
Memory error detection:             enabled
Packet buffer error detection:      errdisable
Port counter error detection:       disabled
Port link-errors action:            port-failover
Port link-errors interval:          30 seconds
Port link-errors high rx-threshold: 1000 packets
Port link-errors low rx-threshold:  1000 packets
Port link-errors high tx-threshold: 1000 packets
Port link-errors low tx-threshold:  1000 packets
Port link-errors sampling:          3
```

```
Console> (enable)
```

A sample syslog message is as follows:

```
1999 Nov 23 16:32:21 PDT -07:00 %SYS-3-SYS_MEMERR: Out of range while freeing address
0xabcdefab
```

NVRAM Logs

The syslog errors are generated for each configuration-related NVRAM log event. These events may indicate configuration or hardware errors or NVRAM configurations that are made without notification of users.The hardware errors NVRAM log is not syslogged. The NVRAM log time stamp is not included in the message.

A sample syslog message is as follows:

```
1999 Nov 23 16:37:21 PDT -07:00 %SYS-4-SYS_NVLOG: convert_post_SAC_CiscoMIB:Block 63
converted from version 0 to 1

1999 Nov 23 16:37:25 PDT -07:00 %SYS-4-SYS_NVLOG: StartupConfig:Auto config started
```

Inband Errors

The inband syslog messages are generated when transmit or receive errors are detected. By default, the inband syslog messages are enabled. This example shows how to enable inband error detection:

```
Console> (enable) set errordetection inband
Usage: set errordetection inband <enable|disable>
Conosle> (enable) set errordetection inband enable
Inband errordetection enabled.
```

When the resource errors on the receive side reach a multiple of 500, this syslog error is generated:

```
2000 Jun 24 06:37:25 PDT -07:00 %SYS-3-INBAND_NORESOURCE: inband resource error warning
(500)
2000 Jun 24 08:12:03 PDT -07:00 %SYS-3-INBAND_NORESOURCE: inband resource error warning
(1000)
```

For each spurious interrupt, a message similar to the following is logged:

```
1999 Dec 25 18:22:08 PDT -07:00 %SYS-3-INBAND_SPRINTR: inband spurious interrupt occurred
(2)
```

For each inband port transmit and receive failure, a message similar to the following is logged:

Note The number in parentheses indicates the number of times that the inband port is reset instead of the number of transmit or receive fails.

```
1999 Dec 25 18:22:08 PDT -07:00 %SYS-3-INBAND_TXRXFAIL: inband driver stuck/reset (2)
```

UDP Errors

When you enter the **show netstat udp** command, each socket overflow generates a message similar to the following:

```
1999 Oct 31 23:59:59 PDT -07:00 %IP-3-UDP_SOCKOVFL: UDP socket overflow
```

When you enter the **show netstat udp/tcp** command, each bad UDP/TCP checksum generates a message similar to the following:

```
1999 Oct 31 23:59:59 PDT -07:00 %IP-3-UDP_BADCKSUM: UDP bad checksum
1999 Oct 31 23:59:59 PDT -07:00 %IP-3-TCP_BADCKSUM: TCP bad checksum
```

Executing Hardware Level Warnings on Port Counters

You can poll selected error counters of each switch port every 30 minutes. If the count goes up between two subsequent polls on the same port, the incidence is logged. Background polling is enabled or disabled by the **set errordetection portcounters** command. By default, polling is enabled.

Enter the **set errordetection portcounters** command as follows:

```
Console> (enable) set errordetection portcounters
Usage: set errordetection portcounters <enable|disable>
Console> (enable) set errordetection portcounters disable
Port Counters error detection disabled.
```

A sample syslog message is as follows:

```
1999 Jan 11 08:02:59 PDT -07:00 %SYS-3-PORT_ERR: Port 3/4 swBusResultEvent (12)
1999 Jan 11 09:03:03 PDT -07:00 %SYS-3-PORT_ERR: Port 3/4 swBusResultEvent (223)
1999 Jan 11 09:03:03 PDT -07:00 %SYS-4-PORT_WARN: Port 3/4 dmaTxFull (7) dmaRetry (33)
dmaLevel2Request(21)
```

Executing Spanning-Tree Warnings on Port Counters

These sections describe how to execute the spanning-tree warnings on the port counters:

- Blocking to Listening Transitions, page 20-23
- BPDU Skewing, page 20-23
- SNMP, page 20-24

Blocking to Listening Transitions

A syslog message is generated whenever a port goes from blocking to listening. The spanning-tree state changes have existing syslog messages.

A sample syslog messages is as follows:

```
1999 Jan 03 00:02:59 PDT -07:00 %SPANTREE-5-PORTLISTEN: Port 3/4 state in vlan 1 changed
to listening
1999 Jan 03 00:02:59 PDT -07:00 %SPANTREE-5-TR_PORTLISTEN: Trcrf 101 in trbrf 102 state
changed to listening
```

BPDU Skewing

A syslog message is generated when the interval between two consecutive BPDUs that are received on a port exceeds the hello time interval by 10 seconds. The throttle interval is one message per port, per minute for all VLAN numbers.

A sample syslog messages is as follows:

```
1999 Jan 01 00:01:19 PDT -07:00 %SPANTREE-3-BPDUSKEW: Port 2/1 vlan 1 BPDU skewed
1999 Jan 01 00:05:19 PDT -07:00 %SPANTREE-3-BPDUSKEW: Port 2/5 vlan 1 BPDU skewed
1999 Jan 01 00:05:23 PDT -07:00 %SPANTREE-3-BPDUSKEW: Port 2/5 vlan 3 BPDU skewed
```

SNMP

A matching SNMP trap generation for each of the syslog warnings using the existing clogMessageGenerated trap is sent every time that any syslog message is generated.

Configuring Packet-Buffer Error Handling

The **set errordetection packet-buffer** {**errdisable** | **powercycle** | **supervisor** {**errdisable** | **shutdown**}} command allows you to specify packet-buffer error handling as follows (the default is **errdisable**):

- **errdisable**—If you enter the **errdisable** keyword, the ports that experience the packet-buffer errors are put in the errdisable state.
- **powercycle**—If you enter the **powercycle** keyword, the modules supporting this option are power cycled when they encounter the packet-buffer errors. When you choose this option, a ROMMON upgrade is automatically performed on the module (if required), and the normal bootup sequence is bypassed to reduce the module downtime (this feature is also referred to as the rapid boot feature).
- **supervisor**—If you enter the **supervisor errdisable** keywords, the supervisor engine ports that experience the packet-buffer errors are put in the errdisable state. If you enter the **supervisor shutdown** keywords, the supervisor engine ports that experience the packet-buffer errors are shut down.

Caution Do not power cycle the module when the ROMMON image is downloading. Doing so might damage the module.

The rapid boot feature is available on the following modules:

- WS-X6248-RJ45
- WS-X6248-TEL
- WS-X6348-RJ45
- WS-X6348-RJ21
- WS-X6148-RJ45
- WS-X6148-RJ21

Note Enter the **show errordetection** command to display information about the error-detection configuration.

Configuring EtherChannel/Link Error Handling

This feature provides for an automatic failover of traffic from one port in an EtherChannel to another port in the same EtherChannel when one of the ports in the channel exceeds a configurable error threshold within the specified interval. The port failover only occurs if there is an operational port left in the EtherChannel. If the failed port is the last port in the EtherChannel, the port does not enter the "port failover" state and continues to pass traffic regardless of the type of errors being received. Single, nonchanneling ports do not go into the port failover state; these ports go into the errdisable state when the error threshold is exceeded within the specified interval.

Note The link errors that are monitored are based on three counters: Inerrors, RXCRC (CRCAlignErrors), and TXCRC. If the errdisable timer for the link is enabled (using the **set errdisable-timeout enable** command), the errdisabled port is automatically reenabled after the timeout interval expires (the timeout interval is specified using the **set errdisable-timeout interval** {*interval*} command). For more information on these commands, see the "Configuring a Timeout Period for Ports in errdisable State" section on page 4-12.

The **set errordetection link-errors** global command allows you to specify EtherChannel/link error handling as follows:

- **set errordetection link-errors action** {**errordisable** | **port-failover**}

  If a port's error count reaches the configurable threshold's high value (within the specified sampling count period), the *action* taken is either errordisable or port-failover. If you select **errordisable**, the port goes into the errdisable state when the high threshold is reached. If you select **port-failover**, the port's channel status is considered and the port goes into the errdisable state if the port is in a channel and it is not the last operational port in the channel (the port also goes into the errdisable state if it is a single port). The default **action** setting is **port-failover**.

- **set errordetection link-errors interval** {*timer-value*}

  The **interval** *timer-value* specified determines how often the port's error counters are read. The default timer value specified is 30 seconds, and the allowed range is from 30 to 1800 seconds.

  Note If the EtherChannel/link error handling feature is not enabled, you can still set the interval. If the feature is enabled, when you specify an interval, the timer restarts with the new value.

- **set errordetection link-errors** {**inerrors** | **rxcrc** | **txcrc**} {[**high** *value*] | [**low** *value*]}

  The **rxcrc** and **txcrc** values specified determine how many link errors are allowed during the specified interval by entering the **interval** *timer-value* command. If the low threshold is reached (within the sampling count period specified), a syslog message is displayed. If the high threshold is reached (within the sampling count period specified), in addition to displaying a syslog message, the port is either errdisabled or the port failover mechanism is triggered. The high threshold range is from 2 to 65535, and the low threshold range is from 1 to 65534. The **inerrors** values specified determines the inerrors threshold. The default threshold values are as follows:

  - The high value for the **inerrors** threshold is 1001 packets.
  - The low value for the **inerrors** threshold is 1000 packets.
  - The high value for the **rxcrc** threshold is 1001 packets.
  - The low value for the **rxcrc** threshold is 1000 packets.
  - The high value for the **txcrc** threshold is 1001 packets.
  - The low value for the **rxcrc** threshold is 1000 packets.

- **set errordetection link-errors sampling** {*sampling_count*}

  To minimize accidentally putting a port into the errdisable state due to a one-time event that is not a true system error condition, you can specify a *sampling_count*. The *sampling_count* determines the number of times that a port must reach the high or low threshold value before the port is placed in the errdisable state. For example, if the port's high threshold value is 1000 and the sampling count is 3, the port is errdisabled only after it has reached the 1000 threshold 3 times. The default sampling count value is 3, and the allowed range is from 1 to 255.

Note Enter the **show errordetection** command to display information about the error-detection configuration.

Configuring IEEE 802.3ah Ethernet OAM

The Ethernet Operations, Administrations, and Maintenance (OAM) feature follows the specifications provided in the IEEE 802.3ah document. The major Ethernet OAM features covered by this protocol are link monitoring, remote failure indication, and a remote loopback test.

Note We do not support remote failure indication.

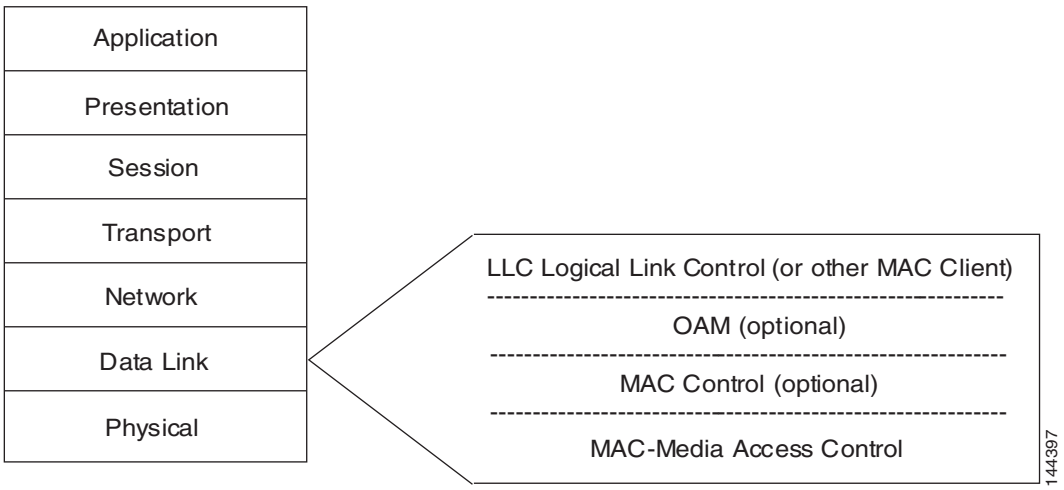
This section describes how to configure IEEE 802.3ah Ethernet OAM:

- Understanding How OAM Works, page 20-26
- Ethernet OAM Configuration Guidelines and Restrictions, page 20-27
- Executing Ethernet OAM, page 20-27

Understanding How OAM Works

In the Open Systems Interconnection (OSI) reference model, Ethernet OAM is an optional sublayer that is implemented in the data link layer between the logical link control (LLC) and MAC sublayers (see Figure 20-1).

Figure 20-1 Position of Ethernet OAM in the OSI Reference Model



Note OAM is a relatively slow protocol with low bandwidth requirements (the frame transmission rate is limited to a maximum of 10 frames per second), and it is not required for normal link operation. OAM frames, referred to as OAM protocol data units (OAMPDUs), use the slow protocol destination MAC address (0180.c200.0002), are intercepted by the MAC sublayer, and cannot propagate beyond a single hop within an Ethernet network.

You can implement OAM on any full-duplex point-to-point or emulated point-to-point Ethernet link. You configure OAM on a per-port basis and the OAM configuration is independent of any other configuration on the port. The port can be a trunk port, access port, or part of an EtherChannel. When you configure OAM on a port, that port's OAM functions are independent of the OAM functions that are configured on other ports.

Ethernet OAM Configuration Guidelines and Restrictions

Follow these configuration guidelines and restrictions when configuring Ethernet OAM:

- The OAM feature is only supported on physical, external Ethernet ports.
- The port that is running OAM must be in full-duplex mode.
- Remote failure indication is not supported.
- To support OAM remote loopback mode, the port needs to *specifically* be configured as follows:
  - The trunk mode must be set to **off**.
  - The channel mode must be set to **off**.
  - The port cannot be a private VLAN port.
- MIB variable requests and responses are not supported.

Executing Ethernet OAM

These sections describe how to execute Ethernet OAM:

Enabling or Disabling Ethernet OAM

You can use the commands in this section to enable or disable OAM on the specified ports. By default, OAM is disabled on all ports.

To enable or disable OAM on the specified ports, perform this task in privileged mode:

Task	Command
Enable or disable OAM on the specified ports.	**set port ethernet-oam** *mod/port* {**disable** | **enable**}

This example shows how to enable OAM on the specified port:

```
Console> (enable) set port ethernet-oam 3/1 enable
Successfully enabled OAM on port(s) 3/1.
Console> (enable)
```

Specifying the Ethernet OAM Port Mode

You can use the commands in this section to specify the OAM port mode on the specified ports. Table 20-2 lists the OAM port functions that are allowed in the active and passive modes. By default, the OAM mode is active on all ports.

Table 20-2 Ethernet OAM Port Modes

Capability	Active	Passive
Initiates the OAM discovery process.	Yes	No
Reacts to the OAM discovery process initiation.	Yes	Yes
Required to send information OAMPDUs.	Yes	Yes
Permitted to send event notification OAMPDUs.	Yes	Yes
Permitted to send variable request OAMPDUs.	Yes	No
Permitted to send variable response OAMPDUs.	Yes[1]	Yes
Permitted to send loopback control OAMPDUs.	Yes	No
Reacts to loopback control OAMPDUs.	Yes[1]	Yes
Permitted to send organization-specific OAMPDUs.	Yes	Yes

1. Requires the peer port to be in active mode.

To specify the OAM port mode on the specified ports, perform this task in privileged mode:

Task	Command
Specify the OAM port mode on the specified ports.	**set port ethernet-oam** *mod/port* **mode** {**active** | **passive**}

This example shows how to specify the OAM port mode to active on the specified port:

```
Console> (enable) set port ethernet-oam 3/1 mode active
Successfully updated OAM mode to active on port(s) 3/1.
Console> (enable)
```

Denying or Permitting Ethernet OAM Remote Loopback Tests

You can use the commands in this section to deny or permit an OAM remote loopback request on the specified ports. The default is permit.

To deny or permit an OAM remote loopback request on the specified ports, perform this task in privileged mode:

Task	Command
Deny or permit an OAM remote loopback request on the specified ports.	**set port ethernet-oam** *mod/port* **remote-loopback** {**deny** | **permit**}

This example shows how to deny an OAM remote loopback request on the specified port:

```
Console> (enable) set port ethernet-oam 3/1 remote-loopback deny
Successfully updated OAM remote-loopback capability to deny on port(s) 3/1.
Console> (enable)
```

Enabling or Disabling the Ethernet OAM Remote Loopback Test

You can use the commands in this section to enable or disable the OAM remote loopback test on the specified ports. The ports that you specify to run this test must be connected to a peer OAM device that is capable of entering into the OAM remote loopback mode.

Note The commands in this section are not saved in your configuration file or NVRAM.

Note During a remote loopback test operation, all packets including data packets are dropped at the port when remote loopback is enabled. This behavior results in many protocols (such as STP, EtherChannel protocols, and so on) resetting their state machines.

To enable or disable the OAM remote loopback test on the specified ports, perform this task in privileged mode:

Task	Command
Enable or disable the OAM remote loopback test on the specified ports.	**set port ethernet-oam** *mod/port* **remote-loopback** {**disable** | **enable**}

This example shows how to enable the OAM remote loopback test on the specified port:

```
Console> (enable) set port ethernet-oam 3/1 remote-loopback enable
Successfully initiated OAM remote-loopback on port(s) 1/1.
Port status set to inactive
Console> (enable)
```

Specifying the Number of Packets and the Packet Size for the Ethernet OAM Remote Loopback Test and Running the Test

You can use the commands in this section to specify the number of packets and the packet size for the OAM remote loopback test and run the test on the specified ports. This command can be used only on ports that have the OAM remote loopback test enabled. After entering this command, the remote loopback test is run and the test result summary is displayed after the test finishes. By default, 10,000 64-byte packets are sent. The number of packets allowed is 1 to 99999999 packets. The allowable packet size is from 64 to 1518 bytes.

Note The commands in this section are not saved in your configuration file or NVRAM.

To specify the number of packets and the packet size for the OAM remote loopback test and run the test on the specified ports, perform this task in privileged mode:

Task	Command
Specify the number of packets and the packet size for the OAM remote loopback test and run the test on the specified ports.	**set port ethernet-oam** *mod/port* **remote-loopback test** [*no of packets* [*packet size*]]

This example shows how to specify the number of packets for the OAM remote loopback test and run the test on the specified port:

```
Console> (enable) set port ethernet-oam 1/1 remote-loopback test 999999
Transmitting packets ...
OAM Remote Loopback Test 1/1: 999999 transmitted, 999999 received
Console> (enable)
```

Enabling or Disabling Ethernet OAM Link Monitoring

You can use the commands in this section to enable or disable OAM link monitoring on the specified ports. The default is enabled.

To enable or disable OAM link monitoring on the specified ports, perform this task in privileged mode:

Task	Command
Enable or disable OAM link monitoring on the specified ports.	**set port ethernet-oam** *mod/port* **link-monitor** {**disable** | **enable**}

This example shows how to enable OAM link monitoring on the specified port:

```
Console> (enable) set port ethernet-oam 3/1 link-monitor enable
Successfully enabled OAM link-monitor on port(s) 3/1.
Console> (enable)
```

Specifying the Window Size for Link Events for Ethernet OAM Link Monitoring

You can use the commands in this section to specify the OAM link monitoring window size for the corresponding link events. The defaults and ranges for the window sizes are as follows:

- **symbol-period**—The default is 625 million symbols. The range is from 1 to 1,000,000 in million-symbol increments.
- **frame**—The default is 30 seconds. The range is from 10 to 65535 in 100-millisecond increments (1 to 6553.5 seconds).
- **frame-period**—The default is 10 million frames. The range is from 200 to 2,000,000,000 frames.

To specify the OAM link monitoring window size for corresponding link events on the specified ports, perform this task in privileged mode:

Task	Command
Specify the OAM link monitoring window size for corresponding link events on the specified ports.	**set port ethernet-oam** *mod/port* **link-monitor** {**symbol-period** | **frame** | **frame-period**} **window** *size*

This example shows how to specify a link monitoring **symbol-period** window size of 10000:

```
Console> (enable) set port ethernet-oam 3/1 link-monitor symbol-period window 10000
Successfully updated OAM symbol-period window on port(s) 3/1.
Console> (enable)
```

This example shows how to specify a link monitoring **frame** window size of 100:

```
Console> (enable) set port ethernet-oam 3/1 link-monitor frame window 100
Successfully updated OAM frame window on port(s) 3/1.
Console> (enable)
```

This example shows how to specify a link monitoring **frame-period** window size of 1000:

```
Console> (enable) set port ethernet-oam 3/1 link-monitor frame-period window 1000
Successfully updated OAM frame-period window on port(s) 3/1.
Console> (enable)
```

Specifying the Low-Threshold Error Count and the Associated Action for Ethernet OAM Link Monitoring

You can use the commands in this section to specify the OAM link monitoring low-threshold error count and the associated action on the specified ports. The default low-threshold error count is one error. The default action is warning.

The low-threshold error count also serves as the monitoring threshold for OAM link monitoring. Once the specified low-threshold error count is met or exceeded, an OAM link event TLV is generated and sent as described in the IEEE 802.3ah document.

To specify the OAM link monitoring low-threshold error count and the associated action on the specified ports, perform this task in privileged mode:

Task	Command
Specify the OAM link monitoring low-threshold error count and the associated action on the specified ports.	**set port ethernet-oam** *mod/port* **link-monitor** {**symbol-period** | **frame** | **frame-period**} **low-threshold** *count* [**action** {**none** | **warning**}]

This example shows how to specify the OAM link monitoring low-threshold error count and the associated action on the specified port:

```
Console> (enable) set port ethernet-oam 3/1 link-monitor frame low-threshold 2 action none
Successfully updated OAM frame low threshold on port(s) 3/1.
Console> (enable)
```

Specifying the High-Threshold Error Count and the Associated Action for Ethernet OAM Link Monitoring

You can use the commands in this section to specify the OAM link monitoring high-threshold error count and the associated action on the specified ports. The default high-threshold error count is 65535 errors. The default action is warning.

To specify the OAM link monitoring high-threshold error count and the associated action on the specified ports, perform this task in privileged mode:

Task	Command
Specify the OAM link monitoring high-threshold error count and the associated action on the specified ports.	**set port ethernet-oam** *mod/port* **link-monitor** {**symbol-period** | **frame** | **frame-period**} **high-threshold** *count* [**action** {**errordisable** | **none** | **warning**}]

This example shows how to specify the OAM link monitoring high-threshold error count and the associated action on the specified port:

```
Console> (enable) set port ethernet-oam 3/1 link-monitor frame high-threshold 100 action
none
Successfully updated OAM frame-period high threshold on port(s) 3/1.
Console> (enable)
```

Specifying the Associated Action for OAM Critical Link Events

You can use the commands in this section to specify the associated action for OAM critical link events (**critical-event**, **dying-gasp**, or **link-fault**) on the specified ports. The default is **warning**. If you specify the **dying-gasp** keyword, the **errordisable** option is not available.

The **error-block** action sets the port to blocking state when a remote link failure flag is received and continues to monitor the link status flag. The error-block action then automatically changes the port to forwarding state whenever the remote link becomes operational.

To specify the associated action for OAM critical link events on the specified ports, perform this task in privileged mode:

Task	Command
Specify the associated action for OAM critical link events on the specified ports.	**set port ethernet-oam** *mod/port* {**critical-event** | **dying-gasp**[1] | **link-fault**} **action** {**errordisable** | **none** | **warning** | **error-block**}

1. Does not support **errordisable** and **error-block**.

This example shows how to specify the associated action for OAM critical link events on the specified port:

```
Console> (enable) set port ethernet-oam 3/1 link-fault action errdisable
Successfully updated OAM link-fault action on port(s) 3/1.
Console> (enable)
```

This example shows how to set the critical link event action to error-block for a port:

```
Console> (enable) set port ethernet-oam 3/2 critical-event action error-block
Successfully updated OAM critical-event action on port(s) 3/2.
```

Clearing Ethernet OAM Statistics and the Ethernet OAM Configuration

To clear OAM statistics and OAM-related configurations on all ports or individual ports, perform this task in privileged mode:

Task	Command
Clear OAM statistics and OAM-related configurations on all ports or individual ports.	**clear port ethernet-oam** [*mod/port*] [**statistics**]

This example shows how to clear the OAM configuration on all ports:

```
Console> (enable) clear port ethernet-oam
Successfully cleared OAM config on port(s) 2/1-2,3/1-48,8/1-8.
Console> (enable)
```

This example shows how to clear the OAM configuration from a specific port:

```
Console> (enable) clear port ethernet-oam 3/1
Successfully cleared OAM config on port(s) 3/1.
Console> (enable)
```

This example shows how to clear OAM statistics from all ports:

```
Console> (enable) clear port ethernet-oam statistics
```

```
Successfully cleared OAM statistics on port(s) 2/1-2,3/1-48,8/1-8.
Console> (enable)
```

This example shows how to clear OAM statistics from a specific port:

```
Console> (enable) clear port ethernet-oam 3/1 statistics
Successfully cleared OAM statistics on port(s) 3/1.
Console> (enable)
```

Clearing User-Configured Parameters for OAM Link Monitoring

When you clear the high-threshold or low-threshold parameters, the associated action is also cleared.

To clear the user-configured parameters for OAM link monitoring on the specified ports, perform this task in privileged mode:

Task	Command
Clear the user-configured parameters for OAM link monitoring on the specified ports.	**clear port ethernet-oam** *mod/port* **link-monitor** {**symbol-period** | **frame** | **frame-period**} {**high-threshold** | **low-threshold** | **window**}

These examples show how to clear the user-configured parameters for OAM link monitoring on the specified port:

```
Console> (enable) clear port ethernet-oam 3/1 link-monitor frame high-threshold
Successfully cleared OAM frame-period high-threshold on port(s) 3/1.
Console> (enable)

Console> (enable) clear port ethernet-oam 3/1 link-monitor frame-period window
Successfully cleared OAM frame-period window on port(s) 3/1.
Console> (enable)
```

Clearing User-Configured Actions for OAM Critical Link Events

To clear the user-configured actions for OAM critical link events on the specified ports, perform this task in privileged mode:

Task	Command
Clear the user-configured actions for the OAM critical link events on the specified ports.	**clear port ethernet-oam** *mod/port* {**critical-event** | **dying-gasp** | **link-fault**} **action**

These examples show how to clear the user-configured actions for OAM critical link events on the specified port:

```
Console> (enable) clear port ethernet-oam 3/1 link-fault action
Successfully cleared OAM link-fault action on port(s) 3/1.
Console> (enable)

Console> (enable) clear port ethernet-oam 3/1 critical-event action
Successfully cleared OAM critical-event action on port(s) 3/1.
Console> (enable)
```

Displaying Ethernet OAM-Related Information

To display the OAM configuration and status for all OAM ports or on the specified OAM ports, perform this task in normal mode:

Task	Command
Display the OAM configuration and status for all OAM ports or on the specified OAM ports.	**show port ethernet-oam** [*mod* | *mod/port*]

This example shows how to display the OAM configuration and status for the specified ports:

```
Console> (enable) show port ethernet-oam 1/1,3/5,4/6

Port  State    Mode    Status       LinkMonitor ConfigRev MaxPdu
----- -------- ------- ------------ ----------- --------- ------
1/1   enable   active  R-Loopback   enable      11        1518
3/5   enable   passive Connecting   enable      38        1518
4/6   disable  active  Operational  disable     0         1518

Port  Remote   Link    UniDir  Variable
      Loopback Event           retrieval
----- -------- ------- ------- ---------
1/1   Permit   enable  disable disable
3/5   Permit   enable  enable  disable
4/6   Deny     enable  disable disable

Port  ErrSymbol Period  ErrSymbol Period    ErrSymbol Period
      Window              LowThreshold        HighThreshold
      (millions)        Count     Action    Count     Action
----- ----------------- --------- --------- --------- ---------
1/1   625               1         None      10        Warning
3/5   65535             1         Warning   1000      ErrDisable
4/6   1                 1         None      1         ErrDisable

Port   Errored Frame    Errored Frame       Errored Frame
       Window            LowThreshold       HighThreshold
       (100 msec)      Count     Action    Count     Action
----- ----------------- --------- --------- --------- ---------
1/1   300               1         None      10        Warning
3/5   65535             1         Warning   1000      ErrDisable
4/6   1000              1         Warning   1         ErrDisable

Port  ErrFrame Period   ErrFrame Period     ErrFrame Period
      Window              LowThreshold        HighThreshold
                        Count     Action    Count     Action
----- ----------------- --------- --------- --------- ---------
1/1   10000             1         None      10        Warning
3/5   1294967000        1         Warning   1000      ErrDisable
4/6   200               1         Warning   1         ErrDisable

Port  LinkFaultAction   DyingGaspAction CriticalEventAction
----- ----------------- --------------- -------------------
1/1   ErrDisable        Warning         Error-Block
3/5   None              Warning         None
4/6   ErrDisable        None            None
Console> (enable)
```

Displaying Ethernet OAM Neighbor Information

You can use the commands in this section to display OAM neighbor information. The *neighbor* is the connected OAM peer.

To display OAM information for the specified neighbor or for all neighbors, perform this task in normal mode:

Task	Command
Display OAM information for the specified neighbor or for all neighbors.	**show port ethernet-oam** [*mod* | *mod/port*] **neighbor**

This example shows how to display OAM information for all neighbors:

```
Console> (enable) show port ethernet-oam neighbor
Port  MAC Addr           OUI    VendorInfo Mode    ConfigRev MaxPDU
----- ------------------ ------ ---------- ------- --------- ------
1/1   00-50-54-6c-b5-20 00000C 0000018C   passive 3         1518
3/5   00-0b-fc-fb-4a-10 00000C 0000018D   active  7         1518

Port  Remote   Link    UniDir  Variable
      Loopback Event           retrieval
----- -------- ------- ------- ---------
1/1   permit   enable  disable disable
3/5    deny      enable  enable   disable
Console> (enable)
```

Displaying Ethernet OAM Remote Loopback Test Information

You can use the commands in this section to display information about the OAM remote loopback test for the specified ports. The **current-session** keyword displays the statistics of the current OAM remote-loopback session. Specifying the **detail** keyword with the **current-session** keyword displays MAC statistics. The **last-session** keyword displays the statistics of the last OAM remote-loopback session. Specifying the **detail** keyword with the **last-session** keyword displays MAC statistics and statistics reported by the remote peer (if supported). After a port starts a new remote-loopback session, the last-session information becomes unavailable.

To display information about the OAM remote loopback test for the specified ports, perform this task in normal mode:

Task	Command
Display information about the OAM remote loopback test for the specified ports.	**show port ethernet-oam** [*mod* | *mod/port*] {**remote-loopback**} {**current-session** | **last-session**} [**detail**]

This example shows how to display information about the OAM remote loopback test for the current session:

```
Console> (enable) show port ethernet-oam 1/2 remote-loopback current-session

Port Loopback at OAM Rx     OAM Tx
---- ----------- ---------- ----------
1/2  Remote           33333      55555

Console> (enable) show port ethernet-oam 1/2 remote-loopback current-session detail
Port: 1/2
Loopback: Remote OAM in loopback mode
Start: Mon Aug 1 2005, 07:30:59
End: Still running
Test statistics:
OAM Rx: 10000
OAM Tx: 10000
MAC Rx: 13415
MAC Tx: 13403
OAMPDU Rx: 3415
OAMPDU Tx: 3403
MAC Rx Drop: 0

Console> (enable)
```

This example shows how to display information about the OAM remote loopback test for the last session:

```
Console> (enable) show port ethernet-oam 1/2 remote-loopback last-session

Port Last Loopback at   OAM Rx     OAM Tx
---- ----------------   ---------- ----------
1/2  Remote                  33333      55555

Console> (enable) show port ethernet-oam 1/2 remote-loopback last-session detail

Port: 1/2
Last Loopback: Remote OAM in loopback mode
Start: Mon Aug 1 2005, 07:30:59
End: Mon Aug 1 2005, 08:29:07
Test statistics:
OAM Rx: 10000
       OAM Tx: 10000
       MAC Rx: 13459
       MAC Tx: 13448
    OAMPDU Rx: 3459
    OAMPDU Tx: 3448
  MAC Rx Drop: 0
Test statistics reported by remote peer:
       OAM Rx: 10000
       OAM Tx: 10000
       MAC Rx: 13448
    OAMPDU Rx: 3448
  MAC Rx Drop: 0
Console> (enable)
```

Displaying Ethernet OAM Statistics

To display OAM statistics, perform this task in normal mode:

Task	Command
Display OAM statistics.	**show port ethernet-oam** [*mod* | *mod/port*] **statistics**

This example shows how to display OAM statistics for port 1/2:

```
Console> (enable) show port ethernet-oam 1/2 statistics
Port  InfoPduRx  UniEventRx DupEventRx RLBCtrlRx  VarReqRx   VarResRx
----- ---------- ---------- ---------- ---------- ---------- ----------
1/2   2579       0          0          0          0          0
Port  InfoPduTx  UniEventTx DupEventTx RLBCtrlTx  VarReqTx   VarResTx
----- ---------- ---------- ---------- ---------- ---------- ----------
1/2   2571       0          0          1          0          0
Port  OrgSpecRx  UnknownRx  CiscoPduRx CiscoTLVRx
----- ---------- ---------- ---------- ----------
1/2   0          0          0          1
Port  OrgSpecTx  UnknownTx  CiscoPduTx CiscoTLVTx
----- ---------- ---------- ---------- ----------
1/2   0          0          0          0
Console> (enable)
```

Configuring Metro Ethernet Connectivity Fault Management

This section describes how to configure Metro Ethernet Connectivity Fault Management (CFM). CFM is part of the Metro Ethernet OAM feature.

These sections describe how to configure Metro Ethernet CFM:

Understanding How Metro Ethernet Connectivity Fault Management Works

Metro Ethernet connects multiple customer sites to form one virtual private network (VPN). A Metro Ethernet network consists of networks from multiple operators that are supported by one service provider. Networks provided and managed by multiple independent service providers have restricted access to each other's equipment. Because of the diversity in these multiple-operator networks, failures must be isolated quickly. As a Layer 2 network, Ethernet must be capable of reporting network faults at Layer 2. IEEE 802.3ah is a point-to-point and per physical wire OAM protocol; it is not a service-aware switch protocol. IEEE 802.1ag CFM is a service level OAM protocol that provides tools for detecting and isolating connectivity failures in the network.

Connectivity Fault Management Protocols

IEEE 802.1ag draft 8.0 Metro Ethernet CFM incorporates several OAM facilities that allow you to manage Metro Ethernet networks, including an Ethernet continuity check, an end-to-end Ethernet traceroute, a Link Trace Message (LTM), a Loopback Message (LBM), and a Loopback Reply (LBR). These Metro Ethernet CFM elements allow you to quickly identify problems in your network.

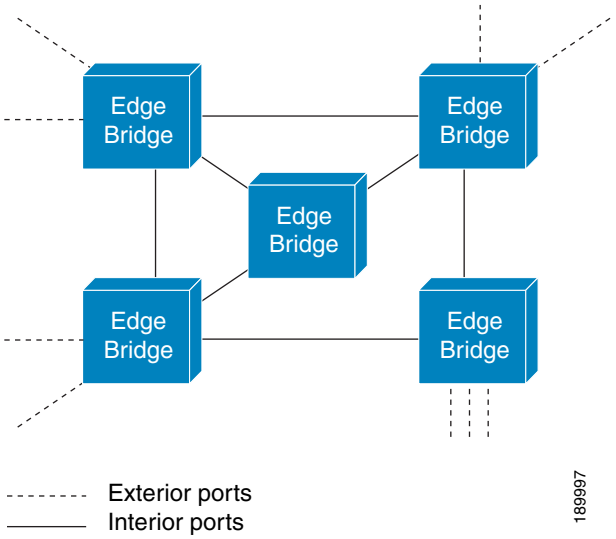
The following three protocols work together to help you debug Ethernet networks:

- Continuity Check—These heartbeat messages are issued periodically by the maintenance endpoints. They allow maintenance endpoints to detect a loss of service connectivity among themselves. They also allow maintenance endpoints to discover other maintenance endpoints within a domain and allow maintenance intermediate points to discover maintenance endpoints.
- Link Trace—These messages are transmitted by a maintenance endpoint by the request of the administrator to track the path (hop-by-hop) discovery to a destination maintenance endpoint. They allow the transmitting node to discover connectivity data about the path. Link trace messages are nonguaranteed datagram delivery packets that are transmitted similarly to User Datagram Protocol (UDP) traceroute messages.
- Loopback—These messages are transmitted by a maintenance endpoint by the request of the administrator to verify connectivity to a particular maintenance point. Loopback indicates whether the destination is reachable or not; it does not allow hop-by-hop discovery of the path. Loopback messages are similar to ICMP echoes (ping).

Maintenance Domains

Ethernet CFM, within any given service provider network, consists of hierarchical maintenance domains. A maintenance domain is an administrative domain for the purpose of managing and administering a network. A domain is assigned a unique maintenance level (among eight possible levels) by the administrator, which is useful for defining the hierarchical relationship of domains. Maintenance domains may nest or touch but cannot intersect. If the two domains nest, the outer domain must have a higher maintenance level that is contained within it. A maintenance domain is defined by determining which bridge ports are interior to the domain. See Figure 20-2 for an example of a maintenance domain.

Figure 20-2 Ethernet CFM Maintenance Domain



.

Often, three different organizations are involved in a Metro Ethernet service: customers, service providers, and operators. Customers purchase the Ethernet service from service providers. Service providers may use their own networks or the networks of other operators to provide connectivity for the requested service. Customers themselves may be service providers. For example, a customer may be an Internet service provider that sells Internet connectivity.

Nested maintenance domains allow the service provider to contract with one or more operators to provide the Ethernet service to a customer. In a nested domain, each operator has its own maintenance domain, the service provider defines its own domain that is a superset of the operators' domains, and the customer has its own end-to-end domain, which is a superset of the service provider's domain. In this scenario, the involved administering organizations communicate between the maintenance levels of the various nesting domains. For example, the service provider would assign the maintenance levels to the operators.

Maintenance Associations

A maintenance association identifies a service that can be uniquely identified within a maintenance domain. The CFM protocol runs within a particular maintenance association. An MA is a set of Maintenance End Points (MEPs), each configured with the same Maintenance Association ID (MAID) and a maintenance domain level established to verify the integrity of a single service instance. Multiple maintenance associations can exist within each maintenance domain.

Maintenance Points

Any port of a bridge is referred to as a maintenance point. A maintenance point may be classified as a maintenance endpoint, maintenance intermediate point, or transparent point for a maintenance level. Table 20-3 shows the maintenance point classifications.

Table 20-3 Maintenance Point Classifications

Functions	Maintenance Endpoint	Maintenance Intermediate Point	Transparent Point
Initiate CFM messages	Yes	No	No
Respond to loopback and link trace messages	Yes	Yes	No
Catalog continuity-check information received	Yes	Yes	No
Forward CFM messages	No	Yes	Yes

Maintenance endpoints reside at the edge of a maintenance domain, while maintenance intermediate points are internal to the domain. An intermediate point will forward CFM packets (unless it is a loopback or link trace destined for that intermediate point), while endpoints do not forward CFM packets because they must keep them within the domain. The only exception is when an endpoint is also acting as an intermediate point for a higher-level domain, in which case, it will forward the CFM packets as long as they are part of the higher-level domain.

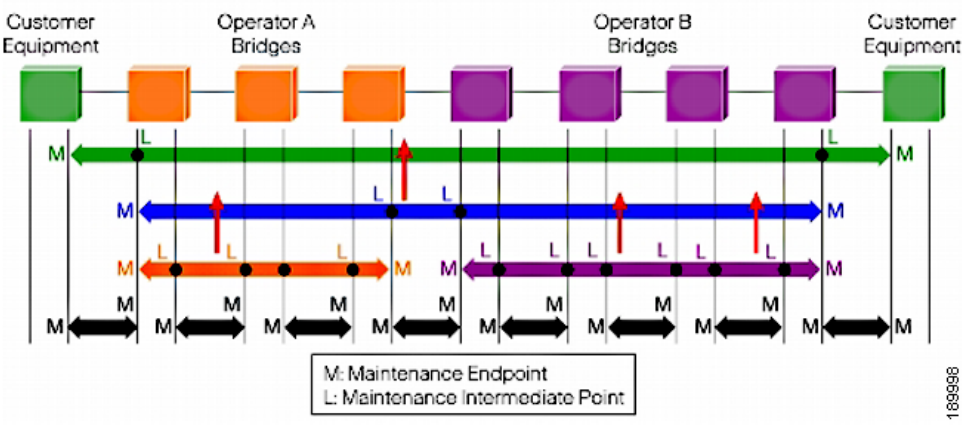
Figure 20-3 shows an example where a service provider is using the networks of two operators to provide service. The service provider maintenance level is shown in blue. The maintenance levels for Operator A and Operator B are shown in orange and violet, respectively. Two special-case maintenance levels are the customer level (shown in green) and the physical layer level (shown in black). The customer level allows the customer to test connectivity (using connectivity checks) and isolate issues (using loopback and link trace). The physical layer level defines the narrowest possible maintenance domain, which is a single link domain.

The designation of maintenance points as maintenance endpoints or maintenance intermediate points for the Operator A (or the Operator B) level is relative to that level only. When these maintenance points are observed relative to the service provider level, maintenance endpoints at the Operator A level translate into either maintenance endpoints or maintenance intermediate points at the service provider level. Maintenance intermediate points at the Operator A level translate into transparent points at the service provider level. Also, the demarcation of maintenance points as maintenance endpoints or maintenance intermediate points within a domain is left to the discretion of the administrator, because these points indicate points of particular relevance for the management of the network.

Figure 20-3 shows an example of how the CFM messages are used across the domains. The customer could use a CFM loopback or link trace to isolate a fault between the maintenance point M, which is on the Customer Premises Equipment (CPE), and the intermediate point L, which is on the user-facing provider edge equipment (U-PE). The link between the CPE and U-PE is a single hop. The customer would know which link has the fault. However, if the fault is between the two intermediate points (the Ls), the customer will need to rely on the service provider to determine between which maintenance (M) or intermediate (L) points that the fault has occurred. The service provider may simply isolate the fault to a specific operator's network and will rely on the operator to isolate the fault to a specific link in its network.

Each different organization (customer, service provider, and operator) can isolate the fault within the organization's maintenance level, without the service provider having to share its network information to the customer, or the operator having to share its network information to the service provider

Figure 20-3 Maintenance Points and Maintenance Domains



CFM Configuration Guidelines and Restrictions

When configuring CFM, follow these guidelines:

- The CFM configuration is allowed only in text configuration mode.
- The Spanning Tree mode should be Multiple Spanning Tree (MST).
- When configuring MEP, follow these steps:
  1. Configure the maintenance domain.
  2. Configure the maintenance association.
  3. Configure the MEPs.
- Configuring a maintenance domain and maintenance association is not mandatory when you configure an MIP.
- You should configure the EtherChannel before enabling CFM on member ports of the EtherChannel.
- Multiple maintenance associations cannot have the same VLAN within a particular domain.
- To avoid a misconfiguration error, use a unique MPID when configuring an MEP and when the customer shifts the local MEP from one port to another port (for the MEP that is down) or from one bridge brain switch to another bridge brain switch (for an MEP that is up).
- When configuring a maintenance association across two domains, a shared VLAN is allowed only if a maintenance association is configured at different levels.
- To optimize on the packet processing time, the sender ID Type-Length-Value (TLV) has been removed from the Continuity Check (CC) packet .The organisational specific TLV which contains the ELMI specific information for the remote MEPs gets populated only, when ELMI is enabled globally. However, all the standard defined TLVs are processed when the CC is being received from the network.

Note CFM domains, maintenance associations and maintenance points can be configured in binary mode or with any other Spanning Tree mode. CFM protocol will be functional only when it is globally enabled in the text configuration mode.

Scalability Data for Connectivity Fault Management and Alarm Indication Signal

- On a Catalyst 6500 series switch with Supervisor Engine 720 that runs software release 8.7(3), when CFM or CFM with MVRP are enabled together on dot1q trunk ports with a 10 second CC interval, the switch supports the following:
  - The CCM traffic up to 2000 services or VLANs.
  - The Customer Edge (CE) switch supports 2000 customer level MIPs, and 2000 higher-level flood traffic (traffic coming at the level higher than the maximum Maintenance level configured on the switch).
  - The Provider Edge (PE) switch up to 200 upward MEPs.

Caution An increase in number of MIPs, provider level MEPs or higher level flood traffic will increase the CPU utilization, and might degrade performance of the system.

- On a Catalyst 6500 series switch with Supervisor Engine 720 that runs software release 8.7(3), when CFM or CFM with MVRP enabled together on the EtherChannel ports (4 ports in a bundle) and with a 10 seconds CC interval, the switch supports the following:
  - The CCM traffic up to 1000 services or VLANs.
  - 1000 customer level MIPs and 1000 higher-level flood traffic (traffic coming at the level higher than the maximum Maintenance level configured on the switch).
  - 200 Provider Level Up MEPs.

Caution An increase in the number of ports to the EtherChannel or increase in the number of MIPs on a bundled port, will increase the CPU utilization. This may result in CC lifetime expiry for the remote MEPs, and trigger false indication of fault in the network.

- On a Catalyst 6500 series switch with Supervisor Engine 720 that runs software release 8.7(3), when CFM-AIS or CFM-AIS and MVRP are enabled together on dot1q trunk/EtherChannel ports with 10 second CC interval the switch supports the following:
  - In the event of link failure, the switch supports CCM traffic for up to 2000 services in the normal state.
  - The switch supports 2000 customer level MIPs and 2000 higher-level flood traffic (traffic coming at the level higher than the maximum Maintenance level configured.
  - Up to 200 Provider level Up MEPs.

Note In the event of link failure, a CPU spike occurs at every one minute time interval because of the AIS timer spread logic.

Caution An increase in the number of MIPs, provider level MEPs or higher level flood traffic will increase the CPU utilization and may degrade system performance.

Note On a Catalyst 6500 series switch that runs software release 8.7(3), when an AIS detects the link fault condition occurs the configured number of AIS PDUs will be sent (default 5) at 1 second transmission interval for each of the affected VLAN on the failed trunk. Then the AIS transmission period is changed to 1 minute automatically in the software (timer spread logic). This will increase the CPU utilization at every 1 minute until the fault condition is cleared, which is an expected behavior.

Configuring Metro Ethernet CFM

Note For complete syntax and usage information for the commands that are used in this section, refer to the *Catalyst 6500 Series Switch Command Reference Software Release 8.x* publication.

This section describes how to configure Metro Ethernet CFM:

Enabling or Disabling Metro Ethernet CFM

To enable or disable Metro Ethernet CFM globally on a switch, perform this task in privileged mode:

Task	Command
Enable or disable Metro Ethernet CFM globally on a switch.	**set ethernet-cfm** {**disable** | **enable**}

This example shows how to enable Metro Ethernet CFM globally on a switch:

```
Console> (enable) set ethernet-cfm enable
Ethernet CFM enabled.
Console> (enable)
```

Configuring Metro Ethernet CFM Domains

To create a maintenance domain and configure the maintenance level, perform this task in privileged mode:

Task	Command
Configure a Metro Ethernet CFM domain.	**set ethernet-cfm domain** *domain name* **level** *level*

This example shows how to configure a maintenance domain with domain name customerxy Domain and at level 6:

```
Console> (enable) set ethernet-cfm domain customerXYDomain level 6
Created a Domain customerXYDomain at level 6.
Console> (enable)
```

Configuring a Metro Ethernet CFM Maintenance Association

To configure a maintenance association within the maintenance domain, perform this task in privileged mode:

Task	Command
Configure a Metro Ethernet CFM maintenance association within the maintenance domain.	**set ethernet-cfm maintenance-association ma-name-fmt** *fmt name* | *value* **domain** *domain-name* **vlan** *vlan_id* [**direction** *up* | *down*]

This example shows how to configure a maintenance association in a domain with a VLAN ID:

```
Console> (enable) set ethernet-cfm maintenance-association ma-name-fmt text customerXMA
domain customerXYDomain vlan 1 direction up
Maintenance Association created successfully for vlan 1 in domain customerXYDomain
Console> (enable)
```

Configuring CFM on a Port as a Maintenance Point

To enable or disable CFM on a port and to configure a port as a Maintenance End Point (MEP) and Maintenance Intermediate Point (MIP) for a specific maintenance level and VLAN, perform this task in privileged mode:

	Task	Command
Step 1	Configure Ethernet CFM on a specific module and port or set the port to transparent mode.	**set port ethernet-cfm** *mod/por*t {**enable** | **disable** | **transparent**}
Step 2	Configure a port as an MEP and configure an MPID for a specific maintenance level.	**set port ethernet-cfm** *mod/port* **mep mpid** *mpid* **domain** *domain-name* **vlan** *vlan-id*
Step 3	Configure an MIP for a specific domain or a specific maintenance level.	**set port ethernet-cfm** *mod/port* **mip level** *level* **vlan** *vlan-id*

This example shows how to initialize an MEP at module 1, port 1, for VLAN 10:

```
Console> (enable) set port ethernet-cfm 1/1 mep mpid 1 domain XYZ vlan 10
MEP is configured for port  1/1 with MPID 1 in domain XYZ for vlan10.
Console> (enable)
```

This example shows how to initialize an MIP at module 1, port 1 at MIP level 5:

```
Console> (enable) set port ethernet-cfm 1/1 mip level 5 vlan 10
MIP is configured for port  1/1 at level 5 for vlan(s)10.
Console> (enable)
```

Configuring Continuity-Check Protocol Parameters

To configure continuity-check message attributes for a specific level of the local Maintenance End Points (MEPs), perform this task in privileged mode:

Task	Command
Configure continuity-check message attributes for a specific level of MEPs.	**set ethernet-cfm continuity-check level** *level* **vlan** *vlans* **interval** *interval-value* [**loss-threshold** threshold]

This example shows how to configure continuity-check message attributes for level 5, VLAN ID 11 at an interval of 1 minute and a loss threshold of three messages:

```
Console> (enable) set ethernet-cfm continuity-check level 5 vlan 11 interval 2
loss-threshold 3
CC Attributes set for level(s)5
Console> (enable)
```

This example shows how to enable the continuity-check protocol for a particular maintenance association or VLAN at a specific level:

```
Console> (enable) set ethernet-cfm continuity-check level 4 vlan 100
```

```
Succesfully enabled CC for level 4 for vlan(s) 100.
Console> (enable)
```

Configuring Ethernet CFM traceroute Protocol Parameters

To enable or disable caching of Ethernet Connectivity Fault Management (CFM) data entered using traceroute messages, perform this task in priviliged mode:

	Task	Command
Step 1	Enable or disable caching of Ethernet CFM data.	**set ethernet-cfm traceroute-database** [**enable** | **disable**]
Step 2	Set the size of the traceroute database. The size varies from 1 to 4095 entries.	**set ethernet-cfm traceroute-database size** *size*
Step 3	Configure the time for retaining the entry in the traceroute database. This time varies from 1 to 2880 minutes.	**set ethernet-cfm traceroute-database hold-time** *hold_time*

This example shows how to enable the caching of the Ethernet CFM data:

```
Console> (enable) set ethernet-cfm traceroute-database enable
Ethernet TRDB Cache enabled
Console> (enable)
```

This example shows how to set the hold time to 300 in the traceroute database:

```
Console> (enable) set ethernet-cfm traceroute-database hold-time 300
Ethernet TRDB hold-time is set to 300 minutes
Console> (enable)
```

This example shows how to set the size of the traceroute database to 300:

```
Console> (enable) set ethernet-cfm traceroute-database size 300
Ethernet TRDB size is set to 300.
Console> (enable)
```

Configuring a System CAM Entry

To configure a system CAM entry for a specified module, port number, and a specific VLAN or VLANs, perform this task in priviledged mode:

Task	Command
Configure a system CAM entry for a specified module, port number, and VLAN.	**set ethernet-cfm port-mac-enable** *mNo/pNo* **vlan** *vlans*

This example shows how to configure a system CAM entry for module 6, port 2, and VLAN 10:

```
Console>(enable) set ethernet-cfm port-mac-enable 6/2 vlan 10
CAM table updated with entries for port(s) 6/2 vlan(s) 10
Console>(enable)
```

Note For LTMs/LBMs to be successful with DOWN MEPs, you should configure the system CAM entry for that VLAN on the DOWN MEP port.

Displaying Metro Ethernet CFM Domains

To display all the configured CFM domains, perform this task in privileged mode:

Task	Command
Display the domains configured for a switch.	**show ethernet-cfm domain** [*domain_name*] **detail**

This example shows how to display information on all the domains on the switch:

```
Console> (enable)
Console> (enable) show ethernet-cfm domain
-----------------------------------------------------
Domain Name                Index   Level   Services
-----------------------------------------------------
dom3                           1     3     2000
dom6                           2     6     0
dom7                           3     7     0
Console> (enable)
```

This example shows how to display information on only the **sjlabf1** domain:

```
Console> (enable) show ethernet-cfm domain customerXYZ detail
* - indicates vlan does not exist
$ - indicates vlan is suspended

Domain ID : 2
Domain Name : customerXYZ
Level : 4
Total Services : 1
Services :
Vlan Direction CC-Enable shortMAName
100 Up Y CUST-MA-10

Console> (enable)
```

Displaying CFM Maintenance Association Information

To display all the maintenance association information, perform this task in privileged mode:

Task	Command
Display all the maintenance association information.	**show ethernet-cfm maintenance-association**

```
Console> (enable) show ethernet-cfm maintenance-association
Maintenance Association Details :

* - indicates vlan does not exist
$ - indicates vlan is suspended
```

```
-------------------------------------------------------------------------------------
Vlan  Dir  Domain                Lvl MA      MA-Name     CC-    Loss  CC-    AIS
           Name                    Format              Intv   Thres Enable state
-------------------------------------------------------------------------------------
2000  up   dom3                   3 text    vlan2000    10 sec   3   TRUE   TRUE
2001  up   dom3                   3 text    vlan2001    10 sec   3   TRUE   TRUE
2002  up   dom3                   3 text    vlan2002    10 sec   3   TRUE   TRUE
2003  up   dom3                   3 text    vlan2003    10 sec   3   TRUE   TRUE
2004  up   dom3                   3 text    vlan2004    10 sec   3   TRUE   TRUE
2005  up   dom3                   3 text    vlan2005    10 sec   3   TRUE   TRUE
2006  up   dom3                   3 text    vlan2006    10 sec   3   TRUE   TRUE
2007  up   dom3                   3 text    vlan2007    10 sec   3   TRUE   TRUE
2008  up   dom3                   3 text    vlan2008    10 sec   3   TRUE   TRUE
2009  up   dom3                   3 text    vlan2009    10 sec   3   TRUE   TRUE
2010  up   dom3                   3 text    vlan2010    10 sec   3   TRUE   TRUE
Console> (enable)
```

Displaying Metro Ethernet CFM Maintenance Point Information

To display all the local or remote maintenance points, perform this task in privileged mode:

Task	Command
Display all local or remote maintenance points.	**show ethernet-cfm maintenance-point** {**local** | **remote**}

This example shows how to display local MEPs/MIPs configured on the switch:

```
Console> (enable) show ethernet-cfm maintenance-point local
* - indicates vlan does not exist
$ - indicates vlan is suspended
@ - indicates vlan is not allowed on this port
LOCAL MEPS:
-------------------------------------------------
Port MPID Dir    Level Domain CC   Vlan MA-name
                       Name stat
-------------------------------------------------
3/20 200 DOWN  4      xyz     1      10     MA-10
Total Local MEP's = 1
LOCAL MIPS:
----------------------
Port Level Vlans
----------------------
3/20   5    10
Total Local MIP's = 1
Console> (enable)
```

This example shows how to display remote maintenance points:

```
Console> (enable) show ethernet-cfm maintenance-point remote
* - indicates port is a channel port
-----------------------------------------------------
MPID Port Vlan Level Mac-Addr Domain Name MA Name RDI
-----------------------------------------------------
200 3/14 10 4 00-30-19-c0-a0-a5 cust-1     MA-10   n
200 3/14 20 4 00-30-19-c0-a0-a5 cust-1     MA-20   n
200 3/14 30 4 00-30-19-c0-a0-a5 cust-1     MA-30   n
200 3/14 40 4 00-30-19-c0-a0-a5 cust-1     MA-40   n
200 3/14 50 4 00-30-19-c0-a0-a5 cust-1     MA-50   n
Console>
```

Displaying the Metro Ethernet CFM Status

To display the global CFM and AIS status, the maximum configured maintenance level, and CFM MAC addresses, perform this task in privileged mode:

Task	Command
Display the global CFM and AIS status and the maximum configured maintenance level.	**show ethernet-cfm status**

This example shows how to display the CFM and AIS status:

```
Console> (enable) show ethernet-cfm status
Ethernet CFM is enabled on this switch.
Max configured level is 4.
Bridge Brain Mac Address is 00-13-5f-1f-67-3b.
CFM CC Multicast Address is 01-80-c2-00-00-30.
CFM LTM Multicast Address is 01-80-c2-00-00-38.
CFM AIS is enabled.
CFM AIS Default Transmission Interval is 1 sec.
CFM AIS configured level is 8.
CFM AIS PDUs to be transmitted at 1 sec Interval is 8.
Console> (enable)
```

Displaying Metro Ethernet CFM Statistics

To display the CFM packet statistics, such as the Continuity Check Messages (CCMs) sent, CCMs received with out-of-order transaction IDs, Loopback Replies (LBRs), or Linktrace Replies (LTRs), perform this task in privileged mode:

Task	Command
Display continuity check packet statistics.	**show ethernet-cfm statistics** [**mpid** *mpid*]

This example shows how to display the CFM statistics:

```
SW8> (enable)
SW8> (enable)
SW8> (enable) show ethernet-cfm statistics

* - indicates vlan does not exist
$ - indicates vlan is suspended
@ - indicates vlan is not allowed on this port

-------------------------------------------------------------------------------
MPID  Port   Vlan   CCM    CCM Seq  LTR        LBR   LBR     LBR     LBR
                    Sent   Error    unexpected sent  seq-err recvd   bad-msdu
-------------------------------------------------------------------------------
2001  3/1    2000   1407   20       0          5     0       0       0
2002  3/1    2001   1407   21       0          0     0       0       0
2003  3/1    2002   1407   15       0          0     0       0       0
2004  3/1    2003   1407   36       0          0     0       0       0
2005  3/1    2004   1406   32       0          0     0       0       0
```

```
2006  3/1    2005   1406   33       0          0      0       0       0
2007  3/1    2006   1406   23       0          0      0       0       0
2008  3/1    2007   1406   36       0          0      0       0       0
2009  3/1    2008   1406   18       0          0      0       0       0
2010  3/1    2009   1405   34       0          0      0       0       0
2011  3/1    2010   1407   22       0          0      0       0       0
2012  3/1    2011   1406   36       0          0      0       0       0
2013  3/1    2012   1407   20       0          0      0       0       0
2014  3/1    2013   1405   33       0          0      0       0       0
Console> (enable)
```

Displaying Metro Ethernet CFM Errors

To display the CFM continuity check and AIS error conditions logged since the last reload, perform this task in privileged mode:

Task	Command
Displays CFM and AIS error condition.	**show ethernet-cfm errors** {**domain** *domain-name*}

This example shows how to display Ethernet CFM errors:

```
Console> (enable) show ethernet-cfm errors
--------------------------------------------------------------------------------------------
Lvl Vlan MPID Remote-MAC        Reason           MA-Name              Domain-Name
--------------------------------------------------------------------------------------------
6   2816 8190 00-0b-45-a8-c4-3b AIS-Error        vlan2816             dom6
6   2816 817  00-11-bc-99-af-fb Lifetime-Expiry  vlan2816             dom6
6   2560 8190 00-0b-45-a8-c4-3b AIS-Error        vlan2560             dom6
6   2560 561  00-11-bc-99-af-fb Lifetime-Expiry  vlan2560             dom6
6   2304 8190 00-0b-45-a8-c4-3b AIS-Error        vlan2304             dom6
6   2304 305  00-11-bc-99-af-fb Lifetime-Expiry  vlan2304             dom6
6   2048 8190 00-0b-45-a8-c4-3b AIS-Error        vlan2048             dom6
6   2048 49   00-11-bc-99-af-fb Lifetime-Expiry  vlan2048             dom6
6   3328 1329 00-11-bc-99-af-fb Lifetime-Expiry  vlan3328             dom6
6   3072 1073 00-11-bc-99-af-fb Lifetime-Expiry  vlan3072             dom6
6   3840 1841 00-11-bc-99-af-fb Lifetime-Expiry  vlan3840             dom6
6   3584 1585 00-11-bc-99-af-fb Lifetime-Expiry  vlan3584             dom6
6   2817 8190 00-0b-45-a8-c4-3b AIS-Error        vlan2817             dom6
6   2817 818  00-11-bc-99-af-fb Lifetime-Expiry  vlan2817              dom6
Console> (enable)
```

Displaying the Metro Ethernet CFM traceroute Database

To display the contents of the traceroute database, perform this task in priviledged mode:

Task	Command
Display the contents of the traceroute database.	**show ethernet-cfm traceroute-database** [**status** | **size** | **hold time**]

This example shows how to display the contents of the traceroute database:

```
Console> (enable) show ethernet-cfm traceroute-database
```

```
Traceroute to 00-50-3e-78-fb-fb on Domain dom3, Level 3,
Vlan 2000 issued at Wed Aug 12 2009, 03:12:17

B = Intermediary Bridge
! = Target Destination
* = Per hop Timeout
-------------------------------------------------------------------------------
                             MAC          Ingress      Ingr Action  Relay Action

  Hops   Host             Forwarded       Egress       Egr Action   Prev Hop

-------------------------------------------------------------------------------
B  1  y69              00-12-da-66-76-3b     4/13        IngOK         RlyMPDB
                            Forwarded                          00-0f-f8-8d-cb-fb
B  2  y72              00-0f-f8-8a-d0-7b                        RlyMPDB
                            Forwarded        3/5        EgrOK 00-12-da-66-76-3b
!  3  y90              00-50-3e-78-fb-fb                         RlyHit
                          Not Forwarded                        00-0f-f8-8a-d0-7b
```

Clearing a Metro Ethernet CFM

To clear CFM parameters, perform one of these tasks in privileged mode:

Task	Command
Clear a Metro Ethernet CFM domain.	**clear ethernet-cfm domain** *domain_name* **level** *level.*
Clear a Metro Ethernet CFM CC information.	**clear ethernet-cfm continuity-check level** *level* **vlan** *vlan.*

This example shows how to clear an Ethernet CFM CC level on a VLAN:

```
Console> (enable) clear ethernet-cfm continuity-check level 3 vlan 1
cc attributes are cleared for level(s) 3
Console> (enable)
```

This example shows how to clear an Ethernet CFM domain:

```
Console> (enable) clear ethernet-cfm domain test level 1
Domain test is cleared from level 1.
Console> (enable)
```

Clearing a Metro Ethernet CFM Maintenance Association

To clear the maintenance association configured within the maintenance domain, perform one of these tasks in priviledged mode:

Task	Command
Clear the CFM configured within a maintenance domain.	**clear ethernet-cfm maintenance-association domain** *domain-name*
Clear the maintenance association name used to construct the Maintenance Association ID (MAID) to be used in CFM frames.	**clear ethernet-cfm maintenance-association ma-name-fmt** *ma_fmt ma-name* **domain** *domain-name*

This example shows how to clear the maintenance association, customerXYA in customerXYADomain:

```
Console> (enable) clear ethernet-cfm maintenance-association ma-name-fmt text customerXYA
domain customerXYADomain
Maintenance Association customerXYA cleared from domain customerXYADomain.
Console> (enable)
```

Clearing a Metro Ethernet CFM Maintenance Point

To clear the Maintenance End Points (MEPs) or Maintenance Intermediate Points (MIPs) for a specific port, perform one of these tasks in priviledged mode:

Task	Command
Clear the CFM configured on a specific module/port.	**clear port ethernet-cfm** *mod/port*
Clear the MEP configuration at a maintenance level and clear the specified VLAN on a specific port. **Note** MEP level values range from 0 to 7. **Note** VLAN values range from 1 to 4094.	**clear port ethernet-cfm** *mod/port* **mep** [**domain** *domain-name* | **vlan** *vlan*]
Clear the MIP configuration.	**clear port ethernet-cfm** *mod/port* **mip** [**level** *level* **vlan** *vlans*]

This example shows how to clear the MEP configuration for module 2, port 1 for a particular domain customerxyz and VLAN 10:

```
Console> (enable) clear port ethernet-cfm 2/1 mep domain customerxyz vlan 10.
MEP config on Port 2/1 is cleared.
Console> (enable)
```

This example shows how to clear the MIP configuration for module 2, port 1:

```
Console> (enable) clear port ethernet-cfm 2/1 mip
MIP config on Port 6/1 is cleared.
Console> (enable)
```

Clearing the MAC Configuration for Maintenance End Points

To clear the port MAC configuration for Maintenance End Points (MEPs) that are down in a particular module and port number of a VLAN, perform one of these tasks in priviledged mode:

Task	Command
Clear the MAC configuration.	**clear ethernet-cfm port-mac-enable**
Clear the MAC configuration for MEPs in a specific module and port number of a VLAN.	**clear ethernet-cfm port-mac-enable** *mNo/pNo* **vlan** *vlans*

This example shows how to clear the port MAC configuration for MEPs that are down in module 3, port 14, and VLAN ID 10:

```
Console> (enable) clear ethernet-cfm port-mac-enable 3/14 vlan 10
```

```
Successfully deleted entries for port(s) 3/14 vlan(s) 10.
Console> (enable)
```

Clearing the Ethernet CFM traceroute Database

To clear the contents of the traceroute database, perform one of these tasks in privileged mode:

Task	Command
Clear the CFM traceroute database information.	**clear ethernet-cfm traceroute-database**
Clear the hold time and the size of the traceroute database.	**clear ethernet-cfm traceroute-database** {**hold-time** | **size**}

This example shows how to clear the contents of the traceroute database:

```
Console> (enable) clear ethernet-cfm traceroute-database
Traceroute database entries cleared.
Console> (enable)
```

This example shows how to clear the hold time of the traceroute database:

```
Console> (enable) clear ethernet-cfm traceroute-database hold-time
Ethernet TRDB Hold time is cleared and set to default.
Console> (enable)
```

This example shows how to clear the size of the traceroute database:

```
Console> (enable) clear ethernet-cfm traceroute-database size
Ethernet TRDB Size cleared and set to default.
Console> (enable)
```

Configuring the Alarm Indication Signal

This section describes how to configure the Alarm Indication Signal (AIS) and the Remote Defect Indication (RDI), which are fault management functions of the Connectivity Fault Management (CFM) protocol. The CFM module works with 802.3ah Link-OAM to support these new extensions.

Note AIS-RDI requires Catalyst 6500 series switch software release 8.7(3) or later.

These sections describe how to configure the AIS:

- Understanding How CFM Works with 802.3ah Link-OAM for AIS-RDI, page 20-55
- Ethernet Alarm Indication Signal, page 20-55
- Ethernet Remote Defect Indication, page 20-56
- ASI and RDI Configuration Guidelines and Restrictions, page 20-56
- Configuring an Alarm Indication Signal, page 20-57
- Ethernet Remote Defect Indication, page 20-56

Understanding How CFM Works with 802.3ah Link-OAM for AIS-RDI

The Ethernet Alarm Indication function (ETH-AIS) and the Ethernet Remote Defect Indication (ETH-RDI) are new functional extensions to Metro Ethernet Connectivity Fault Management (CFM). The ETH-AIS is a standard defined by ITU Y.1731 and the ETH-RDI is part of IEEE 802.1ag. AIS-RDI works together to help reduce the management complexity of large SPAN networks and multiple constituent networks that belong to separate organizations.

Ethernet Alarm Indication Signal

ETH-AIS is an important component of Ethernet-OAM. ETH-AIS is used to suppress alarms after defect conditions are detected at the server (sub) layer.

Note The server (sub) layer is the virtual MEP layer. The IEEE 802.3ah OAM can detect a fault condition.

AIS can differentiate between the faults at the customer level and at the provider level. The AIS serves the following purposes:

1. Notifies the faults from the lower to the upper maintenance domain levels by potentially allowing longer continuity check intervals to be used in the upper levels.
2. Suppresses the multiple redundant alarms by notifying the upper level that the fault detected originates from a lower level.
3. Enables the customer to monitor service availability.

The main functions of the AIS module is as follows:

- To generate the AIS protocol data units (PDUs) upon a signal fault condition that has occurred due to an AIS defect condition.
- To receive to process AIS PDUs, and to maintain an expiry timer.
- To inform the continuity check module about the remote MEP connectivity path failure. The CCM module then generates the Continuity Check Messages (CCMs) with an RDI flag set in the periodic CCMs until the error condition is cleared.
- To signal the configuration management application to suppress alarms.

The CFM works with 802.3ah Finite State Machine (FSM) and has two states:

- SEND_ANY—The 802.3ah OAM link is up and operational. In this state, AIS PDUs are not transmitted.
- AIS—The 802.3ah OAM link has detected traffic that results in a fault condition. The AIS module remains in a sticky state until the OAM link explicitly sends an operational trigger to clear the sticky state. CFM periodically sends AIS PDUs until the defect condition is cleared.

The Server MEP sends the AIS frames with the ETH-AIS information that can be enabled or disabled on a MEP (or on a Server MEP). These frames are issued at the client's maintenance level by a MEP that includes a Server MEP when a defect condition is detected. The defect conditions may include the following:

- The signal fail conditions when an Ethernet Continuity Check (ETH-CC) is enabled.
- The AIS condition when an Ethernet Continuity Check (ETH-CC) is disabled.

Note A Server MEP represents both the server layer termination function and Server/Ethernet adaptation function. In the Cisco IOS software, the Link OAM and Interface/Line Protocol state act as Server MEPs.

Timer Spread Design Logic and Guidelines:

- The AIS transmission interval has been hard coded to 1 second and it can be changed to1 minute dynamically, after the configured number of AIS PDUs are transmitted (default 5).
- The AIS timer logic is designed such that depending upon the global AIS PDU transmission count configured on the switch, you can determine the number of AIS PDUs which will be sent at 1 second interval periodically, when the link failure is detected by the server MEP. The default AIS packet count is set to 5.
- When the defect condition is cleared, the Server MEP sends another set of AIS PDUs (global AIS PDU tx_count configured on the switch) with 1 second interval. So, that the subsequent remote MEPs comes out of the AIS defect condition faster.
- Both the Server MEP and the Local MEP follows the same timer logic to transmit the AIS further in the network.

Ethernet Remote Defect Indication

A MEP uses an ETH-RDI to communicate to its peer MEPs that a defect condition has occurred. A MEP uses an ETH-RDI only when an Ethernet Continuity-Check transmission is enabled.

A MEP that is in a defect condition transmits frames with the ETH-RDI information. When a MEP receives frames with ETH-RDI information, it determines that its peer MEP has encountered a defect condition. However, in a multipoint Ethernet connection, when a MEP receives frames with ETH-RDI information, it cannot determine which peer MEP has a fault condition.

The Ethernet Remote Defect Indication has two management applications:

1. Single-ended fault management—The receiving MEP detects an RDI, which indicates that some of its remote MEPs have failed. When an RDI is not present, it indicates the absence of defects in all the MEPs in a network. This RDI mechanism helps the administrator in the fault management activity on a per service basis.
2. Contribution to far-end performance monitoring—An ETH-RDI indicates that there was a defect condition in the far end of a network. This information is used as an input to the performance monitoring process.

ASI and RDI Configuration Guidelines and Restrictions

When configuring ASI and RDI, follow these configuration guidelines and restrictions:

- You must enable CFM and AIS globally on a switch.
- You must enable CFM on the port before you enable AIS.
- You must enable the Link-OAM on the Server MEP port so that the Link OAM-CFM can function.
- If you explicitly disable CFM globally but the AIS remains enabled, the AIS configuration displays in the configuration. However, the AIS is not functional.

- All the AIS attributes (level, interval, enable/disable, alarm suppression) relate to the MA. The MEP inherits these attributes from the MA. You must create an MA so that you can set any of the AIS parameters.
- You must create a MA for all the VLANs to configure the AIS paramenters for a MA. The AIS configuration is provided for the Server MEP and all of the CFM entities for the local MEPs.
- The software does not support SNMP trap generation to indicate the receipt or transmission of the AIS. A syslog message will be generated to notify the event to the administrator. Because Y.1731 does not define a MIB, this would require either a new MIB, or an extension to that defined by 802.1ag.
- CFM will not generate an SNMP trap for the Server MEP AIS defect condition. Only the syslog messages will be generated to notify the administrator.
- When the CC lifetime expiry occurs for the remote MEP because of a fault in the network, and if the local MEP is already in an AIS condition with alarm-suppression enabled (the default is enable), the trap will be suppressed for that remote MEP. You must explicitly disable alarm suppression for the lifetime expiry trap to be generated.
- For EtherChannel and Server MEP configurations, the AIS is suppressed until the last port of the EtherChannel goes down. The AIS will be generated only when the last port of the EtherChannel leaves the aggregation port. When one of the channel port becomes operational, the AIS condition is cleared.

Configuring an Alarm Indication Signal

This section describes how to configure the Alarm Indication Signal:

Note For complete syntax and usage information for the commands that are used in this section, refer to the *Catalyst 6500 Series Switch Command Reference Software Release 8.x* publication.

Enabling or Disabling a Metro Ethernet CFM Alarm Indication Signal

To enable or disable a CFM AIS globally on a switch, perform this task in privileged mode:

Task	Command
Enable or disable a CFM AIS globally on a switch.	**set ethernet-cfm ais** {**disable** | **enable**}

This example shows how to enable a CFM AIS globally on a switch:

```
Console> (enable) set ethernet-cfm ais disable
Link-Status AIS feature is already disabled on the switch.
Console> (enable)
```

Configuring Continuity-Check Protocol AIS Parameters

To configure the AIS attributes for all MEPs that belong to a specific MA or service, perform this task in privileged mode:

	Task	Command
Step 1	Enable or disable AIS generation to specify the AIS level and VLAN for all MEPs of an MA.	**set ethernet-cfm continuity-check level** *levels* **vlan** *vlans* **ais** {**enable** | **disable**}
Step 2	Set the maintenance level of all MEPs of an MA. Valid values are from 0 to 7.	**set ethernet-cfm continuity-check level** *levels* **vlan** *vlans* **ais level** *level*
Step 3	Enable or disable alarm suppression for all MEPs of an MA.	**set ethernet-cfm continuity-check level** *levels* **vlan** *vlans* **ais alarm-suppress** {**enable** | **disable**}

This example shows how to enable AIS generation for level 0 and VLAN ID 1000:

```
Console> (enable) set ethernet-cfm continuity-check level 0 vlan 1000 ais enable
CC Attributes set for level(s) 0.
Console> (enable)
```

This example shows how to disable AIS generation for level 0 and VLAN ID 1000:

```
Console> (enable) set ethernet-cfm continuity-check level 0 vlan 1000 ais disable
CC Attributes set for level(s) 0.
Console> (enable)
```

This example shows how to enable alarm suppression for level 0 and VLAN ID 1000:

```
Console> (enable) set ethernet-cfm continuity-check level 0 vlan 1000 ais alarm-suppress
enable
CC Attributes set for level(s) 0.
Console> (enable)
```

This example shows how to configure the AIS level for the MEPs:

```
Console> (enable) set ethernet-cfm continuity-check level 5 vlan 5 ais level 6
CC Attributes set for vlan(s) 5 on level 5.
Console> (enable)
```

Configuring the Metro Ethernet CFM Alarm Indication Signal Transmission Level

To configure the CFM AIS transmission level globally on a switch, which will be inherited by all the server MEPs to transmit AIS PDUs when a fault is detected, perform this task in privileged mode:

Task	Command
Configure the CFM AIS transmission level globally on a switch.	**set ethernet-cfm ais level** {*level* | **default**}

This example shows how to configure the CFM AIS level globally on a switch:

```
Console> (enable) set ethernet-cfm ais level 4
Link-Status AIS transmission level configured to 4 on the switch.
Console> (enable)
```

Configuring the Metro Ethernet CFM Alarm Indication Signal PDUs Transmission Count

To configure the CFM Alarm Indication Signal PDUs transmission count on a switch, perform this task in privileged mode:

Task	Command
Configure AIS PDU transmission count globally on a switch. Valid values are from 3 to10. The default is 5.	**set ethernet-cfm ais tx-count** *count*

This example shows how to configure AIS PDUs transmission count globally on a switch:

```
Console> (enable) set ethernet-cfm ais tx-count 10
AIS PDU transmission count set to 10 on the switch.
Console> (enable)
```

Configuring a CFM AIS on an Individual Port

To enable or disable AIS on a port, and to configure an AIS parameter of the port, perform this task in privileged mode:

Task	Command
Enable or disable AIS on a port to specify the AIS server MEP configuration and the AIS generation on a switch port.	**set port ethernet-cfm** *mod/port* **ais** {**enable** | **disable**}

This example shows how to enable a CFM AIS on a port:

```
Console > (enable) set port ethernet-cfm 2/2 ais enable
Server MEP AIS generation is enabled on the port 2/2.
Console > (enable)
```

Displaying CFM AIS/RDI Errors

To display the CFM and AIS/RDI error conditions logged since the last reload, perform this task in privileged mode:

	Task	Command
Step 1	Display the CFM error conditions for maintenance points that have a specific maintenance level.	**show ethernet-cfm errors** [**level** *level*]
Step 1	Display the CFM error conditions for maintenance points and to specify the name of the device domain.	**show ethernet-cfm errors** [**domain** *domain_name*]

This example shows how to display AIS and RDI errors for the local maintenance points:

```
Console> (enable) show ethernet-cfm errors
-------------------------------------------------------------------------------------------
Lvl Vlan MPID Remote-MAC        Reason           MA-Name              Domain-Name
-------------------------------------------------------------------------------------------
0   2010 8190 00-14-f2-31-c1-08 AIS-Errror       vlan2010             dom0
6   2000 8190 00-0b-45-a9-2c-fb RDI-Error        vlan2000             dom6
```

Configuring the Ethernet Local Management Interface

These sections describe how to configure the Ethernet Local Managagement Interface (ELMI):

Understanding How ELMI Works

ELMI is an Ethernet layer operation, administration, and management (OAM) protocol. It provides information that enables autoconfiguration of Customer Edge (CE) devices and provides the status of Ethernet virtual connections (EVCs) for Metro Ethernet Networks (MENs). ELMI notifies a CE device of the operating state of an EVC when an EVC is added or deleted to the interfaces. ELMI also communicates the attributes of an EVC and a user-network interface (UNI) to a CE device.

Ethernet Local Management Protocols

The ELMI protocols are as follows:

- Ethernet Virtual Connections (EVC)—An EVC can be a port level point-to-point or multipoint-to-multipoint Layer 2 circuit. The CE device can use the EVC status to find an alternative path to the service provider network, or in some cases, fall back to a backup path over Ethernet or another alternative service such as Frame Relay or Asynchronous Transfer Mode (ATM).
- Ethernet Local Management Interface (ELMI)—ELMI is an Ethernet layer OAM protocol between a Customer Edge (CE) device and the Provider Edge (PE) in a MEN. It provides information that enables service providers to autoconfigure CE devices with service parameters and parameter changes from a user provider edge (UPE) device.

In a MEN, the EVC status is determined by the OAM protocol. In the Catalyst operating system, ELMI relies on CFM to provide an end-to-end status of the EVC across CFM domains (PE device) in MEN and updates the CE device through ELMI.

Note The Catalyst operating system supports ELMI only in the PE mode.

- User Network Interface (UNI)—UNI is the physical demarcation point between the service provider and the customer. Its attributes, which are similar to the UNI identifier and UNI type, are defined on the PE port that connects to the CE device. The ELMI protocol runs on the UNI interface.

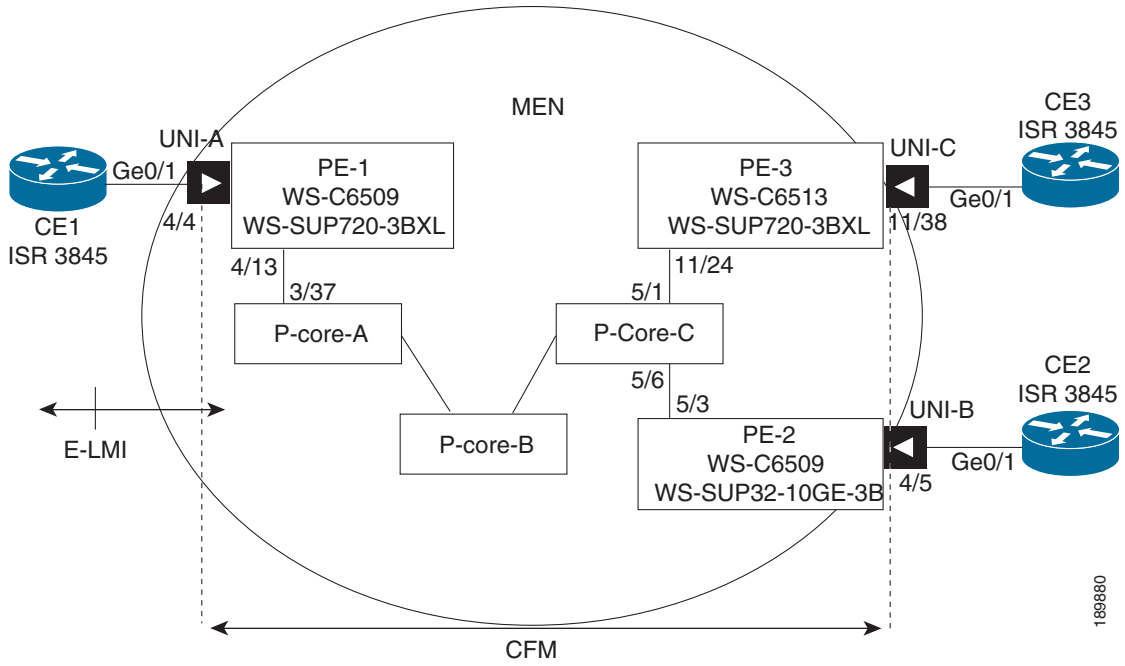
ELMI does the following:

- Notifies the CE when an EVC is added.
- Notifies the CE when an EVC is deleted.
- Notifies the CE of the availability of a configured EVC (Active, Not Active, or Partially Active).
- Communicates UNI and EVC attributes to the CE.

Configuring ELMI

Figure 20-4 shows an example of ELMI that is configured in a multipoint EVC network.

Figure 20-4 ELMI Configured in a Multipoint EVC Network



These guidelines apply to Figure 20-4:

- PE1, PE2, and PE3 are PE switches in the MEN.

Note

- PE1 is a WS-C6509 chassis switch with a WS-SUP720-3BXL as the supervisor engine that runs Catalyst software release 8.7(2).
- PE2 is a WS-C6509 chassis switch with a WS-SUP32-10GE-3B supervisor engine that runs Catalyst software release 8.7(2).
- PE3 is a WS-C6513 chassis switch with a WS-SUP720-3BXL supervisor engine that runs Catalyst software release 8.7(2).

- PE1, PE2, and PE3 switches have VLANs 10 and 250 (switch VLANs) configured as CFM VLANs.
- The ELMI protocol runs between the PE1 switch and CE1-Cisco Internet Switch and Router (ISR) 3845.
- The remote MEPs Continuity Check Database (CCDB) cataloging occurs on all 3 PE switches for VLANs 10 and 250.
- All connected ports are 802.1Q trunk ports that carry the above VLANs.
- Before you can enable and have a working ELMI between PE1-Supervisor Engine 720 WS-C6509 and CE1-ISR 3845, CFM MEPs that are up must exist on edge switch PE1 (port 4/4). Figure 20-4 the CFM MEP that is up also exists on edge switches PE2 and PE3 on ports 11/38 and 4/5, respectively. In the figure, the configuration steps are required so that the PE switch for the ELMI protocol can be enabled and the ELMI frames can be exchanged between the PE1 switch and CE1-ISR3845.

Note You must enable ELMI on the switch globally.

- Enable ELMI on the PE1 port 4/4 that connects to the CE device.
- Multipoint EVCs (EVC 250 and EVC 10 that have a Uni-count of 3 for UNI-A, UNI-B, and UNI-C) are configured on a PE switch that is mapped to VLANs 10 and 250 on which CFM inward MEPs exist. The EVCs are also mapped to CE VLANs 10 and 250.
- The UNI ID and UNI type is configured on the PE edge. Port 4/4 connects to the ISR CE switch. Port 4/4 is a dot1q trunk port and the UNI service bundling type configured on the port is multiplex. The EVCs are mapped onto the PE port 4/4 that connects to the CE1 ISR3845.
- The ELMI frames between PE1 and CE1- ISR3845 are exchanged once you enable ELMI on the CE1-ISR3845 and the ELMI protocol is up between PE1 and CE1-ISR3845.
- The ELMI protocol carries the following information in ELMI frames from the PE to the CE using the ELMI status messages:
  - Notification to the CE about the status of an EVC.
  - Communication of UNI and EVC attributes to the CE.

Configuring ELMI on the Switch

This section describes how to configure ELMI:

- Enabling or Disabling ELMI, page 20-63
- Enabling or Disabling an EVC, page 20-63
- Configuring ELMI on an Individual Port, page 20-64

Note For complete syntax and usage information for the commands that are used in this section, refer to the *Catalyst 6500 Series Switch Command Reference Software Release 8.x* publication.

Enabling or Disabling ELMI

To enable or disable the ELMI globally on a switch, perform this task in privileged mode:

Task	Command
Enable or disable ELMI globally on a switch.	**set ethernet-lmi** {**enable** | **disable**}

This example shows how to enable ELMI globally on a switch:

```
Console> (enable) set ethernet-lmi enable
Ethernet-LMI is enabled.
Console> (enable)
```

Enabling or Disabling an EVC

To create an Ethernet Virtual Connection (EVC) in global configuration mode and configure various parameters associated with the EVC on a switch, perform this task in privileged mode:

.

Task	Command
Enable EVC globally and configure various parameters, such as the EVC identifier and specify the number of endpoints (UNIs), a multipoint service (uni-count of 3), the CFM maintenance domain name, the MA-name format, the maintenance association, and associated CE-VLAN.	**set ethernet-evc** *evc-id* **uni-count** *count* [**multipoint**] **domain** *name* **ma-name-fmt** *fmt ma-name* **ce-vlan** ***any*** | *vlan*

Note By default, an EVC with uni-count 2 is a point-to-point EVC.

These examples show how to configure various EVC parameters:

```
Console>(enable) set ethernet-evc EVC1 uni-count 2
UNI count for EVC1 is configured as 2.

Console> (enable) set ethernet-evc EVC1 domain ELMI ma-name-fmt text CFM1
Successfully create EVC EVC1 and CFM service name CFM1.

Console>(enable) set ethernet-evc EVC1 ce-vlan 10
CE-Vlan 10 is successfully mapped to EVC1.
Console > (enable)
```

Configuring ELMI on an Individual Port

To enable or disable ELMI processing on the port, perform one of these tasks in privileged mode:

Task	Command
Enable or disable ELMI on a switch port.	**set port ethernet-lmi** {*mod/port*} {**enable** | **disable**}
Specify the timer value and polling timer to transmit the status query. **Note** **t391** is part of the CE configuration; the Catalyst operating system supports only PE mode. **Note** The polling timer range is from 5 to 30 seconds.	**set port ethernet-lmi** {*mod/port*} **t391** {*value* | **default** | **disable**}
Specify the polling verification timer to verify the status query that is sent by the CE device and to which the PE responds with status messages. **Note** **t392** should be greater than t391**.** **Note** The polling verification timer range is from 5 to 30 seconds.	**set port ethernet-lmi** {*mod/port*} **t392** {*value* | **default** | **disable**}
Specify the polling counter that gives a full status of the User to Network Interface (UNI) and all the EVC polling counts. **Note** **n391** applies only to the CE. **Note** The EVC polling counts range is from 1 to 65000.	**set port ethernet-lmi** {*mod/port*} **n391** {*value* | **default**}
Specify the event counter that gives a count of monitored events. **Note** **n393** applies to the CE and PE. **Note** The event counter range is from 1 to 10.	**set port ethernet-lmi** {*mod/port*} **n393** {*value* | **default**}

These examples show how to set the ELMI port:

```
Console>(enable) set port ethernet-lmi 3/1 enable
Ethernet LMI is enabled on port 3/1.

Console>(enable) set port ethernet-lmi 3/1 t392 30
Ethernet LMI polling verification timer is set to 30 seconds for port 3/1.
Console>(enable)
```

Configuring a UNI ID on an Individual Port

To set the UNI ID for a particular port, perform this task in privileged mode:

Task	Command
Configure a UNI ID on a specific module and port.	**set port ethernet-uni** {*mod/port*} **id** {*uni-id*}

This example shows how to set the Ethernet UNI ID as CUST_A_PORT1 for module 3, port 1:

```
Console> (enable) set port ethernet-uni 3/1 id CUST_A_PORT1
UNI id CUST_A_PORT1 is configured on port 3/1
Console> (enable)
```

Configuring UNI-TYPE on an Individual Port

To configure the UNI-TYPE for a particular port, perform this task in privileged mode:

Task	Command
Configure a UNI-TYPE for a specific module and port.	**set port ethernet-uni** {*mod/port*} **type** {**all-to-one** | **multiplex**}

This example shows how to to set the UNI TYPE as all-to-one for module 5 and port 1:

```
Console> (enable) set port ethernet-uni 5/1 type all-to-one
Uni type on port 5/1 successfully set to all-to-one.
```

This example shows how to to set the UNI TYPE as multiplex for module 5 and port 1:

```
Console> (enable) set port ethernet-uni 5/1 type multiplex
Uni type on port 5/1 successfully set to multiplex.
```

Configuring an EVC on an Individual Port

To associate an EVC to a port and the corresponding CE-VLANs, perform this task in privileged mode:

Task	Command
Enable or disable an EVC on a particular module and port and associate the EVC identifier.	**set port ethernet-evc** *mod/port* [*evc_id*]

This example shows how to set the Ethernet EVC ID as EVC1 for module 7, port 1:

```
Console> (enable) set port ethernet-evc 7/1 EVC1
EVC1 is associated to port 7/1.
Console> (enable
```

Displaying an EVC

To display the EVCs configured on a device, perform this task in privileged mode:

Task	Command
Display EVCs configured on a device.	**show ethernet-evc** {[**detail**] | *evc_id* [**detail**]}

These examples show how to display EVCs configured on the device:

```
Console> (enable) show ethernet-evc
St EVC Id CE-Vlan
--- ---------- ------------------------------
A EVC1 10
A EVC2 20

Key: St=Status, A=Active, P=Partially Active, I=Inactive, ?=ELMI Link Down

Console> (enable) show ethernet-evc detail
EVC Id: EVC1
EVC Type: P-P
EVC Status: Active
EVC Uni Count: 2
Number of Remote UNIs up: 1
Number of Local UNIs up: 1
CFM Service Maintenance Domain: ELMI
CFM Service Maintenance Name: CFM1
EVC CE-Vlan Mapping: 10
Ports associated to this EVC: 7/1
Remote UNI Details:
UNI Id UNI Status Port
-------------- ------------ -----------
SANFRANCISCO Up 4/47

EVC Id: EVC2
EVC Type: P-P
EVC Status: Inactive
EVC Uni Count: 2
Number of Remote UNIs up: 0
Number of Local UNIs up: 1
CFM Service Maintenance Domain: SJC
CFM Service Maintenance Name: CFM2
EVC CE-Vlan Mapping: 20
Ports associated to this EVC: 7/1
```

Displaying CE-VLAN/ EVC

To display the CE-VLAN/EVC mapping configured for the port, perform this task in privileged mode:

Task	Command
Display the CE-VLAN/EVC mapping.	**show port ethernet-evc** *mod/port* {[**detail**] | *evc-id* [**detail**]}

These examples show how to display the CE-VLAN/EVC mapping configured for module 7, port 1:

```
Console>(enable) show port ethernet-evc 7/1
UNI Id: PE-CUSTA-PORT1
St EVC Id CE-Vlan
------ -------------- ----------
?A EVC1 10
?A EVC2 20
Key: St=Status, A=Active, P=Partially Active, I=Inactive, ?=ELMI Link Down

Console> (enable) show port ethernet-evc 7/1 EVC1 detail
Port: 7/1
EVC Id: EVC1
Time since Last Full Report: Never
Ether LMI Link Status: Down
UNI Id: SANJOSE
UNI Status: Up
CE-VLAN/EVC Map Type: multiplex
CE-VLAN: 10
EVC Status: Inactive
EVC Type: Point-to-Point
Remote UNI Count: Configured = 1, Active = 0
```

Displaying ELMI Statistics and Configuration

To display ELMI statistics and ELMI parameters, perform one of these tasks in privileged mode:

Task	Command
Display ELMI statistics.	**show port ethernet-lmi** *mod/port* **statistics**
Display the ELMI configuration.	**show port ethernet-lmi** *mod/port* **config**

This example shows how to display the ELMI statistics and configuration for module 7, port 1:

```
Console> (enable) show port ethernet-lmi 7/1 statistics
E-LMI statistics for port 7/1
Ethernet LMI Link Status: Up
UNI Status: Up
UNI Id: PE1-CustA-Port1
Reliability Errors:
Status Enq Timeouts 0 Invalid Sequence Number 0
Protocol Errors:
Invalid Protocol Version 0 Invalid EVC Reference Id 0
Invalid Message Type 0 Out of sequence IE 0
Duplicated IE 0 Mandatory IE missing 0
Invalid Mandatory IE 0 Invalid non-mandatory IE 0
Unrecognized IE 0 Unexpected IE 0
Last Full Status Enq Rcvd 00:00:10 Last Full Status Sent 00:00:10
Last Status Check Enq Rcvd 00:00:00 Last Status Check Sent 00:00:00
Last clearing of counters never
Console> (enable) show port ethernet-lmi 7/1 config
```

```
E-LMI parameters for port 7/1
Port Ethernet LMI: Enabled
Operational Status: Disabled
Mode: PE
T391: NA
T392: 15
N391: NA
N393: 4
Console <enable>
```

Clearing an EVC

To clear an EVC configured in the switch, perform this task in privileged mode:

Task	Command
Clear an EVC configured in the switch.	**clear ethernet-evc** [*evc_id*]

This example shows how to clear EVC1:

```
Console> (enable) clear ethernet-evc EVC1
EVC1 is successfully cleared.
Console> (enable)
```

Clearing an EVC on an Individual Port Associated to a UNI

To clear any EVCs associated to the UNI or a specified EVC, perform this task in privileged mode:

Task	Command
Clear an EVC configured in the switch port.	**clear port ethernet-lmi** *mod/port***statistics**

This example shows how to clear EVCs associated with module 7, port 1:

```
Console> (enable) clear port ethernet-evc 7/1
EVCs associated with port 7/1 are cleared.
Console> (enable)
```

Clearing ELMI Statistics Counters

To clear ELMI statistics counters for all ports or a specified port, perform this task in priviledged mode:

Task	Command
Clear ELMI statistics counters.	**clear port ethernet-evc** *mod/port* [*evc_id*]

This example shows how to clear ELMI statistics associated with module 7, port 1:

```
Console> (enable) clear port ethernet-lmi 7/1 statistics
Ethernet LMI statistics cleared on port 7/1.
Console> (enable)
```

Clearing a UNI Configuration

To clear the UNI configuration on the port, perform this task in priviledged mode:

Task	Command
Clear a UNI configuration on the port.	**clear port ethernet-evc** *mod/port* [*id* | *type*]

This example shows how to clear the UNI configuration on module 7, port 1:

```
Console> (enable) clear port ethernet-uni 7/1
UNI configuration is cleared for port 7/1.
Console> (enable)
```

Configuring MAC Address Move Counters

These sections describe the MAC address move counters:

Understanding How MAC Address Move Counters Work

The MAC address move counters feature provides a counter that increments each time that an existing MAC address moves from a given port to another port in the same VLAN. If you see the same MAC address on another port, this situation can indicate a problem in the network (such as a spanning-tree loop, HSRP flapping, or a server link flapping). However, this situation does not always indicate a problem. The following events can result in the same MAC address that is seen on another port but are considered normal behavior and are not indications of a problem:

- A laptop PC is moved throughout a VLAN domain as the laptop PC is moved from port to port.
- A laptop PC with multiple connections to the VLAN is moved through a physical port or wireless connection.
- A server with dual NICs is moved in two separate VLANs.

Before the MAC address move counters feature was introduced, the existing MAC move notification feature generated syslogs for each MAC address move. The two main drawbacks to the existing feature were as follows:

- When there are a large number of MAC moves, the number of generated syslogs can be overwhelming.
- The feature does not provide a convenient means of displaying the MAC addresses that have moved for future examination.

MAC Address Move Counter Configuration Guidelines and Restrictions

When configuring MAC address move counters, follow these configuration guidelines and restrictions :

- Layer 2 AISCs learn any new MAC addresses and associate them with a port. Only dynamic CAM entries are learned.
- MAC address moves are defined when MAC addresses move from a given port to another port in the same VLAN.
- The counter increments each time that an existing MAC address moves from a given port to another port in the same VLAN.
- For private VLANs, MAC address moves are defined as MAC addresses that move from a given port to another port in different secondary VLANs but in the same primary VLAN.
- The MAC address move counters feature coexists with the existing MAC address move feature when you enter the **set cam notification move** {**enable** | **disable**} command.
- The feature allows you to store a maximum of 1000 MAC address move counter tuples per VLAN. When the maximum limit of 1000 tuples is exceeded, new moves that occur in that VLAN are not recorded.
- For proper syslog generation, you need to set the logging level for the EARL facility to 4 or higher by entering the **set logging level earl** *severity* command.

MAC Address Move Counter syslog Generation

The MAC address move counters generate the syslogs that are described in these sections:

- Detecting MAC Address Moves, page 20-70
- Exceeding the Maximum Limit for MAC Address Move Counters for a VLAN, page 20-71

Detecting MAC Address Moves

Table 20-4 describes the scenarios that cause the "%EARL-4-MAC_MOVE_COUNTER:Mac move(s) detected" syslog to be generated.

Table 20-4 MAC Address Move Counter Syslog Generation

Scenario	Are MAC Address Move Counter Syslogs Generated?
MAC address move counters have been enabled for the first time and one or more MAC address moves occurred since the feature was enabled.	Yes
MAC address move counters have been disabled and then enabled, and one or more MAC address moves occurred since the feature was enabled.	Yes
MAC address move counter entries have been cleared for all VLANs by entering the **clear cam notification move counters all** command, and one or more MAC address moves occurred after the entries were cleared.	Yes

Table 20-4 MAC Address Move Counter Syslog Generation (continued)

Scenario	Are MAC Address Move Counter Syslogs Generated?
MAC address move counter entries have been cleared for a specified VLAN by entering the **clear cam notification move counters** *vlan* command, and one or more MAC address moves occurred after entries were cleared.	No
MAC address move counters have been disabled and MAC address moves are occurring.	No

Exceeding the Maximum Limit for MAC Address Move Counters for a VLAN

The following syslog is generated when the maximum limit of 1000 MAC address move counter tuples per VLAN is exceeded: "%EARL-4-MAC_MOVE_COUNTER_COUNT_EXCEEDED: Maximum limit for MAC move counters exceeded for Vlan vlan."

Executing MAC Address Move Counters

These sections describe how to execute MAC address move counters:

- Enabling or Disabling MAC Address Move Counters, page 20-71
- Displaying MAC Address Move Counter Statistics, page 20-72
- Clearing MAC Address Move Counter Statistics, page 20-73

Enabling or Disabling MAC Address Move Counters

To enable or disable MAC address move counters, perform this task in privileged mode:

Task	Command
Enable or disable MAC address move counters.	**set cam notification move counters** {**disable** | **enable**}

This example shows how to enable MAC address move counters:

```
Console> (enable) set cam notification move counters enable
MAC move counters are enabled

Please change the logging level for the Earl facility, as the current logging
level is set to 2 and Mac Move Counters requires a logging level of at least 4.
Console> (enable)
```

This example shows that the logging level for the EARL facility needs to be set to 4 or higher as follows:

```
Console> (enable) set logging level earl 4
System logging facility <earl> for this session set to severity 4(warnings)
Console> (enable)
```

This example shows how to disable MAC address move counters:

```
Console> (enable) set cam notification move counters disable
```

```
MAC move counters are disabled
Console> (enable)
```

Displaying MAC Address Move Counter Statistics

To display MAC address move counter statistics, perform this task in normal mode:

Task	Command
Display MAC address move counter statistics.	**show cam notification move counters** [*vlan*]

This example shows how to display MAC address move counter statistics for all VLANs:

```
Console> (enable) show cam notification move counters
-------------------------------------------------------------------------------
Vlan    Mac Address         From Mod/Port           To Mod/Port           Count
---- ----------------- ---------------------- ---------------------- ------------
  1  00-01-02-04-04-01                   2/3                    3/1           10
 200 00-01-05-03-02-01                   5/3                    5/1           20
Console> (enable)
```

This example shows how to display MAC address move counter statistics for the specified VLAN:

```
Console> (enable) show cam notification move counters 1
-------------------------------------------------------------------------------
Vlan    Mac Address         From Mod/Port           To Mod/Port           Count
---- ----------------- ---------------------- ---------------------- ------------
   1 00-01-02-04-04-01                   2/3                    3/1           15
Console> (enable)
```

This example shows how to display MAC address move counter statistics where the To Mod/Port field is part of an EtherChannel:

```
Console> (enable) show cam notification move counters
-------------------------------------------------------------------------------
Vlan    Mac Address         From Mod/Port           To Mod/Port           Count
---- ----------------- ---------------------- ---------------------- ------------
   1 00-01-02-07-08-01                   3/1         2/1,2/3,2/5,2/7           10
Console> (enable)
```

This example shows how to display MAC address move counter statistics where the From Mod/Port field is part of an EtherChannel:

```
Console> (enable) show cam notification move counters
-------------------------------------------------------------------------------
Vlan    Mac Address         From Mod/Port           To Mod/Port           Count
---- ----------------- ---------------------- ---------------------- ------------
   1 0-01-02-07-03-0A          2/1,2/3,2/5,2/7                    3/1           20
Console> (enable)
```

This example shows how to display MAC address move counter statistics where the To Mod/Port field and the From Mod/Port field are part of an EtherChannel:

```
Console> (enable) show cam notification move counters
-------------------------------------------------------------------------------
Vlan    Mac Address         From Mod/Port           To Mod/Port           Count
---- ----------------- ---------------------- ---------------------- ------------
   1 00-01-02-06-08-01         3/1,3/3,3/5,3/7         2/1,2/3,2/5,2/7           15
Console> (enable)
```

Clearing MAC Address Move Counter Statistics

To clear MAC address move counter statistics, perform this task in privileged mode:

Task	Command
Clear MAC address move counter statistics.	**clear cam notification move counters** {**all** | *vlan*}

This example shows how to clear MAC address move counter statistics for all VLANs:

```
Console> (enable) clear cam notification move counters all
This will clear the mac move counters for all Vlans.
Do you want to continue (y/n) [n]? y
MAC move counters for all Vlans cleared
Console> (enable)
```

This example shows how to clear MAC address move counter statistics for the specified VLAN:

```
Console> (enable) clear cam notification move counters 1
This will clear the mac move counters for Vlan 1.
Do you want to continue (y/n) [n]? y
MAC move counters for Vlan 1 cleared
Console> (enable)
```

Digital Optical Monitoring

The Diagnostic Optical Monitoring (DOM) feature provides real-time access for optical transceivers to operating parameters such as temperature, voltage, laser bias current, and receive/transmit optical power.

Note Xenpak transceivers do not support the voltage parameter. For Xenpak transceivers, voltage will be displayed as “n/a.”

To display the default values provided for the transceivers, use the **show transceivers threshold-table** command. You can overwrite the threshold values by using **per-port set** commands.

Note Bias current is a parameter that is unique to each transceiver, and it cannot be changed by using the **per-port set** commands.

Displaying Transceiver Information

The following sections describe how to display transceiver information:

- Displaying General Port Transceiver Information, page 20-74
- Displaying Detailed Transceiver Information, page 20-74
- Displaying Transceiver Threshold Violations, page 20-75
- Displaying Port Transceiver Information, page 20-75
- Displaying Port Transceiver Configuration Information, page 20-76

Displaying General Port Transceiver Information

To display general port transceiver information, perform this task in enabled mode:

Task	Command
Display general port transceiver information.	**show port transceiver**

This example shows how to display general port transceiver information:

```
Console> show port transceiver
Transceiver monitoring is disabled for all ports.
Monitor interval is set to 10 minutes.

If device is externally calibrated, only calibrated values are printed.
++ : high alarm, +  : high warning, -  : low warning, -- : low alarm.
NA or N/A: not applicable, Tx: transmit, Rx: receive.
mA: milliamperes, dBm: decibels (milliwatts).

                                                                 Optical   Optical
         Temperature  Voltage  Current   Tx Power  Rx Power
Port    (Celsius)       (Volts)   (mA)      (dBm)       (dBm)
-----    --------------   ---------   --------    ------------  -----------
 3/1     34.6             0.00        29.3         -1.7           -2.1
 3/2     32.9             0.00        30.5         -1.8           -2.3
```

Displaying Detailed Transceiver Information

To display detailed transceiver information, perform this task in enabled mode:

Task	Command
Display detailed transceiver information.	**show port transceiver detail**

This example shows how to display detailed transceiver information:

```
Console> show port transceiver detail
Transceiver monitoring is disabled for all ports.
Monitor interval is set to 10 minutes.

mA: milliamperes, dBm: decibels (milliwatts), NA or N/A: not applicable.
## : high alarm, #  : high warning, @  : low warning, @@ : low alarm.
A2D readouts (if they differ), are reported in parentheses.
The threshold values are calibrated.

                                  High Alarm  High Warn  Low Warn   Low Alarm
         Temperature      Threshold     Threshold   Threshold    Threshold
Port    (Celsius)           (Celsius)       (Celsius)     (Celsius)     (Celsius)
-----    -----------------  --------------  -------------  --------------  ---------
 3/1     34.5                  70.0             70.0           0.0              0.0
 3/2     32.9                  70.0             70.0           0.0              0.0

                                 High Alarm  High Warn  Low Warn   Low Alarm
         Voltage             Threshold     Threshold   Threshold    Threshold
Port    (Volts)             (Volts)          (Volts)        (Volts)         (Volts)
-----   ---------------    ---------------  -------------  -------------   ---------------
 3/1    0.00                  5.24             5.24           5.24             5.24
 3/2    0.00                  5.24             5.24           5.24             5.24
```

```
                                    High Alarm  High Warn  Low Warn   Low Alarm
         Current                   Threshold     Threshold   Threshold    Threshold
Port   (milliamperes)  (mA)           (mA)           (mA)           (mA)
-----  -----------------  -------------  -------------  -------------  -------------
 3/1    29.3                 2.5              2.5              2.5             2.5
 3/2    30.4                 2.5              2.5              2.5             2.5

         Optical              High Alarm  High Warn  Low Warn   Low Alarm
         Transmit Power Threshold    Threshold    Threshold    Threshold
Port   (dBm)               (dBm)          (dBm)          (dBm)          (dBm)
-----  -------------------  -------------  -------------  -------------  -------------
 3/1    -1.7                  1.0              0.0              -7.2            -8.2
 3/2    -1.8                  1.0              0.0              -7.2            -8.2

         Optical              High Alarm  High Warn  Low Warn   Low Alarm
         Receive Power  Threshold     Threshold   Threshold    Threshold
Port   (dBm)               (dBm)          (dBm)          (dBm)          (dBm)
-----  -----------------   -------------  -------------  -------------  -------------
 3/1    -2.1                  1.0              0.0              -14.1          -16.4
 3/2    -2.3                  1.0              0.0              -14.1          -16.4
```

Displaying Transceiver Threshold Violations

To display transceiver threshold violations, perform this task in enabled mode:

Task	Command
Display transceiver threshold violations.	**show port transceiver** *slot number* **threshold-violations**

This example shows how to display transceiver threshold violations:

```
Console> show port transceiver 3 threshold-violations
Transceiver monitoring is enabled for all ports.
Monitor interval is set to 5 minutes.

Rx: Receive, Tx: Transmit.
DDDD: days, HH: hours, MM: minutes, SS: seconds
Time since Last Known
         Time in slot     Threshold Violation        Type(s) of Last Known
Port   (DDDD:HH:MM:SS)     (DDDD:HH:MM:SS)          Threshold Violation(s)
-----  -------------------------  -------------------------------  ------------
 3/1   0000:06:39:07             0000:00:03:57                 Tx bias high alarm
                                                                                5.8
mA >   0.5 mA
 3/2   0000:06:39:07             0000:00:03:56                 Tx bias high alarm
                                                                                6.0
mA >   0.5 mA
```

Displaying Port Transceiver Information

To display port transceiver information, perform this task in enabled mode:

Task	Command
Display port transceiver information.	**show port transceiver** *mod/port*

This example shows how to display port transceiver information:

```
Console> show port transceiver 2/1
sh port transceiver 5/1
Transceiver monitoring is enabled.
Monitor interval is set to 1 minute

ITU Channel not available (1550 nm)
## : high alarm, #  : high warning, @  : low warning, @@ : low alarm.
NA or N/A: not applicable, Tx: transmit, Rx: receive.
mA: milliamperes, dBm: decibels (milliwatts).

                                                 Optical       Optical
       Temperature  Voltage    Current           Tx Power      Rx Power
Port   (Celsius)    (Volts)    (mA)              (dBm)         (dBm)
-----  -----------  ---------  ------------  ------------  ------------
 5/1          29.2        N/A         102.5           0.9  @@     -31.0
```

Displaying Port Transceiver Configuration Information

To display port transceiver configuration information, perform this task in enabled mode:

Task	Command
Display port transceiver configuration information.	**show port transceiver** *mod/port* **config**

This example shows how to display port transceiver configuration information:

```
Console> show port transceiver 3/1 config
Transceiver monitoring is disabled.
Monitor interval is set to 1 minute.

                                Transmit Power (dBm)
         High Alarm        High Warn           Low Warn           Low Alarm
         Threshold           Threshold            Threshold            Threshold
Port   Value   Severity  Value   Severity  Value   Severity  Value   Severity
-----    -------  ----------   -------  ----------   -------   ----------  -------   -----
 3/1    default   critical   default   critical   default    critical  default   critical

                                Receiver Power (dBm)
         High Alarm         High Warn          Low Warn           Low Alarm
         Threshold           Threshold            Threshold            Threshold
Port   Value   Severity  Value   Severity  Value   Severity  Value   Severity
-----    -------  ----------   -------   ----------  -------   ----------   -------  -----
 3/1   default    critical   default   critical   default    critical  default   critical

                                Temperature (Celsius)
         High Alarm         High Warn          Low Warn           Low Alarm
         Threshold           Threshold            Threshold            Threshold
Port   Value   Severity  Value   Severity  Value   Severity  Value   Severity
-----    -------  ----------   -------   ----------  -------   ----------  -------  ------
 3/1    default   critical   default    critical  default   critical  default    critical

                                Voltage (volts)
         High Alarm         High Warn          Low Warn           Low Alarm
         Threshold           Threshold            Threshold            Threshold
Port   Value   Severity  Value   Severity  Value   Severity  Value   Severity
-----    -------  ----------   -------   ----------  -------   ----------  -------   -----
 3/1   default    critical  default    critical  default    critical  default    critical
```

Setting Transceiver Monitoring and Thresholds

The following sections describe how to set transceiver monitoring parameters and thresholds:

Enabling or Disabling Transceiver Monitoring

To enable or disable transceiver monitoring, perform this task in enabled mode:

Task	Command
Enable transceiver monitoring.	**set transceiver-monitoring [enable | disable]**

This example shows how to enable transceiver monitoring:

```
Console> (enable) set transceiver-monitoring enable
Transceiver monitoring is successfully enabled
```

This example shows how to disable transceiver monitoring:

```
Console> (enable) set transceiver-monitoring disable
Transceiver monitoring is successfully disabled
```

Setting the Transceiver Monitoring Interval

To set the transceiver monitoring interval, perform this task in enabled mode:

Task	Command
Set the transceiver monitoring interval.	**set transceiver-monitoring interval** *minutes*

This example shows how to set the transceiver monitoring interval:

```
Console> (enable) set transceiver-monitoring interval 10
Transceiver monitoring interval is set to 10 minutes
```

Setting the Transceiver Temperature Threshold

To set a transceiver temperature threshold, perform this task in enabled mode:

Task	Command
Set a transceiver temperature threshold.	**set port transceiver** *mod/port* [**current | rx-power | temperature | tx-power | voltage**] [**high-alarm | high-warn | low-alarm | low-warn**] **severity** *severity value*

This example shows how to set a transceiver temperature threshold without specifying the severity:

```
Console> (enable) set port transceiver 3/1 temperature high-alarm threshold 750
Optical temperature high-alarm threshold is set to 75.0 celsius for port 3/1
```

This example shows how to set a transceiver temperature threshold including the severity:

```
Console> (enable) set port transceiver 3/1 temperature high-alarm threshold 75 severity
critical
Optical temperature high-alarm threshold is set to 75.0 celsius  for port 3/1 and severity
is set to critical
```

CHAPTER 21

Configuring GOLD

This chapter describes how to configure generic online diagnostics (GOLD) on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Understanding How Online Diagnostics Work

Note GOLD is supported on the Supervisor Engine 720 and Supervisor Engine 32 only. However, earlier diagnostic commands are still supported on the Supervisor Engine 1 and Supervisor Engine 2.

Online diagnostics performs the following functions:

- Test and verify the hardware functionality of the supervisor engine, modules, and switch while the switch is connected to a live network.
- Perform packet switching tests that check different hardware components and verify the data path and control signals.
- Detect problems in the following areas:
  - Hardware components
  - Interfaces (GBICs, Ethernet ports, and so forth)
  - Connectors (loose connectors, bent pins, and so forth)
  - Solder joints
  - Memory (failure over time)

Online diagnostics are categorized as follows:

- Bootup—Bootup diagnostics run during bootup, module OIR, or switchover to a backup supervisor engine.
- On-demand—On-demand diagnostics run from the CLI.
- Schedule—Schedule diagnostics run at user-designated intervals or specified times when the switch is connected to a live network.
- Health monitoring—Health-monitoring diagnostics run in the background.

There are two types of online diagnostic tests:

- Disruptive online diagnostic tests—These tests include the built-in self-test (BIST) and the disruptive loopback test.
- Nondisruptive online diagnostic tests—These tests include packet switching, and run during bootup, line card online insertion and removal (OIR), and system reset diagnostic tests. The nondisruptive online diagnostic tests run as part of the background health monitoring or at your request (on-demand).

Online diagnostics is one of the requirements for the high-availability feature. High availability is a set of quality standards that seek to limit the impact of equipment failures on the network. A key part of high availability is detecting hardware failures and taking corrective action while the switch runs in a live network. Online diagnostics in high availability detect hardware failures and provide feedback to high-availability software components to make switchover decisions.

Configuring Online Diagnostics

These sections describe how to configure online diagnostics:

Specifying the Bootup Online Diagnostic Level

You can specify the bootup online diagnostics level as minimal or complete, or you can bypass the bootup diagnostics entirely. Enter the **complete** keyword to run all diagnostic tests; enter the **minimal** keyword to run only PFC tests for the supervisor engine and loopback tests for all ports in the switch; enter the **bypass** keyword to bypass all diagnostic tests. The default bootup diagnostics level is **minimal**.

Note Although the default is **minimal**, we recommend that you set the bootup online diagnostics level to complete. We do not recommend bypassing the bootup online diagnostics.

Note The bootup diagnostic level applies to the entire switch and cannot be configured on a per-module basis.

To specify the bootup diagnostic level, perform this task in privileged mode:

	Task	Command
Step 1	Specify the bootup diagnostic level.	**set diagnostic bootup level** [**bypass** | **minimal** | **complete**]
Step 2	Display the bootup diagnostic level.	**show diagnostic bootup level**

This example shows how to specify bypass as the bootup diagnostic level:

```
Console> (enable) set diagnostic bootup level complete
Diagnostic level set to complete
Console> (enable)

Console> (enable) show diagnostic bootup level
Current bootup diagnostic level: complete
Console> (enable)
```

Configuring On-Demand Online Diagnostics

Caution Most of the online diagnostic memory tests are on-demand tests because of their disruptive nature and time duration. You should use the memory tests only when you suspect a problem in the hardware and only after you have isolated the system from the live production network environment.

Note Online diagnostic tests use the EOBC channel to communicate with the rest of the system. Proper working of the EOBC channel between the supervisor engine and the SLCP, LCP, and the module processors is required for performing the online diagnostic tests.

Use the information in these sections for configuring on-demand online diagnostics:

- Running On-Demand Online Diagnostic Tests, page 21-3
- On-Demand Online Diagnostic Configuration Guidelines and Restrictions, page 21-4
- On-Demand Online Diagnostic Configuration Procedure, page 21-4
- Configuring Diagnostics Operations, page 21-8

Running On-Demand Online Diagnostic Tests

Caution Use this section to familiarize yourself with the on-demand **diagnostic start** and **diagnostic stop** commands. To run any of the on-demand online diagnostic tests, use the procedure in the "On-Demand Online Diagnostic Configuration Procedure" section on page 21-4. Do not attempt to run these tests without following the on-demand online diagnostics configuration procedure.

Use the **diagnostic start** command to start running specific test(s) based on the test IDs. The command accepts one test ID, a range of test IDs, a subgroup of tests, or **all** for all tests. The test ID for a particular test can be different from one module type to another module type or even from one software release to another software release. It is important that you obtain the correct test ID and relevant test name using the **show diagnostic content** command. Use the **diagnostic stop module** *mod* command to stop running tests on the specified module. The complete syntax for the **diagnostic start** and **diagnostic stop** commands is as follows:

diagnostic start module *mod_num* **test** {**all** | *test_ID_num* | *test_list* | **complete** | **minimal** | **non-disruptive** | **per-port**} [**port** {**all** | *port_num* | *port_list*}]

diagnostic stop module *mod*

On-Demand Online Diagnostic Configuration Guidelines and Restrictions

This section describes the configuration guidelines and restrictions for performing the on-demand test configuration steps described in the "On-Demand Online Diagnostic Configuration Procedure" section on page 21-4:

- After running tests in a particular step, the tests in earlier steps may not work.
- You may need to perform certain actions before and after running a test. These actions are described in the configuration procedure.
- Some of the tests are disruptive. The configuration procedure provides guidance for running any disruptive tests.
- You should run packet-switching tests before you run memory tests.
- Memory tests should always be run on modules first and then on the supervisor engine because after running the memory tests on the supervisor engine, the system is in an unusable state and needs to be rebooted immediately for normal operation.

Note With software release 8.5(1), memory tests are available only for the supervisor engine. Memory tests for other modules are planned for subsequent releases.

On-Demand Online Diagnostic Configuration Procedure

To run on-demand online diagnostic tests, perform these steps:

Step 1 Run the nondisruptive tests. Nondisruptive tests are packet-switching tests and do not disrupt the system operation in any way. These tests take only a few seconds to finish.

Additional test requirements are as follows:

- User actions before running the test—None
- User actions after running the test—None

Step 2 The packet-switching tests fall into various functional test groups. Use the following tables to determine which functional test group you want to test and then run tests in that functional test group:

- Table 21-1, On-Demand Tests: Supervisor Engine
- Table 21-2, On-Demand Tests: Fabric-Enabled Modules
- Table 21-3, On-Demand Tests: Non-Fabric-Enabled Modules

Note Not all functional test groups are present for every module because the supported functional test groups vary depending on the module type. If you are not sure which functional test group to select, run all the packet switching tests that are run during bootup when the diagnostic level is set to "complete" by entering the **diagnostic start module** *mod/num* **test complete** command.

Note If you run the loopback test and it fails on one or several ports of a module, disconnect any cables that are connected to the ports on that module, shut down all the ports on that module, and then rerun the loopback test. It is possible that some spurious packets are interfering with the loopback test and causing it to fail. Also, if the module has an inline-power daughter card, disable power to the inline-power daughter card before running the test.

Additional test requirements are as follows:

- User actions before running the test—None
- User actions after running the test—None

Table 21-1 On-Demand Tests: Supervisor Engine

Functional Test Group	Individual Tests
Per-port tests	TestLoopback
Layer-2 forwarding tests	TestNewIndexLearn TestMatchCapture TestDontConditionalLearn TestProtocolMatchChannel TestBadBpduTrap
NetFlow function	TestNetflowInlineRewrite
ACL/QOS function	TestAclPermit TestQosTcam TestAclDeny
IP version 4 function	TestIPv4FibShortcut TestFibDevices TestL3Capture2 TestNATFibShortcut
Multicast function	TestL3VlanMet
SPAN function	TestIngressSpan TestEgressSpan
Fabric connection	TestFabricSnakeForward TestFabricSnakeBackward
EOBC connection	Proceed to Step 3
Packet Buffer issues	Proceed to Step 4

Table 21-2 On-Demand Tests: Fabric-Enabled Modules

Functional Test Group	Individual Tests
Per-port tests	TestLoopback
Multicast function	TestL3VlanMet
SPAN function	TestIngressSpan TestEgressSpan
Fabric Tests	TestSynchedFabChannel

Table 21-3 On-Demand Tests: Non-Fabric-Enabled Modules

Functional Test Group	Individual Tests
Per-port tests	TestLoopback TestNetflowInlineRewrite

Step 3 Run the TestTrafficStress test.

Note With software release 8.5(1), the TestTrafficStress test is not available. This test might be available in subsequent releases. If the test is not available, proceed to the next step.

This disruptive packet-switching test is available only on the supervisor engine. The test pairs ports across the system so that packets are switched between those ports at line-rate for stress-testing the system. The test takes a few minutes to finish. During the test, all the ports are shut down and some ports might go up and down (flap). Note that any ports that are down will not come back up after the test is finished. Additional test requirements are as follows:

- User actions before running the test—All health-monitoring tests for the module should be disabled before running this test.
- User actions after running the test—None

Step 4 Run the TestEobcStressPing test.

Note With software release 8.5(1), the TestEobcStressPing test is not available. This test might be available in subsequent releases. If the test is not available, proceed to the next step.

This disruptive test checks the EOBC connection for the specified module. The test takes a couple of minutes to finish. You cannot run the packet-switching tests described in previous steps after running this test. However, you can run the tests described in Step 5. Additional requirements are as follows:

- User actions before running the test—You should disable all health-monitoring tests for the module before running this test because the EOBC connection is disrupted and will cause the health-monitoring tests to fail.
- User actions after running the test—Either run the tests mentioned in Step 5 or power cycle the module to return to normal operation. After the module comes online, reenable the health-monitoring tests that were disabled.

Step 5 Run the exhaustive memory tests.

Exhaustive memory tests exist for the supervisor engine and other modules. You should execute the memory tests on the supervisor engine only after the memory tests have been run on the other modules. This order is required because after running the supervisor engine memory tests, the system is in an unusable state and needs to be rebooted to return to a normal operating state.

Note With software release 8.5(1), memory tests are available only for the supervisor engine. Memory tests for other modules are planned for subsequent releases.

Note No other tests can be run on the supervisor engine or other modules after running the exhaustive memory tests.

Caution Before running any of the memory tests, you must follow all of the requirements listed in the "User actions before running the test" bullet.

You can run the exhaustive memory tests on an individual basis. Some of the tests can take several hours to finish due to the size of the memory. Since each module has several memory tests and they are interdependent, the order of running these tests on each module is critical.

Note With software release 8.5(1), the TestFibTcamSSRAM test is the only available exhaustive memory test. The other memory tests (items 2 through 5 below), are planned for subsequent releases.

The order for running these tests is as follows:

1. TestFibTcamSSRAM
2. TestAclQosTcam
3. TestNetflowTcam
4. TestAsicMemory
5. TestLinecardMemory

If a particular test does not exist for a module, it can be skipped.

Additional requirements are as follows:

- Before running the test:
  - Turn off all background health-monitoring tests on the supervisor engine and switching modules using the **clear diagnostic monitor module** *num* **test all** command.
  - Isolate network traffic by disabling all connected ports.
  - Before the test, make sure you do not send test packets during a memory test.
  - Remove all switching modules for testing FIB TCAM and SSRAM on the policy feature card (PFC) of the supervisor engine.

  Reset the system or the module that you are testing before returning the system to normal operating mode.

- After running the test:
  - For supervisor engines—Reboot the switch but do not save the configuration while rebooting because the configuration was changed during the test.
  - For other modules—Power cycle the modules. After the modules come online, reenable the health-monitor tests that were disabled.

Configuring Diagnostics Operations

You can specify that the on-demand online diagnostics continue to run until a configurable number of failures occur by entering the **continue** *failure_limit* keyword. The *failure_limit* range is 0 to 65534 failures. You can specify that the on-demand online diagnostics stop running when a single failure occurs by entering the **stop** keyword. You can specify that an on-demand test be run multiple times by entering the **iterations** *number_of_iterations* keyword. The *number_of_iterations* range is 1 to 999.

The complete syntax for these commands is as follows:

set diagnostic ondemand action-on-failure [**continue** *failure_limit* | **stop**]

set diagnostic ondemand iterations *number_of_iterations*

Configuring Online Diagnostic Health-Monitoring Tests

You can configure health-monitoring diagnostic testing on specified modules while the switch is connected to a live network. You can specify the execution interval for each health-monitoring test, whether or not to generate a system message upon test failure, or whether an individual test should be enabled or disabled.

The disruptive tests are disabled by default. A set number of nondisruptive tests (not all) are enabled by default. Use the **show diagnostic content module** *mod_list* command to determine which tests are disruptive (D) and nondisruptive (N) by checking the "Attributes" column. Use this information for configuring additional health-monitoring tests. We recommend that you use only nondisruptive tests for health monitoring.

To configure online diagnostic health-monitoring tests, perform this task in privileged mode:

	Task	Command
Step 1	Specify the online diagnostic monitoring interval.	**set diagnostic monitor interval module** *mod_num* **test** {**all** | *test_ID_num* | *test_list*} *hh:mm:ss*[1]
Step 2	(Optional) Enable health-monitoring diagnostic tests.	**set diagnostic monitor module** *mod_num* **test** {*test-id* | *test-id-range* | **all**}
Step 3	Enable syslog generation when a test fails.	**set diagnostic monitor syslog**
Step 4	Display the online diagnostic monitoring configuration.	**show diagnostic content module** {*mod_list* | **all**}

1. For the **interval** keyword, the range of milliseconds that can be specified is 0 to 999 and the number of days specified is 0 to 20.

This example shows how to specify that the online diagnostic health-monitoring tests (test 18) be run on module 7 at 12:12:12 and 100 milliseconds every 10 days:

```
Console> (enable) set diagnostic monitor interval module 7 test 18 12:12:12 100 10
Diagnostic monitor interval set at 12:12:12 100 10 for module 7 test 18
```

```
Console> (enable)
```

This example shows how to enable test 18 on module 7:

```
Console> (enable) set diagnostic monitor module 7 test 18
Module 7 test 18 diagnostic monitor enable.
Console> (enable)
```

This example shows how to enable syslog generation when a test fails:

```
Console> (enable) set diagnostic monitor syslog
Diagnostic monitor syslog enable.
Console> (enable)
```

Scheduling Online Diagnostics

You can schedule online diagnostics to run at a designated time of day or on a daily, weekly, or monthly basis for a specific module. You can specify that all tests be run or that individual tests be run. The tests can be scheduled to run only once or be repeated at specified intervals.

Note After you schedule the online diagnostics to run at a designated time, the online diagnostics will not run at the designated time if you change the system time using the **set time** command. For example, if you schedule the online diagnostics to run at 3:00 pm, then change the system time to 2:59 pm, the online diagnostics will not run at 3:00 pm.

To schedule online diagnostics, perform this task in privileged mode:

	Task	Command
Step 1	Schedule online diagnostics.	**set diagnostic schedule module** *slot_num* **test** {*test-id* | *test-id-range* | **all**} {[**port** {*port_num* | *port_num_range* | **all**}] | [**daily** *hh*:*mm*] [**on** *month day_of_month year hh*:*mm*] [**weekly** *day hh*:*mm*]}
Step 2	Display the online diagnostic scheduling.	**show diagnostic schedule module** *mod_list*

This example shows how to schedule diagnostic testing (tests 1 and 2 specified) to occur on a specific date and time for a specific module:

```
Console> (enable) set diagnostic schedule module 7 test 1 daily 12:12
Diagnostic schedule set at daily 12:12 for module 7 test 1
Console> (enable)
```

This example shows how to schedule diagnostic testing (test 1 specified) to occur daily at a certain time for a specific port and module:

```
Console> (enable) set diagnostic schedule module 7 test 3 port 1 daily 16:16
Diagnostic schedule set at daily 16:16 for module 7 test 3
Console> (enable)

Console> (enable) show diagnostic schedule module 7

Current Time = Fri Apr 15 2005, 16:56:06

Diagnostic for Module 7:
```

```
Schedule #1:
        To be run daily 12:12
        Test ID(s) to be executed: 1-2.

Schedule #2:
        To be run daily 16:16
        Test ID(s) to be executed: 3.
        Port(s) to be tested: 1.

Console> (enable)
```

Specifying the Online Diagnostic Failure Response

You can specify the online diagnostic failure response for the supervisor engine. If you specify the **ignore** keyword, the supervisor engine boots up after failing the online diagnostics. If you specify the **system** keyword (the default), the supervisor engine is kept offline and module-specific corrective action is taken.

To specify the online diagnostic failure response for the supervisor engine, perform this task in privileged mode:

	Task	Command
Step 1	Specify the online diagnostic failure response for the supervisor engine.	**set diagnostic diagfail-action** {**ignore** | **system**}
Step 2	Display the configuration settings for the online diagnostic failure response for the supervisor engine.	**show diagnostic diagfail-action**

This example shows how to specify that the supervisor engine goes offline after failing the online diagnostics:

```
Console> (enable) set diagnostic diagfail-action system
Diagnostic failure action set to system.
Console> (enable) show diagnostic diagfail-action
Diagnostic failure action at last bootup : system
Diagnostic failure action at next reset  : system
Console> (enable)
```

Specifying the Online Diagnostic Event Log Size

The default setting is 500 entries and the range is 1 to 10000 entries.

To specify the online diagnostic event-log size, perform this task in privileged mode:

Task	Command
Specify the online diagnostic event-log size.	**set diagnostic event-log size** [*size*]

This example shows how to specify 1000 entries for the online diagnostic event-log size:

```
Console> (enable) set diagnostic event-log size 1000
Diagnostic event-log size set to 1000
Console> (enable)
```

Displaying Online Diagnostic Tests and Test Results

You can display the online diagnostic tests that are configured for specific modules and check the results of the tests using the **show** commands.

To display online diagnostic test information, perform these tasks in normal mode:

Task	Command
Display the bootup diagnostic level.	**show diagnostic bootup level**
Display the test content for the specified module(s) or all modules.	**show diagnostic content module** [*mod_num* | **all**]
Display the configuration settings for the online diagnostic failure response for the supervisor engine.	**show diagnostic diagfail-action**
Display the diagnostic event log.	**show diagnostic events** [**event-type** {**error** | **info** | **warning**}] **show diagnostic events** [**module** {*mod_list* | **all**}] **show diagnostic events**
Display the online diagnostic on-demand configuration settings.	**show diagnostic ondemand settings**
Display the diagnostic test results for the specified module(s) or all modules.	**show diagnostic result module** *mod_list* | **all** [**detail** | **test**] [*test_list*] [**detail**]
Display the online diagnostic scheduling.	**show diagnostic schedule module** *mod_list*
Display the current online diagnostic status for all modules.	**show diagnostic status**

Clearing the Online Diagnostic Configuration

To clear online diagnostic configuration parameters, perform these tasks in normal mode:

Task	Command
Clear the bootup online diagnostic level.	**clear diagnostic bootup level**
Clear the online diagnostic event-log size.	**clear diagnostic event-log size**
Clear the online diagnostic health-monitoring configuration.	**clear diagnostic monitor interval module** *mod_list* **test** [*test_list* | **all**] **clear diagnostic monitor module** *mod* **test** *test_list* **clear diagnostic monitor syslog**

Task	Command
Disable syslog generation that occurs when a test fails.	**clear diagnostic monitor syslog**
Clear online diagnostic scheduling information.	**clear diagnostic schedule module** *mod_num* **test** {*test-id* | *test-id-range* | **all**} {[**port** {*port_num* | *port_range* | **all**}] | [**device** {*device_num* | *device_range* | **all**}]}

This example shows how to clear the bootup online diagnostic level:

```
Console> (enable) clear diagnostic bootup level
Diagnostic level set to bypass
Console> (enable)
```

This example shows how to clear the online diagnostic event-log size:

```
Console> (enable) clear diagnostic event-log size
Diagnostic event-log size set to default(500)
Console> (enable)
```

These examples show how to clear the online diagnostic monitoring configuration:

```
Console> (enable) clear diagnostic monitor interval module 7 test 3
Clear diagnostic monitor interval for module 7 test 3
Console> (enable)

Console> (enable) clear diagnostic monitor module 7 test 1
Module 7 test 1 diagnostic monitor disable.
Console> (enable)

Console> (enable) clear diagnostic monitor syslog
Diagnostic monitor syslog disable.
Console> (enable)
```

Clear the online diagnostic scheduling configuration for tests 1 and 2 on module 7:

```
Console> (enable) clear diagnostic schedule module 7 test 1-2 daily 12:12
Clear diagnostic schedule at daily 12:12 for module 7 test 1-2
Console> (enable)
```

CHAPTER 22

Administering the Switch

This chapter describes how to perform the various administrative tasks on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Setting the System Name and System Prompt on the Switch

The system name on the switch is a user-configurable string that is used to identify the device. The default configuration has no system name configured.

If you do not manually configure a system name, the system name is obtained through the Domain Name System (DNS) if you configure the switch as follows:

- Assign an IP address that is mapped to the switch name on the DNS server to the sc0 interface.
- Enable DNS on the switch
- Specify at least one valid DNS server on the switch

If the DNS lookup is successful, the DNS host name of the switch is configured as the system name of the switch and is saved in NVRAM (the domain name is removed).

If you have not configured a system prompt, the first 20 characters of the system name are used as the system prompt (a greater-than symbol [>] is appended). The prompt is updated whenever the system name changes, unless you manually configure the prompt using the **set prompt** command.

The switch performs a DNS lookup for the system name whenever one of the following occurs:

- The switch is initialized (power on or reset)
- You configure the IP address on the sc0 interface using the command-line interface (CLI) or Simple Network Management Protocol (SNMP)
- You configure a route using the **set ip route** command
- You clear the system name using the **set system name** command
- You enable DNS or specify DNS servers

If the system name is user configured, no DNS lookup is performed.

Setting the Static System Name and Prompt

These sections describe how to set the static system name and prompt:

- Setting the Static System Name, page 22-2
- Setting the Static System Prompt, page 22-3
- Clearing the System Name, page 22-3

Setting the Static System Name

To set a static system name, perform this task in privileged mode:

Task	Command
Set the static system name.	**set system name** *name_string*

Note When you set the system name, the system name is used as the system prompt. You can override the prompt string with the **set prompt** command.

This example shows how to configure the system name on the switch:

```
Console> (enable) set system name Catalyst 6500
System name set.
Catalyst 6500> (enable)
```

Setting the Static System Prompt

To set the static system prompt, perform this task in privileged mode:

Task	Command
Set the static system prompt.	**set prompt** *prompt_string*

This example shows how to set the static system prompt on the switch:

```
Console> (enable) set prompt Catalyst6509>
Catalyst6509> (enable)
```

Clearing the System Name

To clear the system name, perform this task in privileged mode:

Task	Command
Clear the system name.	**set system name**

This example shows how to clear the system name:

```
Console> (enable) set system name
System name cleared.
Console> (enable)
```

Setting the System Contact and Location on the Switch

You can set the system contact and location to help you with resource management tasks.

To set the system contact and location, perform this task in privileged mode:

	Task	Command
Step 1	Set the system contact.	**set system contact** [*contact_string*]
Step 2	Set the system location.	**set system location** [*location_string*]
Step 3	Verify the global system information.	**show system**

This example shows how to set the system contact and location and verify the configuration:

```
Catalyst 6500> (enable) set system contact sysadmin@corp.com
System contact set.
Catalyst 6500> (enable) set system location Sunnyvale CA
System location set.
Catalyst 6500> (enable) show system
PS1-Status PS2-Status Fan-Status Temp-Alarm Sys-Status Uptime d,h:m:s Logout
---------- ---------- ---------- ---------- ---------- -------------- ---------
ok         none       ok         off        ok         0,04:04:07     20 min

PS1-Type   PS2-Type   Modem   Baud  Traffic Peak Peak-Time
---------- ---------- ------- ----- ------- ---- -------------------------
other      none       disable  9600   0%      0% Tue Jun 23 1998, 16:51:36

System Name              System Location          System Contact
------------------------ ------------------------ ------------------------
Catalyst 6500            Sunnyvale CA             sysadmin@corp.com
Catalyst 6500> (enable)
```

Setting the System Clock on the Switch

Note You can configure the switch to obtain the time and date using the Network Time Protocol (NTP). For information on configuring NTP, see Chapter 34, "Configuring NTP."

To set the system clock, perform this task in privileged mode:

	Task	Command
Step 1	Set the system clock.	**set time** [*day_of_week*] [*mm/dd/yy*] [*hh:mm:ss*]
Step 2	Display the current date and time.	**show time**

This example shows how to set the system clock and display the current date and time:

```
Console> (enable) set time Mon 06/15/98 12:30:00
Mon Jun 15 1998, 12:30:00
Console> (enable) show time
Mon Jun 15 1998, 12:30:02
Console> (enable)
```

Creating a Login Banner on the Switch

You can create a single or multiline message banner that appears on the screen when someone logs in to the switch. The first character following the **motd** keyword is used to delimit the beginning and end of the banner text. The characters following the ending delimiter are discarded. After entering the ending delimiter, press **Return**. The banner must be fewer than 3070 characters.

These sections describe how to configure and clear a login banner:

- Configuring a Login Banner, page 22-5
- Clearing a Login Banner, page 22-5

Configuring a Login Banner

To configure a login banner, perform this task in privileged mode:

	Task	Command
Step 1	Enter the message of the day.	**set banner motd** *c message_of_the_day c*
Step 2	Display the login banner by logging out and logging back into the switch.	–

This example shows how to configure a login banner on the switch using the **#** symbol as the beginning and ending delimiter:

```
Console> (enable) set banner motd #
Welcome to the Catalyst 6500 Switch!
Unauthorized access prohibited.
Contact sysadmin@corp.com for access.
#
MOTD banner set
Console> (enable)
```

Clearing a Login Banner

To clear a login banner, perform this task in privileged mode:

Task	Command
Clear the message of the day.	**set banner motd** *cc*

This example shows how to clear a login banner:

```
Console> (enable) set banner motd ##
MOTD banner cleared
Console> (enable)
```

Displaying or Suppressing the "Cisco Systems Console" Telnet Login Banner on the Switch

To display or suppress the "Cisco Systems Console" Telnet login banner, perform this task in privileged mode:

Note By default, the Cisco Systems Console Telnet login banner is enabled.

	Task	Command
Step 1	Display or suppress the Cisco Systems Console Telnet login banner.	**set banner telnet** {**enable** | **disable**}
Step 2	Display the Cisco Systems Console Telnet login banner setting.	**show banner**

This example shows how to enable the Cisco Systems Console Telnet login banner:

```
Console> (enable) set banner telnet enable
Cisco Systems Console banner will be printed at telnet.
Console> (enable)
```

This example shows how to disable the Cisco Systems Console Telnet login banner:

```
Console> (enable) set banner telnet disable
Cisco Systems Console banner will not be printed at telnet.
Console> (enable)
```

This example shows how to display the Cisco Systems Console Telnet login banner setting:

```
Console> (enable) show banner
MOTD banner:

LCD config:

Telnet Banner:
disabled
Console> (enable)
```

Defining Command Aliases on the Switch

You can use the **set alias** command to define up to 100 command aliases (shorthand versions of commands) for frequently used or long and complex commands. The command aliases can save you time and can help to prevent typing errors when you are configuring or monitoring the switch.

The *name* argument defines the command alias. The *command* and *parameter* arguments define the command to enter when the command alias is entered at the command line.

To define a command alias on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Define a command alias on the switch.	**set alias** *name command* [*parameter*] [*parameter*]
Step 2	Verify the currently defined command aliases.	**show alias** [*name*]

This example shows how to define two command aliases, **sm8** and **sp8**. **sm8** issues the **show module 8** command, and **sp8** issues the **show port 8** command. This example also shows how to verify the currently defined command aliases and displays what happens when you enter the command aliases at the command line:

```
Console> (enable) set alias sm8 show module 8
Command alias added.
Console> (enable) set alias sp8 show port 8
Command alias added.
Console> (enable) show alias
```

```
sm8             show module 8
sp8             show port 8
Console> (enable) sm8
Mod Module-Name         Ports Module-Type           Model    Serial-Num Status
--- ------------------- ----- --------------------- --------- --------- -------
8                       2     DS3 Dual PHY ATM      WS-X5166  007243262 ok

Mod MAC-Address(es)                         Hw     Fw          Sw
--- --------------------------------------- ------ ----------- ------------------
8   00-60-2f-45-26-2f                       2.0    1.3         51.1(103)
Console> (enable) sp8
Port  Name           Status  Vlan    Level Duplex Speed Type
----- ------------------ ---------- ---------- ------ ------ ----- ------------
 8/1                     notconnect trunk      normal   full    45 DS3 ATM
 8/2                     notconnect trunk      normal   full    45 DS3 ATM

Port   ifIndex
-----  -------
 8/1   285
 8/2   286


Use 'session' command to see ATM counters.

Last-Time-Cleared
--------------------------
Thu Sep 10 1998, 16:56:08
Console> (enable)
```

Defining IP Aliases on the Switch

You can use the **set ip alias** command to define textual aliases for IP addresses. IP aliases can make it easier to refer to other network devices when using **ping**, **telnet**, and other commands, even when DNS is not enabled.

The *name* argument defines the IP alias. The *ip_addr* argument defines the IP address to which the name refers.

To define an IP alias on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Define an IP alias on the switch.	**set ip alias** *name ip_addr*
Step 2	Verify the currently defined IP aliases.	**show ip alias** [*name*]

This example shows how to define two IP aliases, **sparc** and **cat6509**. **sparc** refers to IP address 172.20.52.3, and **cat6509** refers to IP address 172.20.52.71. This example also shows how to verify the currently defined IP aliases and displays what happens when you use the IP aliases with the **ping** command:

```
Console> (enable) set ip alias sparc 172.20.52.3
IP alias added.
Console> (enable) set ip alias cat6509 172.20.52.71
IP alias added.
Console> (enable) show ip alias
default         0.0.0.0
sparc           172.20.52.3
cat6509         172.20.52.71
```

```
Console> (enable) ping sparc
sparc is alive
Console> (enable) ping cat6509
cat6509 is alive
Console> (enable)
```

Configuring Static Routes on the Switch

Note For information on configuring a default gateway (default route), see the "Configuring the Default Gateways" section on page 3-8.

In some situations, you might need to add a static routing table entry for one or more destination networks. The static route entries consist of the destination IP network address, the IP address of the next hop router, and the metric (hop count) for the route.

The destination IP network address can be variably subnetted to support Classless Interdomain Routing (CIDR). You can specify the subnet mask (*netmask*) for a destination network using the number of subnet bits or using the subnet mask in dotted decimal format. If no subnet mask is specified, the default (classful) mask is used.

The switch forwards the IP traffic that is generated by the switch using the longest address match in the IP routing table. The switch does not use the IP routing table to forward the traffic from the connected devices, only the IP traffic that is generated by the switch itself (for example, Telnet, TFTP, and ping).

To configure a static route, perform this task in privileged mode:

	Task	Command
Step 1	Configure a static route to the remote network.	**set ip route** *destination*[*/netmask*] *gateway* [*metric*]
Step 2	Verify that the static route appears correctly in the IP routing table.	**show ip route**

This example shows how to configure a static route on the switch and verify that the route is configured properly in the routing table:

```
Console> (enable) set ip route 172.16.16.0/20 172.20.52.127
Route added.
Console> (enable) show ip route
Fragmentation   Redirect   Unreachable
-------------   --------   -----------
enabled         enabled    enabled

The primary gateway: 172.20.52.121
Destination      Gateway          RouteMask    Flags   Use       Interface
---------------  ---------------  ----------   -----   --------  ---------
172.16.16.0      172.20.52.127    0xfffff000   UG      0           sc0
default          172.20.52.121    0x0          UG      0           sc0
172.20.52.120    172.20.52.124    0xffffffff8  U       1           sc0
default          default          0xff000000   UH      0           sl0
Console> (enable)
```

Configuring Permanent and Static ARP Entries on the Switch

To enable your Catalyst LAN switch to communicate with devices that do not respond to Address Resolution Protocol (ARP) requests, you can configure a static or permanent ARP entry that maps the IP addresses of those devices to their MAC addresses. You can configure an ARP entry so that it does not age out by configuring it as either static or permanent. When you configure a static ARP entry using the **set arp static** command, the entry is removed from the ARP cache after a system reset. When you configure a permanent ARP by using the **set arp permanent** command, the ARP entry is retained even after a system reset.

Because most hosts support dynamic resolution, you usually do not need to specify static or permanent ARP cache entries. When a device does not respond to ARP requests, you can configure an ARP entry to be statically or permanently entered into the ARP cache so that those devices can still be reached.

To configure a static or permanent ARP entry, perform this task in privileged mode:

	Task	Command
Step 1	Configure a static or permanent ARP entry.	**set arp** [**dynamic** | **permanent** | **static**] {*ip_addr hw_addr*}
Step 2	(Optional) Specify the ARP aging time.	**set arp agingtime** *seconds*
Step 3	Verify the ARP configuration.	**show arp**

This example shows how to define a static ARP entry:

```
Console> (enable) set arp static 20.1.1.1 00-80-1c-93-80-40
Static ARP entry added as
20.1.1.1 at 00-80-1c-93-80-40 on vlan 1
Console> (enable)
```

This example shows how to define a permanent ARP entry:

```
Console> (enable) set arp permanent 10.1.1.1 00-80-1c-93-80-60
Permanent ARP entry added as
10.1.1.1 at 00-80-1c-93-80-60 on vlan 1
Console> (enable)
```

This example shows how to set the ARP aging time:

```
Console> (enable) set arp agingtime 300
ARP aging time set to 300 seconds.
Console> (enable)
```

This example shows how to display the ARP cache:

```
Console> (enable) show arp
ARP Aging time = 300 sec
+ - Permanent Arp Entries
* - Static Arp Entries
+ 10.1.1.1 at 00-80-1c-93-80-60 on vlan 1
172.20.52.1 at 00-60-5c-86-5b-28 port  8/1 on vlan 1
* 20.1.1.1 at 00-80-1c-93-80-40 port  8/1 on vlan 1
Console> (enable)
```

To clear the ARP entries, perform this task in privileged mode:

	Task	Command
Step 1	Clear a dynamic, static, or permanent ARP entry.	**clear arp** [**dynamic** | **permanent** | **static**] {*ip_addr hw_addr*}
Step 2	Clear ARP entry for a single host	**clear arp** *x.x.x.x* **Note** *x.x.x.x* is the IP address of the host.
Step 3	Verify the ARP configuration.	**show arp**

This example shows how to clear all the permanent ARP entries and verify the configuration:

```
Console> (enable) clear arp permanent
Permanent ARP entries cleared.
Console> (enable)
Console> (enable) show arp
ARP Aging time = 300 sec
+ - Permanent Arp Entries
* - Static Arp Entries
172.20.52.1 at 00-60-5c-86-5b-28 port  8/1 on vlan 1
* 20.1.1.1 at 00-80-1c-93-80-40 port  8/1 on vlan 1
Console> (enable)
```

This example shows how to clear the ARP entry of a host:

```
Console> (enable) clear arp 172.22.145.1
ARP entry deleted.
Console> (enable)
```

Scheduling a System Reset on the Switch

These sections describe how to schedule a system reset:

- Scheduling a Reset at a Specific Time, page 22-10
- Scheduling a Reset Within a Specified Amount of Time, page 22-11

You can use the **schedule reset** command to schedule a system to reset at a future time. This feature allows you to upgrade the software during business hours and schedule the system upgrade after business hours to avoid a major impact on users.

You can also use schedule reset when trying new features on a switch. To avoid misconfiguring or losing the network connectivity to the device, you can set the startup configuration and schedule a reset to occur in 30 minutes. You can then change the configuration, and if connectivity is lost, the system resets in 30 minutes and returns to the previous configuration.

Scheduling a Reset at a Specific Time

You can specify an absolute time and date at which the reset should take place with the **reset at** command. Entering the month and day argument with this command is optional. If you do not specify the month and day, the reset takes place on the current day if the time that is specified is later than the current time. If the time that is scheduled for reset is earlier than the current time, the reset takes place on the following day.

Note The maximum scheduled reset time is 24 days.

To schedule a reset at a specific time, perform this task in privileged mode:

	Task	Command
Step 1	Schedule the reset time at a specific time.	**reset** [**mindown**] **at** {*hh:mm*} [*mm/dd*] [*reason*]
Step 2	Verify the scheduled reset.	**show reset**

Note The minimum downtime argument is valid only if the system has a standby supervisor engine.

This example shows how to schedule a reset at a specific time:

```
Console> (enable) reset at 20:00
Reset scheduled at 20:00:00, Wed Aug 18 1999.
Proceed with scheduled reset? (y/n) [n]? y
Reset scheduled for 20:00:00, Wed Aug 18 1999 (in 0 day 5 hours 40 minutes).
Console> (enable)
```

This example shows how to schedule a reset at a specific time and include a reason for the reset:

```
Console> (enable) reset at 23:00 8/18 Software upgrade to 5.3(1).
Reset scheduled at 23:00:00, Wed Aug 18 1999.
Reset reason: Software upgrade to 5.3(1).
Proceed with scheduled reset? (y/n) [n]? y
Reset scheduled for 23:00:00, Wed Aug 18 1999 (in 0 day 8 hours 39 minutes).
Console> (enable)
```

This example shows how to schedule a reset with a minimum downtime:

```
Console> (enable) reset mindown at 23:00 8/18 Software upgrade to 5.3(1).
Reset scheduled at 23:00:00, Wed Aug 18 1999.
Reset reason: Software upgrade to 5.3(1).
Proceed with scheduled reset? (y/n) [n]? y
Reset mindown scheduled for 23:00:00, Wed Aug 18 1999 (in 0 day 8 hours 39 minutes).
Console> (enable)
```

Scheduling a Reset Within a Specified Amount of Time

You can schedule a reset within a specified time with the **reset in** command. For instance, if the current system time is 9:00 a.m. and the reset is scheduled in one hour, the scheduled reset takes place at 10:00 a.m. If you or NTP advances the system clock to 10:00 a.m., the reset takes place at 11:00 a.m. If the clock is advanced ahead of the scheduled reset time, the reset takes place 5 minutes after the current time.

To schedule a reset within a specified time, perform this task in privileged mode:

	Task	Command
Step 1	Schedule the reset time within a specific amount of time.	**reset** [**mindown**] **in** [*hh*] {*mm*} [*reason*]
Step 2	Verify the scheduled reset.	**show reset**

Note The minimum downtime argument is valid only if the system has a standby supervisor engine.

This example shows how to schedule a reset in a specified time:

```
Console> (enable) reset in 5:20 Configuration update
Reset scheduled in 5 hours 20 minutes.
Reset reason: Configuration update
Proceed with scheduled reset? (y/n) [n]? y
Reset scheduled for 19:56:01, Wed Aug 18 1999 (in 5 hours 20 minutes).
Reset reason: Configuration update
Console> (enable)
```

Power Management

This section describes power management in the Catalyst 6500 series switches and includes the following information:

Note In systems with redundant power supplies, both power supplies must have the same wattage. The Catalyst 6500 series switches allow you to mix AC-input and DC-input power supplies in the same chassis. For detailed information on supported power supply configurations for each chassis, refer to the *Catalyst 6500 Series Switch Installation Guide*.

Catalyst 6500 series modules have different power requirements. Depending upon the wattage of the power supply, certain switch configurations might require more power than a single power supply can provide. Although the power management feature allows you to power all installed modules with two power supplies, redundancy is not supported in this configuration. The redundant and nonredundant power configurations are discussed in the following sections.

Enabling or Disabling Power Redundancy

Enter the **set power redundancy enable | disable** command to enable or disable redundancy (redundancy is enabled by default). With redundancy enabled and two power supplies of equal wattage installed, the total power that is drawn from both supplies is at no time greater than the capability of one supply. If one supply malfunctions, the other supply can take over the entire system load. When you install and turn on two power supplies of equal wattage, each concurrently provides approximately half of the required power to the system. Load sharing and redundancy are enabled automatically; no software configuration is required.

With redundancy enabled, if you power up the system with two power supplies of unequal wattage, both power supplies come online but a syslog message displays that the lower wattage power supply will be disabled. If the active power supply fails, the lower wattage power supply that was disabled comes online and, if necessary, the modules are powered down to accommodate the lower wattage power supply.

In a nonredundant configuration, the power that is available to the system is the combined power capability of both power supplies. The system powers up as many modules as the combined capacity allows. However, if one supply should fail and there is not enough power for all the previously powered up modules, the system powers down some modules. These modules are marked as *power-deny* in the **show module** Status field.

You can change the configuration of the power supplies to redundant or nonredundant at any time. If you switch from a redundant to a nonredundant configuration, both power supplies are enabled (even a power supply that was disabled because it was of a lower wattage than the other power supply). If you change from a nonredundant to a redundant configuration, both power supplies are initially enabled, and if they are of the same wattage, remain enabled. If they are of different wattage, a syslog message displays and the lower wattage supply is disabled.

Table 22-1 describes how the system responds to changes in the power supply configuration.

Table 22-1 Effects of Power Supply Configuration Changes

Configuration Change	Effect
Redundant to nonredundant	• System log and syslog messages are generated. • The system power is increased to the combined power capability of both supplies. • The modules marked as *power-deny* in the **show module** Status field are brought up if there is sufficient power.
Nonredundant to redundant	• System log and syslog messages are generated. • The system power is the power capability of the larger wattage supply. • If there is not enough power for all previously powered-up modules, some modules are powered down and marked as *power-deny* in the **show module** Status field.
Equal wattage power supply is inserted with redundancy enabled	• System log and syslog messages are generated. • The system power equals the power capability of one supply. • No change in the module status because the power capability is unchanged.
Equal wattage power supply is inserted with redundancy disabled	• System log and syslog messages are generated. • The system power is the combined power capability of both supplies. • The modules marked as *power-deny* in the **show module** Status field are brought up if there is sufficient power.
Higher wattage power supply is inserted with redundancy enabled	• System log and syslog messages are generated. • The system disables the lower wattage power supply; the higher wattage supply powers the system.
Lower wattage power supply is inserted with redundancy enabled	• System log and syslog messages are generated. • The system disables the lower wattage power supply; the higher wattage supply powers the system.

Table 22-1 Effects of Power Supply Configuration Changes (continued)

Configuration Change	Effect
Higher or lower wattage power supply is inserted with redundancy disabled	• System log and syslog messages are generated. • The system power is increased to the combined power capability of both supplies. • The modules marked as *power-deny* in the **show module** Status field are brought up if there is sufficient power.
Power supply is removed with redundancy enabled	• System log and syslog messages are generated. • If the power supplies are of equal wattage, there is no change in the module status because the power capability is unchanged. If the power supplies are of unequal wattage and the lower wattage supply is removed, there is no change in the module status. If the power supplies are of unequal wattage and the higher wattage supply is removed, and if there is not enough power for all previously powered-up modules, some modules are powered down and marked as *power-deny* in the **show module** Status field.
Power supply is removed with redundancy disabled	• System log and syslog messages are generated. • The system power is decreased to the power capability of one supply. • If there is not enough power for all previously powered-up modules, some modules are powered down and marked as *power-deny* in the **show module** Status field.
System is booted with power supplies of different wattage installed and redundancy enabled	• System log and syslog messages are generated. • The lower wattage supply is disabled.
System is booted with power supplies of equal or different wattage installed and redundancy disabled	• System log and syslog messages are generated. • The system power equals the combined power capability of both supplies. • The system powers up as many modules as the combined capacity allows.

Using the CLI to Power Modules Up or Down

You can power down a properly working module from the command-line interface (CLI) by entering the **set module power down** *mod* command. The module is marked as *power-down* in the **show module** Status field. Enter the **set module power up** *mod* command to check if adequate power is available in the system to turn on the power for a module that was previously powered down. If there is not enough power available, the module status changes from *power-down* to *power-deny*.

Environmental Monitoring

Environmental monitoring of chassis components provides early warning indications of possible component failure to ensure safe and reliable system operation and avoid network interruptions. This section describes how to monitor these critical system components, enabling you to identify and rapidly correct the hardware-related problems in your system.

The following sections describe the environmental monitors:

Environmental Monitoring Using CLI Commands

Enter the **show test** [*mod*] command to display the errors that are reported from the diagnostic tests. If you do not specify a module number, the test statistics are given for the general system and for the module in slot 1. If there are no errors, PASS is displayed in the Line Card Status field.

Enter the **show environment** [**temperature** | **all** | **power**] command to display the system status information. The keyword descriptions are as follows:

- **temperature**—(Optional) Displays temperature information.
- **all**—(Optional) Displays environmental status (for example, power supply, fan status, and temperature information) and information about the power that is available to the system.
- **power**—(Optional) Displays environmental power information.

Note By default, the alarm thresholds for environment temperature are set on each hardware component. You cannot modify the thresholds.

LED Indications

There are two alarm types, major and minor. The major alarms indicate a critical problem that could lead to the system being shut down. The minor alarms are for informational purposes only, giving you notice of a problem that could turn critical if corrective action is not taken.

When the system has an alarm (major or minor), indicating an overtemperature condition, the alarm is not canceled or any action taken (such as a module reset or shutdown) for 5 minutes. If the temperature falls 5°C (41°F) below the alarm threshold during this period, the alarm is canceled.

Table 22-2 lists the environmental indicators for the supervisor engine and switching modules.

Note For additional information on the LED indications, refer to the *Catalyst 6500 Series Switch Module Installation Guide*.

Table 22-2 Environmental Monitoring for Supervisor Engine and Switching Modules

Component	Alarm Type	LED Indication	Action
Supervisor engine temperature sensor exceeds major threshold[1]	Major	STATUS[2] LED red[3]	syslog message and SNMP trap generated. If redundancy, system switches to the redundant supervisor engine and the active supervisor engine shuts down. If there is no redundancy and the overtemperature condition is not corrected, the system shuts down after 5 minutes.
Supervisor engine temperature sensor exceeds minor threshold	Minor	STATUS LED orange	syslog message and SNMP trap generated. Monitor the condition.
Redundant supervisor engine temperature sensor exceeds major or minor threshold	Major	STATUS LED red	syslog message and SNMP trap generated. If major alarm and the overtemperature condition is not corrected, the system shuts down after 5 minutes.
	Minor	STATUS LED orange	If minor alarm, monitor the condition.
Switching module temperature sensor exceeds major threshold	Major	STATUS LED red	syslog message and SNMP trap generated. Power down the module[4].
Switching module temperature sensor exceeds minor threshold	Minor	STATUS LED orange	syslog message and SNMP trap generated. Monitor the condition.

1. The temperature sensors monitor the key supervisor engine components including the daughter cards.
2. A STATUS LED is located on the supervisor engine front panel and all the module front panels.
3. The STATUS LED is red on the failed supervisor engine. If there is no redundant supervisor engine, the SYSTEM LED is red also.
4. See the "Power Management" section on page 22-12 for instructions.

Displaying System Status Information for Technical Support

These sections describe how to display the system status information for technical support:

Generating a System Status Report

Using a single command, you can generate a report that contains status information about your switch. The generated information is useful if you need to report a problem to the Cisco Technical Assistance Center (TAC). This command is a combination of several **show system status** commands. You can upload the output of the command to a TFTP server, where you can send it to TAC.

You can use keywords to limit the output to certain areas, such as the specific modules, VLANs, ports, and so forth. If you do not specify any keywords, a report for the entire system is generated.

To generate a report and upload the report to a TFTP server, perform this task in privileged mode:

Task	Command
Generate a system status report that you can send to TAC.	**write tech-support** {*host*} {*filename*} [**module** *mod*] [**port** *mod*/*port*] [**vlan** *vlan*] [**memory**] [**config**]

This example shows a report that is sent to host 172.20.32.10 to a filename that you supply. No keywords are specified, so the complete status of the switch is included in the report.

```
Console> (enable) write tech-support 172.20.32.10 tech.txt
Upload tech-report to tech.txt on 172.20.32.10 (y/n) [n]? y
Finished network upload.  (67784 bytes)
Console> (enable)
```

Using System Dump Files

The core dump and the stack dump generate reports that contain the status information about your switch. Send the images that are captured by the core dump or the stack dump to Cisco TAC for analysis.

Enabling and Disabling the Core Dump

A core dump produces a comprehensive report of images when your system fails due to a software error. This report contains the system memory content, including the text, code, and stack segments. The core image is produced in Cisco core file format and is stored in the file system. By examining the core dump file, TAC can analyze the error condition of a terminated process.

Enter the **set system core-dump** command to enable or disable the core dump. If the switch has a redundant supervisor engine, the standby supervisor engine takes over automatically before the core dump occurs. The previously active supervisor engine resets itself after the core dump is complete.

To enable or disable the core dump, perform this task in privileged mode:

Task	Command
Enable or disable the core dump.	**set system core-dump** {**enable** | **disable**}

This example shows how to enable the core dump:

```
Console> (enable) set system core-dump enable
(1) In the event of a system crash, this feature will
     cause a core file to be written out.
(2) Core file generation may take up to 20 minutes.
(3) Selected core file is slot0:crash.hz
```

```
(4) Please make sure the above device has been installed,
    and ready to use
Core-dump enabled
Console> (enable)
```

This example shows how to disable the core dump:

```
Console> (enable) set system core-dump disable
Core-dump disabled
Console> (enable)
```

The size of the file system depends on the size of your memory card. An error process will generate a core image that is proportional to the size of the system DRAM. Make sure that you have enough available memory to store the core dump file.

Specifying the Core Image Filename

Enter the **set system core-file** command to specify the core image filename. The default filename is "slot0:crash.hz." This command automatically checks the validity of the device name that you input.

To specify the core image filename, perform this task in privileged mode:

Task	Command
Specify the core image filename.	**set system core-file** {*device:filename*}

This example shows how to specify the core image filename:

```
Console> (enable) set system core-file slot0:core.hz
System core-file set.
Console> (enable)
```

Displaying the Stack Dump

A stack dump provides only the images that are related to a particular process that has caused the system to fail. This image stack is displayed on the console and is also saved in the log area. The stack dump is automatic and becomes available when you enter the **show log** command after you reboot your system.

To display the log information, perform this task in normal mode:

Task	Command
Display the stack dump.	**show log**

This example shows an image stack that may display after you enter the **show log** command:

```
Breakpoint Exception occurred.
Software version =  6.2(0.83)
Process ID #52, Name
= Console
             EPC: 807523F4
    Stack content:
    sp+00: 00000000 80A75698 00000005 00000005
    sp+10: BE000A00 00000000 83F84150 801194B8
    sp+20: 80A75698 80A74BC8 80C8DBDC 000006E8
    sp+30: 8006AF30 8006AE98 82040664 00000630
    sp+40: 801AC744 801AC734 80A32488 80A32484
```

```
    sp+50: 80A3249C 00000000 00000002 000009E4
    sp+60: 8204067B 82040670 8011812C 81CAFC98
    sp+70: 8011814C 82040670 8011812C 81CAFC98
    sp+80: 00000002 000009E4 80110160 80110088
    sp+90: 82040670 80A71EB4 81F1E9F8 00000004
    sp+A0: 00000000 81F25EAC 81FF5750 00000000
    sp+B0: 00000000 00000000 81F1E314 800840BC
    sp+C0: 0000000B 80084EB0 00000001 8073A358
    sp+D0: 00000003 0000000D 00000000 0000000A
    sp+E0: 00000020 00000000 800831B4 0000001A
    sp+F0: 00000000 00000000 00000000 000D84F0
    Register content:
      Status: 3401FC23    Cause: 00000024
AT: 81640000
          V0: 00000007       V1: 00000007
          A0: 00000000       A1: 80A756A6
          A2: 00000011       A3: BE000BD0
          T0: BFFFFFFE       T1: 80000000
          T2: 00000000       T3: 00000001
          T4: 00000000       T5: 00000007
          T6: 00000000       T7: 00000000
          S0: 00000001       S1: 00000032
          S2: 81F1E9F8       S3: 80A74BC8
          S4: 80C8DBDC       S5: 000006E8
          S6: 00000000       S7: 00000000
          T8: F0D09E3A       T9: 82940828
          K0: 3041C001       K1: 80C73038
          GP: 811F39C0       SP: 83F84010
          S8: 83F84010       RA: 807523F4
        HIGH: 00000001      LOW: D5555559
    BADVADDR: 7DFF7FFF  ERR EPC: 58982466
GDB: Breakpoint Exception
GDB: The system has trapped into the debugger.
GDB: It will hang until examined with gdb.
```

Using System Crash-Info Files

The crash-info file contains extended system information that is captured when the system reloads due to an error. Similar to the crash-dump file, the crash-info file is stored in the file system. You should look at the information in the crash-info file in addition to the core dump information. By examining both the crash-info file and core dump file, Cisco TAC can better analyze the error.

Enabling and Disabling the Crash-Info File

To enable the system to write a crash-info file after a system reload occurs due to an error, perform this task in privileged mode:

Task	Command
Enable or disable creation of the crash-info file. **Note** This feature is disabled by default.	**set system crashinfo enable | disable**

This example shows how to enable the system to write a crash-info file:

```
Console> (enable) set system crashinfo enable
Crashinfo enabled
```

Specifying the Crash-Info Filename

Enter the **set system crash-info-file** command to specify the crash-info filename. This command automatically checks the validity of the device name that you input.

To specify the crash-info filename, perform this task in privileged mode:

Task	Command
Specify the crash-info filename. The default filename is **crashinfo**.	**set system crashinfo-file** {*device:filename*}

This example shows how to specify the crash-info filename:

```
Console> (enable) set system crashinfo-file slot0:crashinfo
System crashinfo-file set.
Console> (enable)
```

Logging System Information to a TFTP or rcp Server

You can configure your system to execute up to 15 **show** commands and to log the output of these commands in a file on a specified server. You can use the information in the output for debugging and troubleshooting purposes.

These sections describe how to configure system information logging on the switch:

Enabling System Information Logging

By default, system information logging is disabled.

To enable system information logging on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Enable system information logging.	**set system info-log enable**
Step 2	Verify that system information logging is enabled.	**show system info-log**

This example shows how to enable system information logging and verify that it is enabled:

```
Console> (enable) set system info-log enable
Successfully enabled system information logging.
Console> (enable) show system info-log
System Logging  Host            File                                Interval
--------------  --------------  ----------------------------------  --------
Enabled         -               tftp:sysinfo                        1440
Index        System Command
-----        ------------------------------------------------------------
Console> (enable)
```

Specifying show Commands for System Information Logging

You can specify up to 15 **show** commands whose output is periodically logged in a file on a specified server. You must use a delimiting character on either side of the **show** command. You can enter only one **show** command at a time.

You can specify the order in which the **show** command is executed by entering the *position* argument; the valid values are from 1–15. The *position* argument is the number of the **show** command in the system information logging index.

To specify the **show** commands whose output is logged in a file, perform this task in privileged mode:

	Task	Command
Step 1	Specify the **show** commands whose output is logged.	**set system info-log command** {*command_string*} [*position*]
Step 2	Verify that system information logging is enabled.	**show system info-log**

This example shows how to specify a **show** command and verify that it is included in the system information logging:

```
Console> (enable) set system info-log command $show version$
System command was successfully added to the list.
Console> (enable) set system info-log command $show module$
System command was successfully added to the list.
Console> (enable) set system info-log command $show environment$
System command was successfully added to the list.
Console> (enable) set system info-log command $show config$
System command was successfully added to the list.
Console> (enable) show system info-log
System Logging  Host            File                                Interval
--------------  --------------  ----------------------------------  --------
Enabled         -               tftp:sysinfo                        1440
Index        System Command
-----        ------------------------------------------------------------
    1        show version
    2        show module
    3        show environment
    4        show config
Console> (enable)
```

Specifying How Often System Information Logging Occurs

You can specify the amount of time that elapses between the occurrences of system information logging. Specify the amount of time in minutes; the valid values are between 1–35000 minutes (25 days). By default, the amount of time between the logging occurrences is 1440 minutes (1 day).

To specify the amount of time and verify the time interval, perform this task in privileged mode:

	Task	Command
Step 1	Specify the amount of time between the occurrences of system information logging.	**set system info-log interval** *mins*
Step 2	Verify the time interval.	**show system info-log**

This example shows how to specify the amount of time and verify the time interval:

```
Console> (enable) set system info-log interval 4320
Successfully set system information logging interval to 4320 minutes.
Console> (enable) show system info-log
System Logging  Host             File                                Interval
--------------  ---------------  ----------------------------------  -------
Enabled         -                tftp:sysinfo                        4320
Index        System Command
-----        ------------------------------------------------------------
    1        show config
    2        show version
    3        show module
    4        show environment
Console> (enable)
```

Specifying the Filename and Server for System Information Logging

You can specify the filename and the server for system information logging. If you do not specify a path for the file, the default directory for TFTP is tftpboot, and the default directory for rcp is the user's home directory.

To specify the filename and the server for system information logging, perform this task in privileged mode:

	Task	Command
Step 1	Specify the filename and the server for system information logging.	**set system info-log** {**tftp** | **rcp** *username*} *host filename*
Step 2	Verify the time interval.	**show system info-log**

This example shows how to specify the filename and the server and verify the configuration:

```
Console> (enable) set system info-log rcp hcavende 10.5.2.10 sysinfo
Successfully set the system information logging file to rcp:sysinfo
Console> (enable) show system info-log
System Logging  Host            File                                Interval
--------------  --------------  ----------------------------------  --------
Enabled         10.5.2.10       rcp:sysinfo                         4320
Index        System Command
-----        ------------------------------------------------------------
    1        show config
    2        show version
    3        show module
    4        show environment
Console> (enable)
```

Clearing a show Command from System Information Logging

To clear all the **show** commands or a specific **show** command from system information logging and verify its removal, perform this task in privileged mode:

	Task	Command
Step 1	Clear a **show** command from system information logging.	**clear system info-log command** {**all** | *index*}
Step 2	Verify the removal of the **show** command.	**show system info-log**

This example shows how to clear the **show** command number 1 from the system information logging index:

```
Console> (enable) clear system info-log command 2
Successfully cleared the configured command.
Console> (enable) show system info-log
System Logging  Host            File                                Interval
--------------  --------------  ----------------------------------  --------
Enabled         10.5.2.10       rcp:sysinfo                         4320
Index        System Command
-----        ------------------------------------------------------------
    1        show config
    2        show module
    3        show environment
Console> (enable)
```

Clearing the Configuration of System Information Logging

To clear the configuration of system information logging and restore the default settings, perform this task in privileged mode:

	Task	Command
Step 1	Clear the configuration of system information logging.	**clear config sysinfo-log**
Step 2	Verify that the configuration is cleared.	**show system info-log**

This example shows how to clear the configuration of system information logging and restore the defaults:

```
Console> (enable) clear config sysinfo-log
Successfully cleared the system information logging configuration.
Console> (enable) show system info-log
System Logging  Host             File                                Interval
--------------  ---------------  ----------------------------------  --------
Disabled        -                tftp:sysinfo                        1440
Index        System Command
-----        ------------------------------------------------------------
Console> (enable)
```

Disabling System Information Logging

To disable system information logging, perform this task in privileged mode:

	Task	Command
Step 1	Disable system information logging.	**set system info-log disable**
Step 2	Verify that system information logging is disabled.	**show system info-log**

This example shows how to disable system information logging and verify that it is disabled:

```
Console> (enable) set system info-log disable
Successfully disabled system information logging.
Console> (enable) show system info-log
System Logging  Host             File                                Interval
--------------  ---------------  ----------------------------------  --------
Disabled        -                tftp:sysinfo                        1440
Index        System Command
-----        ------------------------------------------------------------
Console> (enable)
```

TCL Scripting

Tool Command Language (TCL) is a simple, programmable, text-based language that allows you to write the command procedures that expand the capabilities of the built-in set of commands. TCL is used with interactive programs such as text editors, debuggers, illustrators, and shells. The Catalyst 6500 series switch software supports TCL version 7.4.

TCL is open source code. You can find information about the TCL commands and about using, licensing, or programming in TCL at this URL:

http://www.tcl.tk

Table 22-3 lists the supported TCL commands. The commands with a *t* prefix (**tformat**, **trename**, **tset**, and **tswitch**) have been customized from the standard TCL command set to avoid conflicts with the Catalyst 6500 series switch software. The following two commands have been specifically added to the software:

- **auto answer** {**on** | **off**}

  When set to **on**, the TCL shell will answer *yes* if prompted by the switch for a yes or no answer. The default setting is **off**.

- **echo** {**on** | **off**}

  When set to **off**, the output from the switch commands is not displayed on the screen. The default is **on**.

Table 22-3 TCL Commands

append	array	auto answer	break
case	catch	concat	continue
echo	error	eval	expr
for	foreach	global	if
incr	info	join	lappend
lindex	linsert	list	llength
lrange	lreplace	lsearch	lsort
proc	puts	regexp	regsub
return	scan	source	split
string	subst	tformat	trename
tset	tswitch	unset	uplevel
upvar	while		

Entering TCL Commands

You must enter the TCL commands using the TCL shell. To open the TCL shell, perform this task in privileged mode:

Task	Command
Open the TCL shell.	**tclsh**

This example shows how to open the TCL shell:

```
Console> (enable) tclsh
Console> (tclsh)(enable)
```

To close the TCL shell, perform this task in privileged mode:

Task	Command
Close the TCL shell.	**tclquit**

This example shows how to close the TCL shell:

```
Console> (enable) tclquit
Console> (enable)
```

CHAPTER 23

Configuring Redundancy

This chapter describes how to configure the redundant supervisor engines and how to configure redundancy on the Multilayer Switch Feature Cards (MSFCs) on the Catalyst 6500 series switches.

This chapter consists of these sections:

Note For information on configuring MSFC redundancy using Cisco nonstop forwarding (NSF) with stateful switchover (SSO), see Chapter 24, "Configuring NSF with SSO MSFC Redundancy."

Caution The dual MSFCs in a single chassis are designed to be used in redundant mode only and must have identical configurations. See the "MSFC Redundancy" section on page 23-21 for detailed information.

We do not support configurations where the MSFCs are not configured identically.

Note Except where specifically differentiated, the information and procedures in this chapter apply to Supervisor Engine 32 with PFC3B/PFC3BXL, Supervisor Engine 720 with PFC3A/PFC3B/PFC3BXL, Supervisor Engine 2 with PFC2, and Supervisor Engine 1 with PFC.

Note The term *MSFC* is used throughout this publication to refer to MSFC, MSFC2, MSFC2A, and MSFC3 except where specifically differentiated.

For more information about installing the redundant Catalyst 6500 series supervisor engines, refer to the *Catalyst 6500 Series Switch Module Installation Guide*. For syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

Understanding How Supervisor Engine Redundancy Works

Note The redundant supervisor engines must be of the same type with the same model feature card. The WS-X6K-SUP1-2GE and the WS-X6K-SUP1A-2GE (both without PFCs) are compatible for redundancy. For supervisor engines with PFCs, the PFCs must be identical for redundancy (two PFCs, two PFC2s, two PFC3As, two PFC3Bs, or two PFC3BXLs).

When you install two supervisor engines, the first supervisor engine to come online becomes the active module; the second supervisor engine goes into standby mode. All administrative and network management functions, such as SNMP, command-line interface (CLI) console, Telnet, Spanning Tree Protocol (STP), Cisco Discovery Protocol (CDP), and VLAN Trunking Protocol (VTP) are processed on the active supervisor engine.

On the standby supervisor engine, the console port is inactive, the module status shows as "standby," and the status for the uplink ports is shown normally.

For Supervisor Engine 1 and Supervisor Engine 2, you must install the redundant supervisor engines in slots 1 and 2 of the chassis. The Supervisor Engine 720 and Supervisor Engine 32 slot requirements are as follows: With a 3-slot chassis, install Supervisor Engine 720 and Supervisor Engine 32 in either slot 1 or 2. With a 6-slot or a 9-slot chassis, install Supervisor Engine 720 and Supervisor Engine 32 in either slot 5 or 6. With a 13-slot chassis, install Supervisor Engine 720 and Supervisor Engine 32 in either slot 7 or 8. You must install redundant supervisor engines in both slots.

The redundant supervisor engines are hot swappable. The system continues to operate with the same configuration after switching over to the redundant supervisor engine.

Note To allow you to control the booting of each supervisor engine separately, the configuration registers are not synchronized between the supervisor engines.

Note The switchover time from the active supervisor engine to the standby supervisor engine does not include the spanning-tree convergence time.

At power-up, both supervisor engines run initial module-level diagnostics. Assuming that both supervisor engines pass this level of diagnostics, the two supervisor engines communicate over the backplane, allowing them to cooperate during the switching-bus diagnostics. The supervisor engine in slot 1 becomes active, and the supervisor engine in slot 2 enters standby mode. If the software versions of the two supervisor engines are different, or if the NVRAM configuration of the two supervisor engines is different, the active supervisor engine automatically downloads its software image and configuration to the standby supervisor engine.

Note The terms *slot 1* and *slot 2* refer to the redundant supervisor engines. As noted earlier, Supervisor Engine 720 and Supervisor Engine 32 have different slot requirements.

If the background diagnostics on the active supervisor engine detect a major problem or an exception occurs, the active supervisor engine resets. The standby supervisor engine detects that the active supervisor engine is no longer running and becomes active. The standby supervisor engine can detect if the active supervisor engine is not functioning and can force a reset, if necessary. If the reset supervisor engine comes online again, it enters standby mode.

If you hot insert a second supervisor engine, the second module communicates with the active supervisor engine after completing its initial module-level diagnostics. Because the active supervisor engine is already switching traffic on the backplane, no switching-bus diagnostics are run for the second supervisor engine because running diagnostics can disrupt the normal traffic. The second supervisor engine immediately enters standby mode. The active supervisor engine downloads the software image and configuration to the standby supervisor engine, if necessary.

The supervisor engines use two flash images: the *boot image* and the *run-time image*. The boot image filename, which is specified in the BOOT environment variable, is stored in NVRAM. The run-time image is the boot image that the ROM monitor uses to boot the supervisor engine. After the system boots, the run-time image resides in dynamic RAM (DRAM).

When you power up or reset a switch with the redundant supervisor engines, synchronization occurs to ensure that the run-time and boot images on the standby supervisor engine are the same as the images on the active supervisor engine.

The supervisor engines can have different run-time and boot images. If the boot image and the run-time image are the same, and you change the BOOT environment variable or overwrite or destroy the current boot image on the flash device that was used to boot the system, the run-time and boot images will differ. Whenever you reconfigure the boot image, the active supervisor engine synchronizes its current boot image with the standby supervisor engine.

The boot image is read directly into the flash file system. You can perform operations (such as **copy**, **delete**, **undelete**, and so on) on the files that are stored on flash memory devices, and you can store the boot image of the active supervisor engine in the standby supervisor engine bootflash. For more information about using the flash file system, see Chapter 26, "Working With the Flash File System."

Supervisor Engine 1 and Supervisor Engine 2 have a Flash PC card (PCMCIA) slot (slot0) in addition to the onboard flash memory; this slot can hold a Flash PC card that can store additional boot images. The keywords for the slot are **slot0:** for linear flash devices and **disk0:** for ATA flash devices.

Note The term *Flash PC card* is used throughout this publication in place of the term *PCMCIA card*.

Supervisor Engine 720 has two CompactFlash Type II slots. The CompactFlash Type II slots support the CompactFlash Type II Flash PC cards. The keywords for the slots on the active Supervisor Engine 720 are **disk0:** and **disk1:**. The keywords for the slots on a redundant Supervisor Engine 720 are **slavedisk0:** and **slavedisk1:**. Supervisor Engine 32 has one CompactFlash Type II slot. The CompactFlash Type II slot supports the CompactFlash Type II Flash PC cards. The keyword for the slot on the active Supervisor Engine 32 is **disk0:**. The keyword for the slot on a redundant Supervisor Engine 32 is **slavedisk0:**.

Because you can store multiple boot images, you must specify the name of the boot file image and the location of the image file in the flash file system to boot and synchronize properly. For information about how to specify the name and location of the boot image, see Chapter 25, "Modifying the Switch Boot Configuration."

In the synchronization process, the active supervisor engine checks the standby supervisor engine run-time image to make sure that it matches its own run-time image. The active supervisor engine checks three conditions:

- If it needs to copy its boot image to the standby supervisor engine
- If the standby supervisor engine bootstring needs to be changed
- If the standby supervisor engine needs to be reset

The following section describes the conditions that can initiate the flash synchronization. For examples of how the system synchronizes the supervisor engine flash images with various configurations, see the "Supervisor Engine Synchronization Examples" section on page 23-15.

Configuring Redundant Supervisor Engines on the Switch

These sections describe how to configure the redundant supervisor engines:

Synchronization Process Initiation

These conditions initiate the synchronization of the run-time and boot images on the active and standby supervisor engines:

- Time stamp mismatch between the run-time images on the active and standby supervisor engines—The active supervisor engine synchronizes its run-time image with the standby supervisor engine if the time stamps of their respective run-time images differ when the system is booted or reset.
- Time stamp mismatch between the boot images on the active and standby supervisor engines—The active supervisor engine synchronizes its boot image with the standby supervisor engine if the time stamps of their respective boot images differ when the system is booted or reset, or if you change the BOOT environment variable.
- Current boot image overwritten—If you overwrite the current boot image that is stored on one of the flash devices, the file system management module detects this event and initiates synchronization. The active supervisor engine copies its new boot image to the standby supervisor engine.
- BOOT environment variables changed—If you change the BOOT environment variables to specify a different default boot image, the active supervisor engine initiates the boot-image synchronization. The NVRAM configuration module detects this event and calls the flash synchronization function with the next probable boot filename by looking at the boot configuration parameter.
- Flash PC cards with the same boot-image filename—If you change the flash device on either the active or standby supervisor engine and the new flash device contains a boot image that has the same name (but a different time stamp) as the boot image from the previous flash device, the flash file management module initiates synchronization.
- Current run-time image deleted—If you delete the current run-time image from the flash device, the flash file management module prompts you to verify that you want to delete the current run-time image. If you confirm the deletion, the flash file management module initiates flash synchronization and informs the NVRAM configuration module of the change. The NVRAM configuration module examines the BOOT environment variable to determine the next probable image to boot and calls the flash synchronization function using the new image name.

Redundant Supervisor Engine Configuration Guidelines and Restrictions

These conditions and events can cause the synchronization of the images between the redundant supervisor engines to fail or to produce unexpected results:

- Downloading a new image to the active supervisor engine

  When you download a new image to the active supervisor engine, it is copied to the file system (in bootflash or on a Flash PC card). Because you may or may not have configured this image as the boot image, the newly downloaded image is not copied to the standby supervisor engine automatically.

  To initiate the synchronization function between the active and standby supervisor engines, you must configure this newly downloaded image as the boot image on the active supervisor engine. Synchronization occurs when you change the boot variable. To run the new image, you must reset the system.

- Unable to find the current run-time image

  If the active supervisor engine is unable to find the current run-time image on any of the flash devices, it signals an error condition. If you insert or reset the standby supervisor engine, flash synchronization does not occur. In addition, the STATUS LED on the standby supervisor engine turns red and the system generates a syslog error message.

- Active supervisor engine in slot 2

  When the active supervisor engine is in slot 2, the standby supervisor engine is in slot 1. If you change the configuration to specify a new boot image and then reset the system, the supervisor engine in slot 1 becomes the active supervisor engine and loads its default boot image, canceling the configuration changes that you have just made. To avoid this problem, the switch prompts you for flash synchronization as soon as you change the boot file configuration.

Verifying the Standby Supervisor Engine Status

You can verify the status of the standby supervisor engine by using the CLI commands described in this section.

Note The **show module** command output provides information about the installed daughter cards. The **show test** command provides information about the onboard application-specific integrated circuits (ASICs).

To verify the status of the standby supervisor engine, perform one or more of these tasks:

Task	Command
Show the status of the standby supervisor engine.	**show module** [*mod*]
Show the state of the standby supervisor engine uplink ports.	**show port** [*mod*[*/port*]]
Show diagnostic test results for the standby supervisor engine.	**show test** [*mod*]

This example shows how to check the status of the standby supervisor engine by entering the **show module** and **show test** commands:

```
Console> (enable) show module 2
Mod Slot Ports Module-Type               Model               Status
--- ---- ----- ------------------------- ------------------- --------
2   2    2     1000BaseX Supervisor      WS-X6K-SUP1-2GE     ok

Mod Module-Name         Serial-Num
--- ------------------- -----------
2                       SAD02330231

Mod MAC-Address(es)                        Hw     Fw          Sw
--- -------------------------------------- ------ ----------  -----------------
2   00-e0-14-0e-f5-6c to 00-e0-14-0e-f5-6d 0.404  4.2(2038)  4.2(0.24)VAI50
    00-e0-14-0e-f5-6e to 00-e0-14-0e-f5-6f
    00-10-7b-bb-2b-00 to 00-10-7b-bb-2e-ff

Mod Sub-Type            Sub-Model           Sub-Serial  Sub-Hw
--- ------------------- ------------------- ----------- ------
2   L2 Switching Engine WS-F6020            SAD02350211 0.101
Console> (enable)

Console> (enable) show test 2
Module 2 : 2-port 1000BaseX Supervisor
Network Management Processor (NMP) Status: (. = Pass, F = Fail, U = Unknown)
  ROM:  .   Flash-EEPROM: .   Ser-EEPROM: .   NVRAM: .   EOBC Comm: .

Line Card Status for Module 1 : PASS

Port Status :
  Ports 1  2
  -----------
        .  .
Line Card Diag Status for Module 2  (. = Pass, F = Fail, N = N/A)

 Module 2
  Cafe II Status :
        NewLearnTest:             .
        IndexLearnTest:           .
        DontForwardTest:          .
        DontLearnTest:            .
        ConditionalLearnTest:     .
        BadBpduTest:              .
        TrapTest:                 .
 Loopback Status [Reported by Module 2] :
  Ports 1  2
  -----------
        .  .
Console> (enable)
```

Forcing a Switchover to the Standby Supervisor Engine

You can force a switchover to the standby supervisor engine by resetting the active supervisor engine.

Note Resetting the active supervisor engine disconnects any open Telnet sessions.

To force a switchover to the standby supervisor engine, perform this task in privileged mode:

Task	Command
Reset the active supervisor engine (where *mod* is the number of the active supervisor engine).	**reset** *mod*

In addition, you can also force a switchover to the standby supervisor engine by setting the CISCO-STACK-MIB moduleAction variable to reset(2) on the active supervisor engine. When the switchover occurs, the system sends a standard SNMP warm-start trap to the configured trap receivers.

This example shows the console output on the active supervisor engine when you force a switchover from the active to the standby supervisor engine:

```
Console> (enable) reset 1
This command will force a switch-over to the standby Supervisor module.
Do you want to continue (y/n) [n]? y
Console> (enable) 12/07/1998,17:04:39:SYS-5:Module 1 reset from Console//

System Bootstrap, Version 3.1(2)
Copyright (c) 1994-1997 by cisco Systems, Inc.

System Bootstrap, Version 3.1(2)
Copyright (c) 1994-1997 by cisco Systems, Inc.
Presto processor with 32768 Kbytes of main memory

Autoboot executing command: "boot bootflash:cat6000-sup.5-4-1a.bin"
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
Uncompressing file:  ###########################################################

System Power On Diagnostics
NVRAM Size .. .................512KB
ID Prom Test .................Passed
DPRAM Size ...................16KB
DPRAM Data 0x55 Test .........Passed
DPRAM Data 0xaa Test .........Passed
DPRAM Address Test ...........Passed
Clearing DPRAM ...............Done
System DRAM Memory Size .......32MB
DRAM Data 0x55 Test ..........Passed
DRAM Data 0xaa Test ..........Passed
DRAM Address Test  ...........Passed
Clearing DRAM ................Done
EARLII .......................Present
EARLII RAM Test ..............Passed
EARL Serial Prom Test .........Passed
Level2 Cache .................Present
Level2 Cache test.............Passed

Boot image: bootflash:cat6000-sup.5-4-1a.bin
Downloading epld sram device please wait ...
Programming successful for Altera 10K50 SRAM EPLD
This module is now in standby mode.
Console is disabled for standby supervisor
```

This example shows the console output on the standby supervisor engine when you force a switchover from the active to the standby supervisor engine:

```
Cisco Systems Console

Enter password:
12/07/1998,17:04:43:MLS-5:Multilayer switching is enabled
12/07/1998,17:04:43:MLS-5:Netflow Data Export disabled
12/07/1998,17:04:44:SYS-5:Module 2 is online
12/07/1998,17:04:45:SYS-5:Module 5 is online
12/07/1998,17:04:45:SYS-5:Module 7 is online
12/07/1998,17:04:45:SYS-5:Module 3 is online
12/07/1998,17:04:52:MLS-5:Route Processor 172.20.52.6 added
12/07/1998,17:05:10:SYS-5:Module 8 is online
12/07/1998,17:05:14:SYS-5:Module 9 is online
12/07/1998,17:05:22:SYS-5:Module 4 is online
12/07/1998,17:06:13:SYS-5:Module 1 is in standby mode
Supervisor image synchronization process will start in 10 seconds
12/07/1998,17:06:37:SYS-5:Ports on standby supervisor(Module 1) are UP
12/07/1998,17:06:41:SYS-5:Active supervisor is synchronizing the NMP image.
12/07/1998,17:06:44:SYS-5:The active supervisor has synchronized the NMP image.

Console>
```

High Availability

High availability allows you to minimize the switchover time from the active supervisor engine to the standby supervisor engine if the active supervisor engine fails.

Prior to this feature, fast switchover ensured that a switchover to the standby supervisor engine happened quickly. However, with fast switchover, because the state of the switch features before the switchover was unknown, you had to reinitialize and restart all the switch features when the standby supervisor engine assumed the active role.

High availability removes this limitation; high availability allows the active supervisor engine to communicate with the standby supervisor engine, keeping feature protocol states synchronized. Synchronization between the supervisor engines allows the standby supervisor engine to take over in the event of a failure.

In addition, high availability provides a *versioning* option that allows you to run the different software images on the active and standby supervisor engines.

These features are discussed in these sections:

High-Availability Overview

For high availability, a system database is maintained on the active supervisor engine and the updates are sent to the standby supervisor engine for any change of data in the system database. The active supervisor engine communicates and updates the standby supervisor engine when any state changes occur, ensuring that the standby supervisor engine knows the current protocol state of the supported features. The standby supervisor engine knows the current protocol states for all modules, ports, and VLANs; the protocols can initialize with this state information and start running immediately.

The active supervisor engine controls the system bus (backplane), sends and receives the packets to and from the network, and controls all modules. The protocols run on the active supervisor engine only.

The standby supervisor engine is isolated from the system bus and does not switch packets, but it *does* receive the packets from the switching bus to learn and populate its Layer 2 forwarding table for Layer 2-switched flows. In addition, the standby supervisor engine receives the packets from the switching bus to learn and populate tables for the Layer 3-switched flows. The standby supervisor engine does not participate in forwarding any packets and does not communicate with any modules.

If you enable high availability when the standby supervisor engine is running, the image version compatibility is checked and if found compatible, the database synchronization is started. High-availability compatible features continue from the saved states on the standby supervisor engine after a switchover.

When you disable high availability, the database synchronization is not done and all features must restart on the standby supervisor engine after a switchover.

If you change high availability from enabled to disabled, synchronization from the active supervisor engine is stopped and the standby supervisor engine discards all the current synchronization data.

If you change high availability from disabled to enabled, synchronization from the active to the standby supervisor engine is started (if the standby supervisor engine is present and its image version is compatible).

NVRAM synchronization occurs regardless of high availability being enabled or disabled (if there are compatible NVRAM versions on the two supervisor engines).

If you do not install a standby supervisor engine during the system bootup, the active supervisor engine detects this and the database updates are not queued for synchronization. Similarly, when you reset or remove the standby supervisor engine, the synchronization updates are not queued and any pending updates in the synchronization queue are discarded. When you hot insert or restart a second supervisor engine that becomes the standby supervisor engine, the active supervisor engine downloads the entire system database to the standby supervisor engine. Only after this global synchronization is completed, the active supervisor engine queues and synchronizes the individual updates to the standby supervisor engine.

Note When you hot insert or restart a second supervisor engine, it might take a few minutes for the global synchronization to complete.

High-Availability Supported Features

The high-availability features for the Catalyst 6500 series switch are classified into three categories (see Table 23-1):

- Supported features—High availability is fully supported; the feature's database is synchronized from the active supervisor engine to the standby supervisor engine.
- Compatible features—High availability is not supported; the feature's database is not synchronized from the active supervisor engine to the standby supervisor engine. However, you can enable the compatible features when you enable high availability.
- Incompatible features—High availability is not supported. The feature's database is not synchronized from the active supervisor engine to the standby supervisor engine. You cannot enable the incompatible features if you enable high availability, and you cannot enable high availability if you enable these incompatible features.

Table 23-1 High-Availability Feature Support

Supported Features	Compatible Features	Incompatible Features
CEF	ASLB	Dynamic VLAN
COPS-DS	CDP	GVRP
COPS-PR	GMRP	Protocol filtering
DTP	IGMP snooping	
EtherChannel	RMON	
Cisco IOS ACLs	RSVP	
MLS	SNMP	
PAgP	Telnet sessions	
QoS	UplinkFast	
SPAN	VTP pruning	
STP		
Trunking		
UDLD		
VACLs		
VTP		
Port security		
802.1x		

High-Availability Configuration Guidelines

This section describes the guidelines for configuring high availability:

- High availability does not preserve the routing table entries on the active MSFC because high availability is not run on the Cisco IOS software. However, you can configure both MSFCs on the active and standby supervisor engines with the same configuration to preserve the routing table entries across the active and standby MSFCs. You can then configure HSRP on the MSFCs to provide automatic routing backup. See the "MSFC Redundancy" section on page 23-21 for detailed information.
- The timers and statistics are not synchronized from the active to the standby supervisor engine.
- The MLS flows are preserved from the active supervisor engine to the standby supervisor engine.
- On the 802.1X ports, only the authorized and unauthorized states are synchronized from the active to the standby supervisor engine. The ports in any other state are initialized or restarted after switchover occurs.
- The 802.1X record updates are minimized by grouping similar types of updates into a single record. The active supervisor engine sends the record to the standby supervisor engine when a variable in the record changes.
- The 802.1X reauthentication timers for the authorized ports restart after the switchover occurs.
- The port security statistics are not synchronized from the active to the standby supervisor engine.
- When you enable high availability or hot insert a standby supervisor engine on a switch that has secure ports, all the per-port and MAC-related information is synchronized from the active to the standby supervisor engine.

Versioning Overview

With high-availability versioning enabled, you can have two different but compatible images on the active and standby supervisor engines. The active supervisor engine exchanges image version information with the standby supervisor engine and determines whether the images are compatible for enabling high availability. If the active and standby supervisor engines are not running compatible image versions, you cannot enable high availability.

Image versioning is supported in supervisor engine software releases 5.4(1) and later. With versioning enabled, high availability is fully supported with the active and standby supervisor engines running different images as long as the images are compatible. The only fully compatible images are as follows:

Note There is no software image version compatibility in the 8.x software release train. This includes major releases such as 8.1(x) to 8.2(x) to 8.3(x) and so on. This also includes subreleases such as 8.1(1) to 8.1(2), 8.2(1) to 8.2(2) and so on.

- Supervisor Engine 1
  - 5.5(3) and 5.5(4)
  - 6.1(3) and 6.1(4)
  - 6.2(2) and 6.2(3)
  - 6.3(2) and 6.3(3)
  - 6.3(4) and 6.3(5)
  - 6.3(6) and 6.3(7)

- Supervisor Engine 2
  - 6.1(3) and 6.1(4)
  - 6.2(2) and 6.2(3)
  - 6.3(2) and 6.3(3)

Images that are compatible with all modules except Gigabit Ethernet switching modules are as follows:

- Supervisor Engine 1
  - 5.4(3) and 5.4(4)
  - 5.5(3) and 5.5(5)
  - 5.5(4) and 5.5(5)

Images that are compatible with Gigabit Ethernet switching modules but not compatible with 10/100BASE-T modules are as follows:

- Supervisor Engine 1
  - 5.5(6a) and 5.5(7)

Images that are compatible with all modules except the SFM/SFM2 and fabric-enabled modules are as follows:

- Supervisor Engine 2
  - 6.3(4) and 6.3(5)
  - 6.3(6) and 6.3(7)

Note Attempting to run incompatible image versions could result in configuration loss.

Note When you install two supervisor engines, the first supervisor engine to come online becomes the active module; the second supervisor engine goes into standby mode. If two supervisor engines are installed in your system, at power up the supervisor engine in slot 1 becomes active, and the supervisor engine in slot 2 enters standby mode. If the software versions of the two supervisor engines are different, or if the NVRAM configuration of the two supervisor engines is different, and if you do not enable versioning, the active supervisor engine automatically downloads its software image and configuration to the standby supervisor engine.

CLI Commands

This section describes the CLI commands for high availability and versioning.

Enabling or Disabling High Availability

High availability is disabled by default. To enable or disable high availability, perform this task in privileged mode:

Task	Command
Enable or disable high availability.	**set system highavailability** {**enable** | **disable**}

This example shows how to enable high availability:

```
Console> (enable) set system highavailability enable
System high availability enabled.
Console> (enable)
```

This example shows how to disable high availability:

```
Console> (enable) set system highavailability disable
System high availability disabled.
Console> (enable)
```

Enabling or Disabling High-Availability Versioning

High-availability versioning is disabled by default. To enable or disable high-availability versioning, perform this task in privileged mode:

Task	Command
Enable or disable high-availability versioning.	**set system highavailability versioning** {**enable** | **disable**}

This example shows how to enable high-availability versioning:

```
Console> (enable) set system highavailability versioning enable
Image versioning enabled.
Console> (enable)
```

This example shows how to disable high-availability versioning:

```
Console> (enable) set system highavailability versioning disable
Image versioning disabled.
Console> (enable)
```

Showing High-Availability Settings and Operational Status

The **show system highavailability** command displays the following:

- High-availability setting (enabled or disabled)
- Versioning setting (enabled or disabled)
- High-availability operational status (based on whether the standby supervisor engine is present and operational). The operational status field displays one of the following:
  - OFF (high-availability-not-enabled): The high availability option in NVRAM is disabled.
  - OFF (standby-supervisor-not-present): The standby supervisor engine is not installed.
  - OFF (standby-supervisor-image-incompatible): The standby supervisor engine is running a different image than the active supervisor engine and it is not version compatible (the versioning option in NVRAM is enabled). No synchronization is done (even a configuration change in NVRAM on the active supervisor engine cannot be propagated to the standby supervisor engine because of the version incompatibility).
  - OFF (standby-supervisor-image-nvram-only-compat): The standby supervisor engine is running a different image than the active supervisor engine (the versioning option in NVRAM is enabled) and the image is only NVRAM compatible (a configuration change in NVRAM on the active supervisor engine is propagated to the standby supervisor engine). However, high availability cannot be supported.

- OFF (standby-supervisor-not-operational-yet): The standby supervisor engine is detected but is not operational (not online yet).
- OFF (high-availability-not-operational-yet): The standby supervisor engine is operational (online), but high availability is not operational yet (when the system is booted from reset, it takes a few minutes before high availability is operational).
- ON: High availability is operational. The active supervisor engine's features have started queuing their state changes for synchronizing to the standby supervisor engine.

To display the high-availability configuration and operational states, perform this task:

Task	Command
Display the high-availability configuration and operational states.	**show system highavailability**

This example shows how to display the high-availability configuration and operational states:

```
Console> (enable) show system highavailability
Highavailability: disabled
Highavailability versioning: disabled
Highavailability Operational-status: OFF (high-availability-not-enabled)
Console> (enable)
```

This example shows how to enable high availability:

```
Console> (enable) set system highavailability enable
System high availability enabled.
Console> (enable)

Console> (enable) show system highavailability
Highavailability: enabled
Highavailability versioning: disabled
Highavailability Operational-status: ON
Console> (enable)
```

Loading a Different but Compatible Image on the Standby Supervisor Engine

Use this procedure to load a new image on the standby supervisor engine that is different from the image on the active supervisor engine. From the active supervisor engine console port, perform these steps (active supervisor engine is in slot 1):

Step 1 Enable high availability versioning.

```
Console> (enable) set system highavailability versioning enable
System high availability enabled.
Console> (enable)
```

Step 2 Download the new image to the active supervisor engine bootflash.

```
Console> (enable) copy tftp:image2.bin bootflash
IP address or name of remote host []? 172.20.52.3

8763532 bytes available on device bootflash, proceed (y/n) [n]? y
.
.
.
Console> (enable)
```

Step 3 Copy the new image to the standby supervisor engine bootflash.

```
Console> (enable) copy bootflash:image2.bin 2/bootflash:

5786532 bytes available on device bootflash, proceed (y/n) [n]? y
.
.
.
Console> (enable)
```

Step 4 Modify the BOOT environment variable so that the standby supervisor engine boots the new image.

```
Console> (enable) set boot system flash bootflash:image2.bin prepend 2
BOOT variable = bootflash:image2.bin,1;slot0:image1.bin,1
Console> (enable)
```

Step 5 To boot the new image, reset the standby supervisor engine.

```
Console> (enable) reset 2
This command will reset the system.
Do you want to continue (y/n) [n]? y
.
.
.
Console> (enable)
```

Configuring Supervisor Engine Redundancy Using NSF with SSO

Cisco NSF works with SSO to minimize the amount of time that a network is unavailable to its users following a switchover while continuing to forward the IP packets.

For information about configuring NSF with SSO, refer to "Configuring Supervisor Engine Redundancy using NSF with SSO" in the *Catalyst 6500 Series Cisco IOS Software Configuration Guide, 12.2SX* at this URL:.
http://www.cisco.com/en/US/docs/switches/lan/catalyst6500/ios/12.2SXF/native/configuration/guide/nsfsso.html.

Supervisor Engine Synchronization Examples

These sections explain what happens when the synchronization function encounters certain conditions:

- Synchronizing the Run-Time Image with the Bootstring, page 23-16
- Synchronizing the Boot Images on the Active and Standby Supervisor Engines, page 23-18

Note In the following examples, the number **1** following the filename in the bootstring (for example, **bootflash:f1,1**) indicates the number of Trivial File Transfer Protocol (TFTP) boot retries that are attempted. However, the supervisor engine does not support TFTP booting. The number is included in these examples to be consistent with Cisco IOS conventions.

Note These examples are not intended to cover every possible condition.

Synchronizing the Run-Time Image with the Bootstring

This section contains four examples in which the active supervisor engine run-time image is synchronized with the standby supervisor engine.

Example 1: Run-time image not synchronized

The configuration for example 1 is as follows:

- The active supervisor engine configuration is as follows (if the image in the standby supervisor engine is identical to the image in the active supervisor engine, the output is the same):
  - Run-time image: `bootflash:f1`
  - Bootstring: `bootflash:f1,1`
  - Bootflash: `f1`
- The time stamp for **f1** on the active supervisor engine is the same as **f1** on the standby supervisor engine.
- The expected results are as follows:
  - The active supervisor engine **f1** image is not copied to the standby supervisor engine.
  - The standby supervisor engine bootstring is not modified.
  - The standby supervisor engine is not reset.

Example 2: File copied, bootstring changed, standby supervisor engine reset

The configuration for example 2 is as follows:

- The active supervisor engine configuration is as follows:
  - Run-time image: `bootflash:f1`
  - Bootstring: `bootflash:f1,1`
  - Bootflash: `f1`
- The standby supervisor engine configuration is as follows:
  - Run-time image: `bootflash:f2`
  - Bootstring: `bootflash:f2,1`
  - Bootflash: `f2`
- The time stamp for **f1** on the active supervisor engine is not the same as **f2** on the standby supervisor engine.
- The expected results are as follows:
  - The active supervisor engine copies **f1** to the standby supervisor engine and renames the file **RTSYNC_f1**.
  - The standby supervisor engine bootflash is modified to the following: **f2, RTSYNC_f1**.
  - The standby supervisor engine bootstring is modified to the following: **bootflash:RTSYNC_f1,1;f2,1;**.
  - The standby supervisor engine is reset.

Example 3: File not copied, bootstring changed, standby supervisor engine reset

The configuration for example 3 is as follows:

- The active supervisor engine configuration is as follows:
  - Run-time image: `bootflash:f1`
  - Bootstring: `bootflash:f1,1`
  - Bootflash: `f1`
- The standby supervisor engine configuration is as follows:
  - Run-time image: `bootflash:f2`
  - Bootstring: `bootflash:f2,1`
  - Bootflash: `f1,f2`
- The time stamp for **f1** on the active supervisor engine is the same as **f1** on the standby supervisor engine but is not the same as **f2** on the standby supervisor engine.
- The expected results are as follows:
  - The active supervisor engine run-time image is synchronized to the standby supervisor engine.
  - The active supervisor engine **f1** image is not copied to the standby supervisor engine.
  - The standby supervisor engine bootstring is modified to the following: **f1,1;f2,1;**.
  - The standby supervisor engine is reset.

Example 4: Oldest bootflash file deleted, bootflash squeezed

The configuration for example 4 is as follows:

- The active supervisor engine configuration is as follows:
  - Run-time image: `bootflash:f1`
  - Bootstring: `bootflash:f1,1`
  - Bootflash: `f1`
- The standby supervisor engine configuration is as follows:
  - Run-time image: `bootflash:f2`
  - Bootstring: `bootflash:f2,1;`
  - Bootflash: `f2, f3, f4` (less than 1 MB left on device)
- The time stamp for **f1** on the active supervisor engine is not the same as **f2** on the standby supervisor engine. The **f2** time stamp is older than **f3**, and the **f3** time stamp is older than **f4**.

- The expected results are as follows:
  - The active supervisor engine run-time image is synchronized with the standby supervisor engine.
  - The active supervisor engine attempts to copy its **f1** image to the standby supervisor engine.
  - Because there is not enough space on the standby supervisor engine bootflash, the redundant synchronization function finds the oldest file, deletes it, and squeezes bootflash.
  - The active supervisor engine copies the **f1** image to the standby supervisor engine and renames it **RTSYNC_f1**.
  - The standby supervisor engine bootflash is modified to the following: **f3, f4, RTSYNC_f1**.
  - The standby supervisor engine bootstring is modified to the following: **RTSYNC_f1,1;f2,1;**.
  - The standby supervisor engine is reset.

Synchronizing the Boot Images on the Active and Standby Supervisor Engines

This section contains four examples in which the bootstrings on the active and standby supervisor engines are synchronized.

Example 1: Unable to allocate the boot image

The configuration for this example is as follows:

- The active supervisor engine configuration is as follows:
  - Run-time image: `bootflash:f1`
  - Bootstring: `bootflash:f1,1;`
  - Bootflash: `f1`
- The standby supervisor engine configuration is as follows:
  - Run-time image: `bootflash:f1`
  - Bootstring: `bootflash:f1,1;`
  - Bootflash: `f1`
- The time stamp for **f1** on the active supervisor engine is the same as **f1** on the standby supervisor engine.
- The system attempts to modify the active supervisor engine bootstring to the following: **f2,1;**.
- The expected results are as follows:
  - The active supervisor engine is unable to allocate **f2**, causing the synchronization to fail.
  - An error is recorded in syslog.
  - The active supervisor engine **f1** image is not copied to the standby supervisor engine.
  - The standby supervisor engine bootstring is not modified.
  - The standby supervisor engine is not reset.

Example 2: File copied, bootflash modified, standby supervisor engine not reset

The configuration for this example is as follows:

- The active supervisor engine configuration is as follows:
  - Run-time image: `bootflash:f1`
  - Bootstring: `bootflash:f1,1;`
  - Bootflash: `f1,f2`
- The standby supervisor engine configuration is as follows:
  - Run-time image: `bootflash:f1`
  - Bootstring: `bootflash:f1,1;`
  - Bootflash: `f1`
- The time stamp for **f1** on the active supervisor engine is the same as **f1** on the standby supervisor engine.
- You modify the active supervisor engine bootstring to the following: **f2,1;**.
- The expected results are as follows:
  - The active supervisor engine copies its **f2** image to the standby supervisor engine and renames it **BTSYNC_f2**.
  - The standby supervisor engine bootflash is modified to the following: **f1, BTSYNC_f2**.
  - The standby supervisor engine bootstring is modified to the following: **bootflash:BTSYNC_f2,1;f1,1;**.
  - The standby supervisor engine is not reset.

Example 3: File not copied, bootstring modified, standby supervisor engine not reset

The configuration for this example is as follows:

- The active supervisor engine configuration is as follows:
  - Run-time image: `bootflash:f1`
  - Bootstring: `bootflash:f1,1;`
  - Bootflash: `f1,f2`
- The standby supervisor engine configuration is as follows:
  - Run-time image: `bootflash:f1`
  - Bootstring: `bootflash:f1,1;`
  - Bootflash: `f1,f2`
- The time stamp for **f1** on the active supervisor engine is the same as **f1** on the standby supervisor engine; the time stamp for **f2** on the active supervisor engine is the same as **f2** on the standby supervisor engine.
- The active supervisor engine bootstring is modified to the following: **f2,1; f1,1;**.

- The expected results are as follows:
  - The active supervisor engine **f1** image is not copied to the standby supervisor engine.
  - The standby supervisor engine bootstring is modified to the following: **bootflash:f2,1;bootflash:f1,1;**.
  - The standby supervisor engine is not reset.

Example 4: File copied, oldest file deleted, bootflash squeezed, bootstring modified, standby supervisor engine not reset

The configuration for this example is as follows:

- The active supervisor engine configuration is as follows:
  - Run-time image: `bootflash:f1`
  - Bootstring: `bootflash:f1,1;`
  - Bootflash: `f1,f2`
- The standby supervisor engine configuration is as follows:
  - Run-time image: `bootflash:f1`
  - Bootstring: `bootflash:f1,1;`
  - Bootflash: `f0,f1,f3` (less than 1 MB left on device)
- The time stamp for **f1** on the active supervisor engine is the same as **f1** on the standby supervisor engine. The time stamp for **f0** is older than **f1**, and the time stamp for **f1** is older than **f3**.
- The active supervisor engine bootstring is modified to the following: **bootflash:f2,1;bootflash:f1,1;**
- The expected results are as follows:
  - The active supervisor engine attempts to copy its **f2** image to the standby supervisor engine.
  - Because there is not enough space available on the standby supervisor engine bootflash, the redundant synchronization function finds the oldest file (**f0**), deletes it, and squeezes bootflash.
  - The active supervisor engine copies its **f2** image to the standby supervisor engine and renames it **BTSYNC_f2**.
  - The standby supervisor engine bootflash is modified to the following: **f1, f3, BTSYNC_f2**.
  - The standby supervisor engine bootstring is modified to the following: **bootflash:BTSYNC_f2,1;bootflash:f1,1;**.

MSFC Redundancy

MSFC redundancy is described in these sections:

- Dual MSFC Redundancy, page 23-21
- Single Router Mode Redundancy, page 23-43
- Manual-Mode MSFC Redundancy, page 23-49

Note For information on configuring MSFC redundancy using Cisco nonstop forwarding (NSF) with stateful switchover (SSO), see Chapter 24, "Configuring NSF with SSO MSFC Redundancy."

Note Single router mode redundancy is the only supported MSFC redundancy option for Supervisor Engine 720 and Supervisor Engine 32.

Dual MSFC Redundancy

Caution You must configure both MSFCs identically. Table 23-2 on page 23-22 summarizes the identical requirements and the exceptions for Layer 3 redundancy for a single switch chassis.

We do not support configurations where the MSFCs are not configured identically.

These sections describe how to configure MSFC redundancy:

- Hardware and Software Requirements, page 23-21
- Layer 3 Redundancy for a Single Chassis, page 23-22
- Routing Protocol Peering, page 23-23
- Access Control List Configuration, page 23-24
- Dual MSFC Operational Model for Redundancy and Load Sharing, page 23-25
- Understanding Failure Scenarios, page 23-26

Hardware and Software Requirements

To configure Layer 3 redundancy, you must have at least one of the following configurations:

- A single chassis with two identical supervisor engine daughter card configurations:
  - Supervisor Engine 1 with Policy Feature Card (PFC) and MSFC or MSFC2 (both supervisor engines must have the same type of MSFC)
  - Supervisor Engine 2 with PFC2 and MSFC2
- Two chassis with a supervisor engine in each—You must have at least one supervisor engine in each chassis. Each supervisor engine must be equipped with a PFC and an MSFC.

Note Each MSFC must be running the same release of Cisco IOS software.

Layer 3 Redundancy for a Single Chassis

In a single Catalyst 6500 series chassis, you can have the redundant supervisor engines, each with an MSFC. You can configure Hot Standby Router Protocol (HSRP) on the MSFCs to provide transparent default gateway redundancy for the IP hosts in the network. HSRP configuration can coexist with the IPX and AppleTalk configuration on the same interfaces.

If one MSFC fails, HSRP allows one MSFC (router) to assume the function automatically of the other MSFC. Combined with the high-availability feature of supervisor engine software release 5.4(1), this configuration provides an added level of redundancy for your network.

Caution You *must* configure both MSFCs identically. Table 23-2 summarizes the identical requirements and the exceptions for Layer 3 redundancy for a single switch chassis.

Table 23-2 Single Chassis Layer 3-Redundancy Requirements

Identical Requirements—Global and Interface Levels	Exceptions—Interface Level	Exceptions—Global Level
• Both MSFCs *must* have the following: – Same routing protocols – Same static routes – Same default routes – Same policy routes – Same VLAN interfaces – Same Cisco IOS ACLs[1, 2] • All interfaces *must* have the same administrative status	• HSRP standby commands • IP address commands[3] • IPX network[3]	• IP default-gateway • IPX internal-network • IPX default-route

1. The dynamic and reflexive ACLs, which are based on actual data flow, may be programmed by either MSFC.
2. In addition to defining the same ACLs on both MSFCs, you must also apply the ACLs to the same VLAN interfaces in the same direction on both MSFCs.
3. The IP or IPX addresses do not have to be identical on both MSFCs, but there *must* be an IP or IPX address configured on both MSFCs.

For information on specifying the alternate configurations for the interface and global level exceptions that are listed in Table 23-2, see the "alt Keyword Usage" section on page 23-36.

The redundant supervisor engines *must* have identical hardware (MSFC and PFC). See the "Hardware and Software Requirements" section on page 23-21 for more information.

Note For the MSFC and MSFC2 memory requirements, refer to the Release Notes for MSFC publication at this URL: http://www.cisco.com/en/US/products/hw/switches/ps708/prod_release_notes_list.html

Routing Protocol Peering

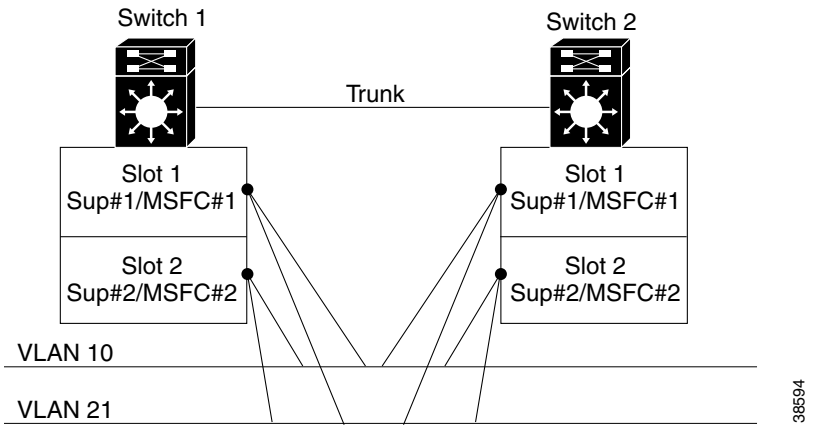
In a redundant supervisor engine and dual MSFC configuration, one supervisor engine is fully operational (active) and the other supervisor engine is in standby mode; however, both MSFCs are operational (in terms of programming the PFC on the active supervisor engine) and act as independent routers.

Note **PFC:** With the PFC, MLS entries can be associated with either MSFC (based on which MSFC routed the first packet). Only the PFC on the active supervisor engine switches the packets.

Note **PFC2:** With PFC2, only the designated MSFC programs the forwarding information base (FIB), the adjacency table, Cisco IOS software, and policy routing ACLs on the active supervisor engine. If you configure static routes or policy routing, you must have the *identical* configuration on both MSFCs. If you have a static route on the nondesignated MSFC that is not on the designated MSFC, that route *will not* be programmed in the PFC2.

Both MSFCs are operational from a routing protocol peering perspective. For example, if you have two MSFCs in a single Catalyst 6500 series switch chassis, each configured with interface VLAN 10 and VLAN 21, the MSFCs are peered to each other over these VLANs. Combined with a dual chassis and dual MSFC design for the same VLANs, each MSFC has 6 peers: its peer in the same chassis and the 2 MSFCs in the second chassis (3 in VLAN 10 and 3 in VLAN 21). See Figure 23-1.

Figure 23-1 Dual Chassis and Dual MSFC Peering



Although the MSFCs (from a peering perspective) act as independent routers, the two MSFCs in the chassis operate at the same time, have the same interfaces, and run the same routing protocols.

If you combine high availability on the supervisor engines with HSRP on the MSFCs, you have the following Layer 2 and Layer 3 redundancy mechanisms:

- Layer 2 redundancy for the supervisor engines (one active and one in standby)—If the active supervisor engine fails (the MSFC installed on it will also fail), both Layer 2 and Layer 3 functions roll over to the redundant supervisor engine and MSFC combination.
- Layer 3 redundancy and load sharing for the two MSFCs—If one MSFC fails, the other MSFC takes over almost immediately (using HSRP) without any Layer 2 disruption (the active supervisor engine continues to forward Layer 2 traffic).

The Layer 3 entries that are programmed by the failed MSFC on the active supervisor engine are used until they gracefully age out and are replaced by the Layer 3 entries that are populated by the newly active MSFC. Aging takes 4 minutes and allows the newly active MSFC to repopulate the MLS entries using its XTAG value, while concurrently hardware-switching flows that are yet to be aged. In addition, this process prevents a newly active MSFC from being overwhelmed with the initial flow traffic.

Note Each MSFC has its own XTAG value to identify itself as the MLS Route Processor. MSFC #1 (on the active supervisor engine) has an XTAG of 1, and MSFC #2 (on the standby supervisor engine) has an XTAG of 2.

Only Supervisor Engine 1 uses the XTAG values; XTAG values are not used on Supervisor Engine 2.

Caution For same-chassis Layer 3 redundancy to function as expected, the configuration on each MSFC *must* be the same (see Table 23-2 on page 23-22).

Note Table 23-2 lists the configuration exceptions. For example, in Figure 23-1, there are 4 MSFCs on VLAN 10; each MSFC has different IP addresses and HSRP priorities.

Access Control List Configuration

If you use the Cisco IOS access control lists (ACLs) on the MSFC, you *must* configure the ACLs on both MSFCs identically, globally, and at the interface level. Only the designated MSFC (the MSFC to come online first or the MSFC that has been online the longest) programs the PFC with ACL information.

The active supervisor engine's PFC multilayer switches the packets (CEF [Cisco Express Forwarding] for PFC2) after consulting with its ACL ASIC to determine whether a packet is forwarded or not, depending on the Cisco IOS ACL that is configured. If a designated MSFC fails, the new designated MSFC must reprogram the PFC for static ACLs. For consistent results, both MSFCs *must* have identical ACL configurations, including the static ACLs.

Note In addition to defining the same ACLs on both MSFCs, you must also apply the ACLs to the same VLAN interfaces on both MSFCs.

Note The dynamic and reflexive ACLs, which are based on the actual data flow, may be programmed by either MSFC.

Note **PFC:** For detailed information on the hardware and software handling of the Cisco IOS ACLs with the PFC, see the "Hardware and Software Handling of Cisco IOS ACLs with PFC" section on page 15-10.

Note **PFC2:** For detailed information on the hardware and software handling of the Cisco IOS ACLs with PFC2, see the "Hardware and Software Handling of Cisco IOS ACLs with PFC2 and PFC3A/PFC3B/PFC3BXL" section on page 15-13.

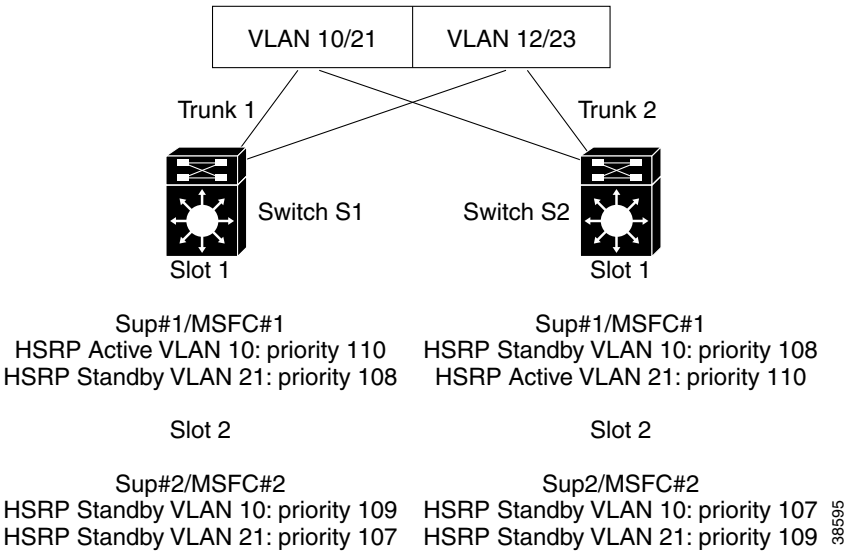
To determine the status of the designated MSFC, enter the **show fm features** or the **show redundancy** command:

```
Router-15# show redundancy
Designated Router: 1 Non-designated Router:2
Redundancy Status: non-designated
Config Sync AdminStatus  : enabled
Config Sync RuntimeStatus: enabled

Router-16# show redundancy
Designated Router: 1 Non-designated Router:2
Redundancy Status: designated
Config Sync AdminStatus  : enabled
Config sync RuntimeStatus: enabled
```

Dual MSFC Operational Model for Redundancy and Load Sharing

Figure 23-2 shows a typical access and distribution layer building block with multiple VLANs in an access layer switch. Because there is no Layer 2 loop, HSRP is used for convergence and load sharing. Switches S1 and S2 have a supervisor engine with an MSFC in slot 1 (Sup #1/MSFC #1) and in slot 2 (Sup #2/MSFC #2). Sup #1 is active and Sup #2 is in standby mode in both switches. High availability is enabled on the supervisor engines. The supervisor engines automatically perform image and configuration synchronization; you must manually synchronize the images and configurations on the MSFCs.

Figure 23-2 Dual MSFC Operational Model for Redundancy and Load Sharing—VLANs 10 and 21



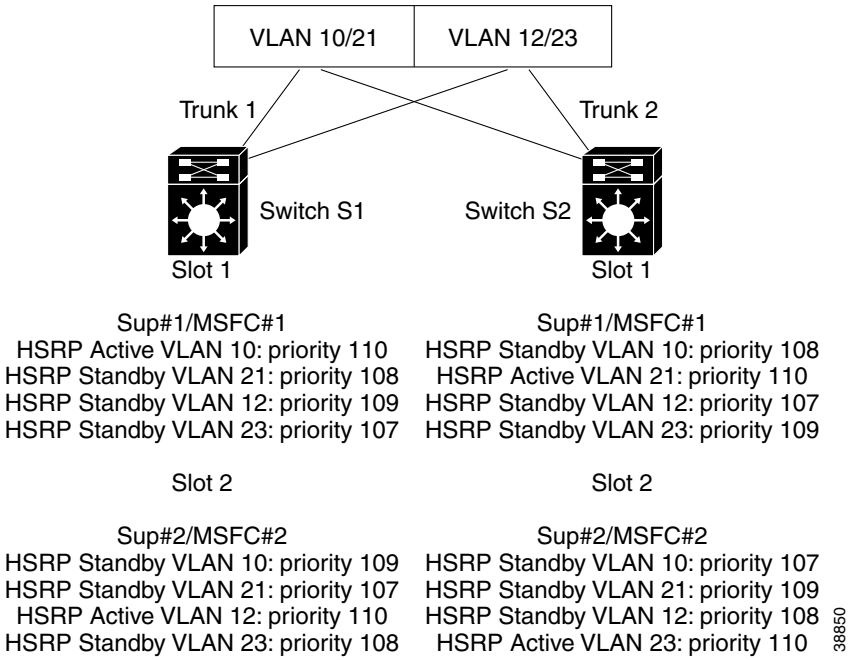
In Figure 23-2, you should configure redundancy and load sharing as follows:

- VLAN 10 (even-numbered VLANs)—Configure MSFC #1 in Switch S1 as the primary HSRP router (priority 110), and configure MSFC #2 as the standby router (priority 109).
- VLAN 21 (odd-numbered VLANs)—Configure MSFC #1 in Switch S2 as the primary HSRP router (priority 110), and configure MSFC #2 as the standby router (priority 109).

Load sharing is achieved by having the even-numbered VLANs routed by Switch S1 and the odd-numbered VLANs routed by Switch S2. In a complete switch failure, the remaining switch would service both the even and odd VLANs.

You can achieve further load sharing by using MSFC #2 in Switch S1 as the primary HSRP router for VLAN 12 and MSFC #2 as the primary HSRP router in Switch S2 for VLAN 23 (see Figure 23-3).

Figure 23-3 Dual MSFC Operational Model for Redundancy and Load Sharing—VLANs 10, 12, 21, and 23



Only the active HSRP router for a VLAN will respond with the HSRP MAC address for ARP requests to the HSRP IP address. The active HSRP router will in turn ARP for the end stations' MAC address and populate its ARP cache. By using both MSFCs in a single chassis to share the HSRP duties for the even VLANs, you can share the control plane ARP traffic. In an MSFC failure, only the ARP entries on the affected VLAN would need to be relearned.

The tradeoff for this level of redundancy and load sharing is the added complexity of keeping track of the even and odd VLANs on the MSFCs within a Catalyst 6500 series switch chassis.

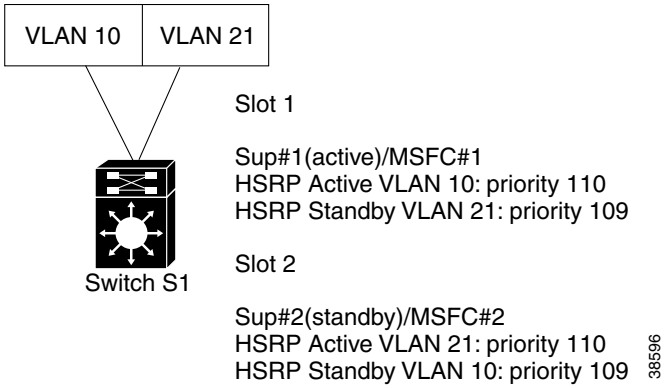
The MLS entries are created for the packets arriving at the HSRP MAC addresses and those packets arriving with the router's real MAC addresses. HSRP is used for unicast traffic first-hop redundancy; for traffic that is received through another router attached to VLAN 10, for example, the actual MAC address of Sup #1/MSFC #1 is used.

Understanding Failure Scenarios

These five examples describe the possible failure scenarios within a single chassis with dual supervisor engines and dual MSFCs (see Figure 23-4) when you enable high availability. The designated MSFC refers to the MSFC that is used to program the ACL ASIC for the static ACLs.

Note While the examples are specific to the PFC, the failover scenarios for the PFC2/MSFC2 would be similar for handling the ACLs and the CEF table entries. On a Supervisor Engine 2, the designated MSFC2 programs many of the ASICs on the PFC2 including building the CEF table. In a designated MSFC2 HSRP failover to the nondesignated MSFC2, the PFC2 continues to function with the CEF table that is programmed by the previously designated MSFC2. Similar to the process with the MLS cache in a Supervisor Engine 1/MSFC configuration, the newly designated MSFC2 reprograms the CEF table with its own entries and the old entries age out.

Figure 23-4 Single Chassis with Dual Supervisor Engines and Dual MSFCs



Failure Case 1: Designated MSFC #1 Fails

This sequence occurs when the designated MSFC #1 fails:

1. The MLS entries for MSFC #1 gracefully age out of the Sup #1 Layer-3 cache, while MSFC #2 takes temporary ownership of these MLS entries using its XTAG value.
2. The MLS entries for MSFC #2 are not affected.
3. MSFC #2 removes all the dynamic and reflexive ACLs that are programmed in the hardware by MSFC #1.
4. MSFC #2 reprograms the static ACLs in the Sup #1 ACL ASIC because it is now the designated MSFC.

Failure Case 2: Nondesignated MSFC #2 Fails

This sequence occurs when the nondesignated MSFC #2 fails:

1. The MLS entries for MSFC #2 gracefully age out of the Sup #1 Layer 3 cache, while MSFC #1 takes temporary ownership of these MLS entries using its XTAG value.
2. The MLS entries from MSFC #1 are not affected.
3. MSFC #1 removes all the dynamic and reflexive ACLs that are programmed in the hardware by MSFC #2.
4. MSFC #1 remains the designated MSFC.

Failure Case 3: Active Sup #1 Fails

This sequence occurs when the active supervisor engine (Sup #1) fails:

1. Because the Layer 3 state is maintained, the MLS entries of MSFC #1 gracefully age out of the Sup #2 Layer 3 cache while MSFC #2 takes temporary ownership of these MLS entries using its XTAG value.
2. The standby supervisor engine maintains the Layer 2 state so that there is no Layer 2 convergence time.
3. MSFC #2 removes all the dynamic and reflexive ACLs that are programmed in the hardware by MSFC #1.
4. MSFC #2 reprograms the static ACLs in the Sup #2 ACL ASIC. MSFC #2 is now the designated MSFC.

Failure Case 4: Standby Sup #2 Fails

This sequence occurs when the standby supervisor engine (Sup #2) fails:

1. The MLS entries for MSFC #2 gracefully age out of the Sup #1 Layer 3 cache while MSFC #1 takes temporary ownership of these MLS entries using its XTAG value.
2. The MLS entries from MSFC #1 are not affected.
3. MSFC #1 removes all the dynamic and reflexive ACLs that are programmed in the hardware by MSFC #2. MSFC #1 remains the designated MSFC.

Failure Case 5: New or Previously Failed Supervisor Comes Back Online

This sequence occurs when the previously failed supervisor engine (Sup #2) comes online:

1. Sup #1 continues to be the active supervisor engine.
2. Sup #2 synchronizes its image and configuration with Sup #1 (unless high-availability versioning is enabled).
3. MSFC #2 (on Sup #2) comes up. If the HSRP preempt for VLAN 21 is configured, then MSFC #2 becomes HSRP active. The MLS entries for MSFC #1 are purged and then relearned through MSFC #2.
4. MSFC #1 remains the designated MSFC for the static ACLs.

Configuring Redundancy with HSRP

Although the supervisor engine software high-availability feature maintains the protocol state between the redundant supervisor engines, you need to configure HSRP for failover between the redundant MSFCs. HSRP is used to provide first-hop, unicast redundancy. You can configure one or more HSRP groups on the MSFC VLAN interfaces to provide automatic routing backup for your network. Each VLAN interface in an HSRP group shares a virtual IP address and MAC address. You can configure the end stations and the other devices to use the HSRP address as the default gateway so that if one router interface fails, the service is not interrupted to those devices.

The interface with the highest HSRP priority is the active interface for that HSRP group.

Note **PFC2:** The PFC2 supports a maximum of 16 unique HSRP group numbers. You can use the same HSRP group numbers in different VLANs. If you configure more than 16 HSRP groups, this restriction prevents use of the VLAN number as the HSRP group number.

Note **PFC2:** Identically numbered HSRP groups use the same virtual MAC address, which might cause errors if you configure bridging on the MSFC.

Do not enter the **standby use-bia** option in an HSRP configuration. The MLS entries are not created when you enter the **standby use-bia** option. When the **standby use-bia** option is configured, if an HSRP active interface goes up and down, there will be no router CAM address for the standby VLAN interface. Without the router CAM entry, no shortcuts are created. This problem is independent of any MSFC Cisco IOS release. (This problem is documented in caveat CSCdz17169.)

To configure HSRP on an MSFC VLAN interface, perform this task in interface configuration mode:

	Task	Command
Step 1	Enable HSRP and specify the HSRP IP address. If you do not specify a *group_number*, group 0 is used. To assist in troubleshooting, configure the group number to match the VLAN number.	Router(config-if)# **standby** [*group_number*] **ip** [*ip_address*]
Step 2	Specify the priority for the HSRP interface. Increase the priority of at least one interface in the HSRP group (the default is 100). The interface with the highest priority becomes active for that HSRP group.	Router(config-if)# **standby** [*group_number*] **priority** *priority*
Step 3	Configure the interface to preempt the current active HSRP interface and become active if the interface priority is higher than the priority of the current active interface.	Router(config-if)# **standby** [*group_number*] **preempt** [**delay** *delay*]
Step 4	(Optional) Set the HSRP hello timer and holdtime timer for the interface. The default values are 3 (hello) and 10 (holdtime). All interfaces in the HSRP group should use the same timer values.	Router(config-if)# **standby** [*group_number*] **timers** *hellotime holdtime*
Step 5	(Optional) Specify a clear-text HSRP authentication string for the interface. All interfaces in the HSRP group should use the same authentication string.	Router(config-if)# **standby** [*group_number*] **authentication** *string*

This example shows how to configure an interface as part of HSRP group 100:

```
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# interface vlan100
Router(config-if)# standby 100 ip 172.20.100.10
Router(config-if)# standby 100 priority 110
Router(config-if)# standby 100 preempt
Router(config-if)# standby 100 timers 5 15
Router(config-if)# standby 100 authentication Secret
Router(config-if)# ^Z
Router#
```

Configuration Examples

This section describes three configuration options for achieving redundancy:

- Example 1: Two Chassis with One Supervisor Engine and One MSFC Each, page 23-30
- Example 2: Single Chassis with Dual Supervisor Engines and MSFCs, page 23-31
- Example 3: Double Chassis with Dual Supervisor Engines and MSFCs, page 23-33

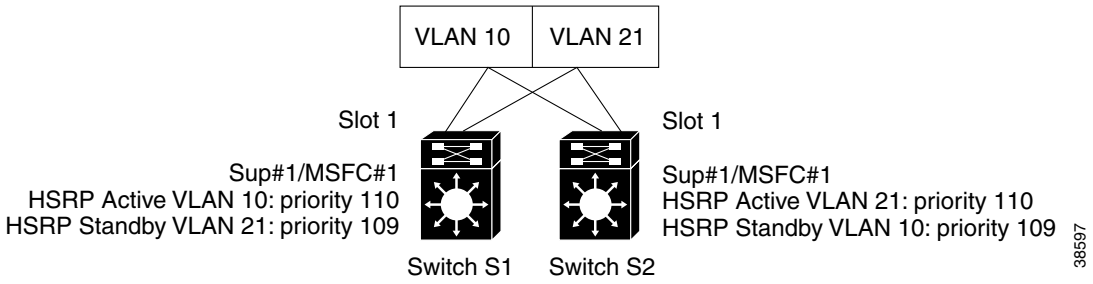
For the following examples, the designated MSFC is on the active supervisor engine. To determine the status of the designated MSFC, enter the **show fm features** or the **show redundancy** command. This example shows that Router-16 is the designated MSFC:

```
Router-15# show redundancy
Designated Router: 1 Non-designated Router:2

Redundancy Status: non-designated
Config Sync AdminStatus  : enabled
Config Sync RuntimeStatus: enabled

Router-16# show redundancy
Designated Router: 1 Non-designated Router:2

Redundancy Status: designated
Config Sync AdminStatus  : enabled
Config sync RuntimeStatus: enabled
```

Example 1: Two Chassis with One Supervisor Engine and One MSFC Each

In Figure 23-5, high availability cannot be configured on the supervisor engines, but HSRP can be configured on the MSFCs.

Figure 23-5 Two Chassis with One Supervisor Engine and One MSFC Each



This example shows how to configure HSRP on the MSFC in Switch S1:

```
Console> (enable) switch console 15
Trying Router-15...
Connected to Router-15.
Type ^C^C^C to switch back...
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# interface vlan10
Router(config-if)# standby 10 ip 172.20.100.10
Router(config-if)# standby 10 priority 110
Router(config-if)# standby 10 preempt
Router(config-if)# standby 10 timers 5 15
Router(config-if)# standby 10 authentication Secret
Router(config-if)# interface vlan21
Router(config-if)# standby 21 ip 192.20.100.21
Router(config-if)# standby 21 priority 109
Router(config-if)# standby 21 preempt
Router(config-if)# standby 21 timers 5 15
Router(config-if)# standby 21 authentication Secret
Router(config-if)# ^Z
Router# ^C^C^C
```

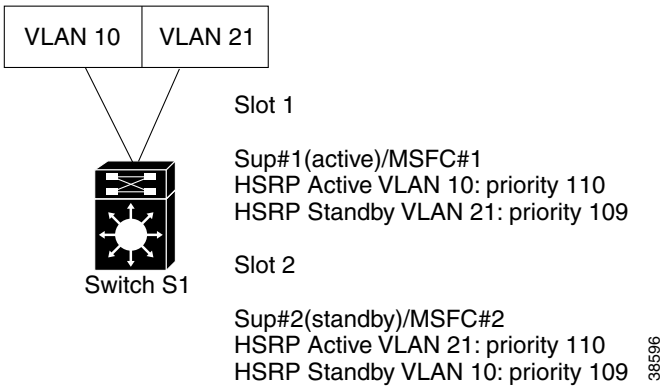
This example shows how to configure HSRP on the MSFC in Switch S2:

```
Console> (enable) switch console 15
Trying Router-15...
Connected to Router-15.
Type ^C^C^C to switch back...
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# interface vlan10
Router(config-if)# standby 10 ip 172.20.100.10
Router(config-if)# standby 10 priority 109
Router(config-if)# standby 10 preempt
Router(config-if)# standby 10 timers 5 15
Router(config-if)# standby 10 authentication Secret
Router(config-if)# interface vlan21
Router(config-if)# standby 21 ip 192.20.100.21
Router(config-if)# standby 21 priority 110
Router(config-if)# standby 21 preempt
Router(config-if)# standby 21 timers 5 15
Router(config-if)# standby 21 authentication Secret
Router(config-if)# ^Z
Router# ^C^C^C
```

Example 2: Single Chassis with Dual Supervisor Engines and MSFCs

In Figure 23-6, high availability is configured on the supervisor engines, and HSRP is configured on the MSFCs.

Figure 23-6 Single Chassis with Redundant Supervisor Engines and MSFCs



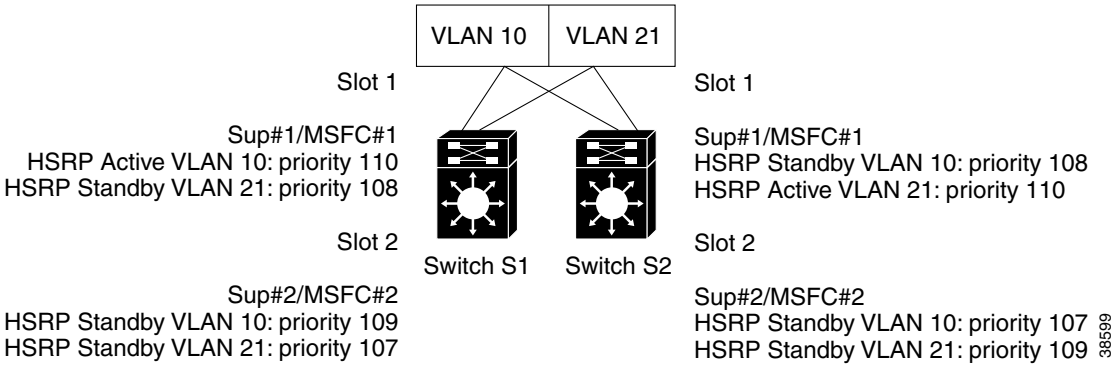
This example shows how to configure HSRP on the MSFC in Switch S1:

```
Console> (enable) switch console 15
Trying Router-15...
Connected to Router-15.
Type ^C^C^C to switch back...
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# interface vlan10
Router(config-if)# standby 10 ip 172.20.100.10
Router(config-if)# standby 10 priority 110
Router(config-if)# standby 10 preempt
Router(config-if)# standby 10 timers 5 15
Router(config-if)# standby 10 authentication Secret
Router(config-if)# interface vlan21
Router(config-if)# standby 21 ip 192.20.100.21
Router(config-if)# standby 21 priority 109
Router(config-if)# standby 21 preempt
Router(config-if)# standby 21 timers 5 15
Router(config-if)# standby 21 authentication Secret
Router(config-if)# ^Z
Router# ^C^C^C

Console> (enable) switch console 16
Trying Router-16...
Connected to Router-16.
Type ^C^C^C to switch back...
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# interface vlan10
Router(config-if)# standby 10 ip 172.20.100.10
Router(config-if)# standby 10 priority 109
Router(config-if)# standby 10 preempt
Router(config-if)# standby 10 timers 5 15
Router(config-if)# standby 10 authentication Secret
Router(config-if)# interface vlan21
Router(config-if)# standby 21 ip 192.20.100.21
Router(config-if)# standby 21 priority 110
Router(config-if)# standby 21 preempt
Router(config-if)# standby 21 timers 5 15
Router(config-if)# standby 21 authentication Secret
Router(config-if)# ^Z
Router# ^C^C^C
```

Example 3: Double Chassis with Dual Supervisor Engines and MSFCs

Figure 23-7 shows two Catalyst 6500 series switches (S1 and S2), each with a supervisor engine and MSFC in slot 1 (Sup #1/MSFC #1) and slot 2 (Sup #2/MSFC #2). Because there is no Layer-2 loop, HSRP is used for convergence and load sharing. In both switches, Sup #1 is the active supervisor engine, and Sup #2 is the standby supervisor engine.

Figure 23-7 Dual MSFC Operational Model for Redundancy and Load Sharing



This example shows how to configure HSRP on the MSFC in Switch S1:

```
Console> (enable) switch console 15
Trying Router-15...
Connected to Router-15.
Type ^C^C^C to switch back...
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# interface vlan10
Router(config-if)# standby 10 ip 172.20.100.10
Router(config-if)# standby 10 priority 110
Router(config-if)# standby 10 preempt
Router(config-if)# standby 10 timers 5 15
Router(config-if)# standby 10 authentication Secret
Router(config-if)# interface vlan21
Router(config-if)# standby 21 ip 192.20.100.21
Router(config-if)# standby 21 priority 108
Router(config-if)# standby 21 preempt
Router(config-if)# standby 21 timers 5 15
Router(config-if)# standby 21 authentication Secret
Router(config-if)# ^Z
Router# ^C^C^C

Console> (enable) switch console 16
Trying Router-16...
Connected to Router-16.
Type ^C^C^C to switch back...
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# interface vlan10
Router(config-if)# standby 10 ip 172.20.100.10
Router(config-if)# standby 10 priority 109
Router(config-if)# standby 10 preempt
Router(config-if)# standby 10 timers 5 15
Router(config-if)# standby 10 authentication Secret
Router(config-if)# interface vlan21
Router(config-if)# standby 21 ip 192.20.100.21
Router(config-if)# standby 21 priority 107
```

```
Router(config-if)# standby 21 preempt
Router(config-if)# standby 21 timers 5 15
Router(config-if)# standby 21 authentication Secret
Router(config-if)# ^Z
Router# ^C^C^C
```

This example shows how to configure HSRP on the MSFC in Switch S2:

```
Console> (enable) switch console 15
Trying Router-15...
Connected to Router-15.
Type ^C^C^C to switch back...
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# interface vlan10
Router(config-if)# standby 10 ip 172.20.100.10
Router(config-if)# standby 10 priority 108
Router(config-if)# standby 10 preempt
Router(config-if)# standby 10 timers 5 15
Router(config-if)# standby 10 authentication Secret
Router(config-if)# interface vlan21
Router(config-if)# standby 21 ip 192.20.100.21
Router(config-if)# standby 21 priority 110
Router(config-if)# standby 21 preempt
Router(config-if)# standby 21 timers 5 15
Router(config-if)# standby 21 authentication Secret
Router(config-if)# ^Z
Router# ^C^C^C

Console> (enable) switch console 16
Trying Router-16...
Connected to Router-16.
Type ^C^C^C to switch back...
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# interface vlan10
Router(config-if)# standby 10 ip 172.20.100.10
Router(config-if)# standby 10 priority 107
Router(config-if)# standby 10 preempt
Router(config-if)# standby 10 timers 5 15
Router(config-if)# standby 10 authentication Secret
Router(config-if)# interface vlan21
Router(config-if)# standby 21 ip 192.20.100.21
Router(config-if)# standby 21 priority 109
Router(config-if)# standby 21 preempt
Router(config-if)# standby 21 timers 5 15
Router(config-if)# standby 21 authentication Secret
Router(config-if)# ^Z
Router# ^C^C^C
```

MSFC Configuration Synchronization Overview

MSFC high availability allows for automatic synchronization of the startup configuration and running configuration between the designated MSFC (the MSFC to come online first or the MSFC that has been online the longest) and the nondesignated MSFC. High-availability redundancy is disabled by default.

Caution Configuration synchronization is only supported for the IP and IPX configurations. Before enabling synchronization, you must ensure that both MSFCs have identical configurations for all protocols. If you are using AppleTalk, DECnet, VINES, or any other routing, you must manually ensure that identical configurations are on both MSFCs for all protocols.

To determine the status of the designated MSFC, enter the **show fm features** or the **show redundancy** command:

```
Router-15# show redundancy
Designated Router: 1 Non-designated Router:2

Redundancy Status: non-designated
Config Sync AdminStatus  : enabled
Config Sync RuntimeStatus: enabled

Router-16# show redundancy
Designated Router: 1 Non-designated Router:2

Redundancy Status: designated
Config Sync AdminStatus  : enabled
Config sync RuntimeStatus: enabled
```

High-availability redundancy provides the startup and running configuration synchronization.

When you enable high-availability redundancy, the startup configuration of both MSFCs is updated when you enter either of these commands on the designated MSFC:

- **write mem**
- **copy** *source* **startup-config**

When you enable high-availability redundancy, every configuration command that is executed on the designated MSFC is sent to the nondesignated MSFC. In addition, the running configuration synchronization is updated when you enter the **copy** *source* **running-config** command on the designated MSFC.

These sections provide information about the MSFC configuration synchronization:

- Configuration Synchronization States, page 23-35
- alt Keyword Usage, page 23-36

Configuration Synchronization States

The two states for the configuration synchronization are as follows:

- Config Sync AdminStatus—Signifies what the user has configured for this feature at that moment
- Config Sync RuntimeStatus—Enabled only when the following occurs:
  - The Config Sync AdminStatus is enabled on both the designated and nondesignated MSFCs
  - The designated and nondesignated MSFCs are running compatible images

When you enable the Config Sync RuntimeStatus, the following occurs:

- No configuration mode is available on the CLI of the nondesignated MSFC; EXEC mode is available
- The **alt** keyword is available and required (see the "alt Keyword Usage" section on page 23-36 for more information on the **alt** keyword)
- The running and startup configurations are synchronized

When the Config Sync RuntimeStatus is in disabled mode, the following occurs:

- Configuration mode is available on the CLI of both MSFCs
- The **alt** keyword is available but optional
- The running and startup configurations are not synchronized

The various configuration and operation cases are covered in the "High-Availability Redundancy Configuration Examples" section on page 23-37.

alt Keyword Usage

When you enable the Config Sync RuntimeStatus, the configuration mode on the nondesignated MSFC is disabled; only the EXEC mode is still available. The configuration of both MSFCs is made through the console or a Telnet session on the designated MSFC.

To configure both MSFCs from a single console, enter the **alt** keyword to specify an alternate configuration. When specifying the alternate configuration, the configuration that is specified before the **alt** keyword relates to the MSFC on the supervisor engine in slot 1 of the switch; the configuration that is specified after the **alt** keyword relates to the MSFC on the supervisor engine in slot 2.

Note You must enter the **alt** keyword when you enable Config Sync AdminStatus.

Table 23-3 shows the interface and global configuration commands that contain the **alt** keyword.

Table 23-3 Interface and Global Configuration Commands Containing the alt Keyword

Interface Configuration Commands	Global Configuration Commands
• [**no**] **standby** [*group_number*] **ip** [*ip_address* [**secondary**]] **alt** [**no**] **standby** [*group_number*] **ip** [*ip_address* [**secondary**]] • [**no**] **standby** [*group_number*] **priority** *priority* [**preempt** [**delay** *delay*]] **alt** [**no**] **standby** [*group_number*] **priority** *priority* [**preempt** [**delay** *delay*]] • [**no**] **ip address** *ip_address mask* [**secondary**] **alt** [**no**] **ip address** *ip_address mask* [**secondary**] • [**no**] **ipx network** *network* [**encapsulation** *encapsulation_type* [**secondary**]] [**alt** [**no**] **ipx network** *network* [**encapsulation** *encapsulation_type* [**secondary**]]]	• [**no**] **hostname** *hostname* **alt hostname** *hostname* • [**no**] **ip default-gateway** *ip_address* **alt** [**no**] **ip default-gateway** *ip_address* • **router bgp** *autonomous_system* **bgp router-id** *ip_address* [**alt** *ip_address*] • **router ospf** *process_id* **router-id** *ip_address* [**alt** *ip_address*]

This example shows how to use the **alt** keyword when entering the **ip address** command:

```
Router-1(config-if)# ip address 1.2.3.4 255.255.255.0 alt ip address 1.2.3.5 255.255.255.0
```

Enabling or Disabling Configuration Synchronization

To enable high-availability redundancy, perform this task in privileged mode:

	Task	Command
Step 1	Enable redundancy.	**redundancy**
Step 2	Enable high availability.	**high-availability**
Step 3	Enable or disable configuration synchronization.	[**no**] **config-sync**

This example shows how to enable high-availability redundancy and configuration synchronization (Router-15 is the designated MSFC):

```
Console>(enable) session 15
Trying Router-15...
Connected to Router-15.
Escape character is '^]'.

Router-15> enable
Router-15# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router-15(config)# redundancy
Router-15(config-r)# high-availability
Router-15(config-r-ha)# config-sync
Router-15(config-r-ha)# end
```

Note When you enable high-availability redundancy, the configuration mode is disabled on the nondesignated MSFC; only EXEC mode is available.

In this example, Router-16 is the nondesignated MSFC; high-availability redundancy and configuration synchronization are enabled:

```
Console>(enable) session 16
Trying Router-16...
Connected to Router-16.
Escape character is '^]'.

Router-16> enable
Router-16# configure terminal
Config mode is disabled on non-designated Router, please configure from designated Router
```

High-Availability Redundancy Configuration Examples

This section describes the different scenarios for enabling high availability and configuration synchronization:

Scenario 1: Enabling Configuration Synchronization on Both MSFCs

This scenario assumes that both MSFCs are up.

When you enable the configuration synchronization on both MSFCs, the IP addresses on all the interfaces are checked first. If an IP address is specified for the designated MSFC but is not specified for the nondesignated MSFC, a message is displayed indicating the first interface for which the alternate IP address was not specified.

After checking the IP addresses, the HSRP addresses are checked. If an HSRP address is specified for the designated MSFC but not specified for the nondesignated MSFC, a message is displayed indicating the first interface for which the alternate HSRP (standby) address was not specified.

After checking the HSRP addresses, the IPX network address is checked.

The designated MSFC is configured first. This example shows a missing alternate configuration for the VLAN 1 interface:

```
Router-16# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router-16(config)# redundancy
Router-16(config-r)# high-availability
Router-16(config-r-ha)# config-sync

Alternate IP address missing for Vlan1
The alternate configuration is missing. The auto-config sync can not be enabled
```

Note When specifying the alternate IP configuration, the configuration that is specified before the **alt** keyword relates to the MSFC on the supervisor engine in slot 1 of the switch; the configuration that is specified after the **alt** keyword relates to the MSFC on the supervisor engine in slot 2. See the "alt Keyword Usage" section on page 23-36 for more information.

This example shows how to specify the alternate configuration for VLAN 1:

```
Router-16(config)# interface vlan 1
Router-16(config-if)# ip address 70.0.70.4 255.255.0.0 alt ip address 70.0.70.5
255.255.0.0
Router-16(config-if)# exit
```

This example shows that high-availability redundancy is accepted:

```
Router-16(config)# redundancy
Router-16(config-r)# high-availability
Router-16(config-r-ha)# config-sync
Router-16(config-r-ha)# end
Router-16#
00:03:31: %SYS-5-CONFIG_I: Configured from console by console
```

Because the Config Sync AdminStatus on the nondesignated MSFC is disabled, the Config Sync RuntimeStatus on the designated MSFC will remain in disabled mode. The following message is displayed on the designated MSFC:

```
00:17:05: %RUNCFGSYNC-6-SYNCEVENT:
Non-Designated Router is now online
High-Availability Redundancy Feature is not enabled on the Non-Designated Router
```

This example shows how to enable the configuration synchronization feature on the nondesignated MSFC:

```
Router-151(config)# redundancy
Router-15(config-r)# high-availability
Router-15(config-r-ha)# config-sync
Router-15(config-r-ha)# end
Router-15#
00:03:31: %SYS-5-CONFIG_I: Configured from console by console
```

Note When you enable high-availability redundancy, the configuration mode is disabled on the console of the nondesignated MSFC; only EXEC mode is available.

This message, which acknowledges that the high-availability redundancy is enabled and that the configuration mode is automatically exited, is displayed on the nondesignated MSFC:

```
00:18:57: %RUNCFGSYNC-6-SYNCEVENT:
The High-Availability Redundancy Feature is enabled
The config mode is no longer accessible

Router-15#

00:19:41: %RUNCFGSYNC-6-SYNCEVENT:
Non-Designated Router is now online
Running Configuration Synchronization will begin in 1 minute
```

A 1-minute timer will start, allowing for the nondesignated MSFC to stabilize. When the timer expires, a snapshot of the current running configuration is sent to the nondesignated MSFC. This message is displayed before the running configuration is synchronized:

```
00:20:41: %RUNCFGSYNC-6-SYNCEVENT:
Syncing Running Configuration to the Non-Designated Router

00:20:41: %RUNCFGSYNC-6-SYNCEVENT:
Syncing Startup Configuration to the Non-Designated Router
```

These examples show that the designated MSFC and nondesignated MSFC have the same running configuration after synchronization:

```
<designated MSFC>
Router-16# show running-config
Building configuration...

Current configuration:
!
version 12.1
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Router-15 alt hostname Router-16
!
boot bootldr bootflash:c6msfc-boot-mz.120-7.XE1
!
ip subnet-zero
!
ip cef
redundancy
 high-availability
 config-sync
cns event-service server
!
!
!
interface Vlan1
 ip address 70.0.70.4 255.255.0.0 alt ip address 70.0.70.5 255.255.0.0
!
interface Vlan10
ip address 192.10.10.1 255.255.255.0 alt ip address 192.10.10.2 255.255.255.0
 no ip redirects
 shutdown
 standby ip 192.20.20.1 alt standby ip 192.20.20.1
!
ip classless
ip route 223.255.254.0 255.255.255.0 70.0.100.0
no ip http server
!
```

```
!
!
line con 0
 transport input none
line vty 0 4
 login
 transport input lat pad mop telnet rlogin udptn nasi
!
end


<nondesignated MSFC>
Router-15# show running-config
Building configuration...

Current configuration:
!
version 12.1
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Router1 alt hostname Router2
!
boot bootldr bootflash:c6msfc-boot-mz.120-7.XE1
!
ip subnet-zero
!
ip cef
redundancy
 high-availability
 config-sync
cns event-service server
!
!
!
interface Vlan1
 ip address 70.0.70.4 255.255.0.0 alt ip address 70.0.70.5 255.255.0.0
!
interface Vlan10
ip address 192.10.10.1 255.255.255.0 alt ip address 192.10.10.2 255.255.255.0
 no ip redirects
 shutdown
 standby ip 192.20.20.1 alt standby ip 192.20.20.1
!
ip classless
ip route 223.255.254.0 255.255.255.0 70.0.100.0
no ip http server
!
!
!
line con 0
 transport input none
line vty 0 4
 login
 transport input lat pad mop telnet rlogin udptn nasi
!
end
```

Scenario 2: Disabling Configuration Synchronization on the Designated MSFC

In this scenario, the configuration synchronization is enabled. These examples show how to disable the configuration synchronization:

```
Router-16# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router2(config)# redundancy
Router2(config-r)# high-availability
Router2(config-r-ha)# no config-sync
```

When the configuration synchronization is disabled, the following message is displayed on the nondesignated MSFC:

```
00:13:00: %RUNCFGSYNC-6-SYNCEVENT:
The High-Availability Redundancy Feature is now disabled
The config mode is now accessible
```

Configuration mode is available on the CLI of the designated and nondesignated MSFC.

Scenario 3: Designated MSFC Comes Up

In this scenario, Config Sync AdminStatus is enabled. The designated MSFC validates the alternate configuration, allowing the configuration synchronization to occur when the nondesignated MSFC comes up.

Because the nondesignated MSFC is not up yet, Config Sync RuntimeStatus is disabled, and there is no configuration synchronization. See the "Scenario 4: Nondesignated MSFC Comes Up" section on page 23-41 for information on the nondesignated MSFC.

This example shows that Router-16 is the designated MSFC, Config Sync AdminStatus is enabled, and Config Sync RuntimeStatus is disabled:

```
Router-16# show redundancy
Designated Router: 1 Non-designated Router:0

Redundancy Status: designated
Config Sync AdminStatus  : enabled
Config Sync RuntimeStatus: disabled
```

Scenario 4: Nondesignated MSFC Comes Up

Config Sync AdminStatus is Enabled

In this scenario, the nondesignated MSFC notifies the designated MSFC that it is up and Config Sync AdminStatus is enabled. The designated MSFC requests the nondesignated MSFC to enable Config Sync RuntimeStatus. The nondesignated MSFC enables Config Sync RuntimeStatus.

This message is displayed on the nondesignated MSFC:

```
00:00:07: %RUNCFGSYNC-6-SYNCEVENT:
The High-Availability Redundancy Feature is enabled
The config mode is no longer accessible

00:00:51: %RUNCFGSYNC-6-SYNCEVENT:
Non-Designated Router is now online
Running Configuration Synchronization will begin in 1 minute
```

A 1-minute timer will start, allowing the nondesignated MSFC to stabilize. When the timer expires, a snapshot of the current running configuration is sent to the nondesignated MSFC. This message is displayed before synchronizing the running configuration:

```
00:01:51: %RUNCFGSYNC-6-SYNCEVENT:
Syncing Running Configuration to the Non-Designated Router
```

Config Sync AdminStatus is Disabled

In this scenario, the nondesignated MSFC notifies the designated MSFC that it is up. Because the Config Sync AdminStatus is disabled on the nondesignated MSFC, the designated MSFC displays the following message indicating that high-availability redundancy needs to be enabled on the nondesignated MSFC:

```
Router-16#
Non-Designated Router came up.
High-Availability Redundancy Feature is not enabled on the Non-Designated Router
```

This example shows how to enable the high-redundancy availability on the nondesignated MSFC:

```
Router-15# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router-15(config)# redundancy
Router-15(config-r)# high-availability
Router-15(config-r-ha)# config-sync
Router-15(config-r-ha)#
00:03:47: %SYS-5-CONFIG_I: Configured from console by console
00:03:47: %RUNCFGSYNC-6-SYNCEVENT:
The High-Availability Redundancy Feature is enabled
The config mode is no longer accessible

00:00:51: %RUNCFGSYNC-6-SYNCEVENT:
Non-Designated Router is now online
Running Configuration Synchronization will begin in 1 minute
```

A 1-minute timer will start, allowing the nondesignated MSFC to stabilize. When the timer expires, a snapshot of the current running configuration is sent to the nondesignated MSFC. This message is displayed before synchronizing the running configuration:

```
00:01:51: %RUNCFGSYNC-6-SYNCEVENT:
Syncing Running Configuration to the Non-Designated Router
```

These examples show that Config Sync AdminStatus and RuntimeStatus are enabled on the designated and nondesignated MSFCs:

```
Router-15# show redundancy
Designated Router: 1 Non-designated Router:2

Redundancy Status: non-designated
Config Sync AdminStatus  : enabled
Config Sync RuntimeStatus: enabled


Router-16# show redundancy
Designated Router: 1 Non-designated Router:2

Redundancy Status: designated
Config Sync AdminStatus  : enabled
Config sync RuntimeStatus: enabled
```

Scenario 5: Designated MSFC Goes Down

In this scenario, the nondesignated MSFC becomes the designated MSFC, the configuration synchronization is disabled, and the configuration mode on the CLI is now available.

When the previously designated MSFC comes back up, it becomes the nondesignated MSFC; see the "Scenario 4: Nondesignated MSFC Comes Up" section on page 23-41.

Single Router Mode Redundancy

These sections describe how to configure single router mode (SRM) redundancy:

SRM redundancy is an alternative to the internally redundant (dual) MSFC2 configurations where both MSFC2s are active at the same time. In SRM redundancy, only the designated router is visible to the network at any given time. The nondesignated router is booted up completely and participates in the configuration synchronization which is automatically enabled when entering SRM. All configuration following the "alt" keyword is ignored in SRM; the nondesignated router's configuration is exactly the same as the designated router but its interfaces are kept in a line down state and are not visible to the network. The processes, such as the routing protocols, are created on the nondesignated router and the designated router, but all the nondesignated router interfaces are in a line down state; they do not send or receive updates from the network.

When the designated router fails, the nondesignated router changes its state from a nondesignated router to a designated router and its interface state changes to link up. The newly designated router builds up its routing table while the existing supervisor engine switch processor entries are used to forward the Layer 3 traffic. The switch processor continues to forward the Layer 3 packets using the old entries. After a predefined time, the newly designated router downloads the new Layer 3 switching information to the switch processor.

Note With Cisco IOS Release 12.1(11b)E and later releases, you can specify the transition time that the newly designated router waits before downloading the new Layer 3 switching information to the supervisor engine switch processor. For configuration details, see the "Specifying the Transition Time on the Newly Designated Active Router" section on page 23-47.

Hardware and Software Requirements

To configure SRM redundancy, you must have the following hardware and software:

- A single chassis with two identical supervisor engine daughter card configurations:
  - Supervisor Engine 32 with Policy Feature Card 3B/3BXL (PFC3B/PFC3BXL) and MSFC2A
  - Supervisor Engine 720 with PFC3A/PFC3B/PFC3BXL and MSFC3

  - Supervisor Engine 2 with PFC2 and MSFC2
  - Supervisor Engine 1 with PFC and MSFC or MSFC2

Note Cisco IOS Release 12.1(8a)E4 provides initial support for SRM redundancy with Supervisor Engine 1 and the MSFC.

Note **Multicast support:** In software releases prior to software release 7.1(1), when using Supervisor Engine 1 with the MSFC or MSFC2 for SRM redundancy, be aware that failover to the second MSFC is not stateful for multicast MLS. When the primary MSFC fails, all the multicast MLS entries are removed and are then recreated and reinstalled in the hardware by the newly active MSFC.

Note **Multicast support:** In software release 7.1(1) and later releases, there is improved SRM redundancy support for multicast traffic for Supervisor Engine 1 with PFC and MSFC2 and Supervisor Engine 2 with PFC2 and MSFC2. The multicast improvements are not supported on Supervisor Engine 1 with the MSFC.

When SRM redundancy is enabled, there are improved convergence times and less disruption of multicast traffic during the switchovers. The MSFC2 is protected from being overloaded with the multicast traffic during the switchover. The switch caches the flows from the MSFC2 that went down and uses the cached flows to forward traffic until the newly activated MSFC2 learns the routes. Only a few flows at a time are provided to the MSFC2 to prevent it from being overwhelmed.

- Supervisor engine software release 6.3(1) or later releases (Supervisor Engine 720 requires supervisor engine software release 8.1(1) or later releases and Supervisor Engine 32 requires supervisor engine software release 8.4(1) or later releases)
- Cisco IOS Release 12.1(8a)E2 or later releases (Supervisor Engine 720 MSFC3 requires Cisco IOS Release 12.2(14)SX2 or later releases)

SRM Redundancy Configuration Guidelines

This section describes the guidelines for configuring SRM redundancy:

- Both the designated router and nondesignated router must run the same Cisco IOS image.
- A Cisco IOS image must be present in the bootflash of both the designated router and nondesignated router.
- The nondesignated router cannot connect to the external networks.
- Do not boot from an external network with the designated router because booting from the network could severely degrade the SRM functionality.
- With SRM redundancy, the designated router can reach the external networks and copy commands, such as **copy tftp:**, can be used without any restrictions.
- You must enable high availability on the supervisor engine.

- When using the authentication methods to control access to the switch, such as RADIUS or TACACS+, you must configure a fallback option to log in with a local username and password if you want to access the nondesignated router through the **switch console** or **session** commands.

  See Chapter 39, "Configuring the Switch Access Using AAA" for information on configuring the fallback option.

Configuring Single Router Mode Redundancy on Supervisor Engine 720

Note Single router mode redundancy is the only supported MSFC redundancy option for Supervisor Engine 720 and Supervisor Engine 32.

With Supervisor Engine 720 and Supervisor Engine 32, you do not have to explicitly enable SRM on the MSFC; SRM is enabled by default. To ensure that SRM works properly, you must verify that both MSFCs have the identical startup configuration with one of these two methods:

1. After the system is reset and has reloaded, enter the **write erase** command on the nondesignated MSFC and reload the nondesignated MSFC.
2. After the system is reset and has reloaded, enter the **show redundancy** command to verify that the SRM run-time status is enabled. After verifying that the SRM run-time status is enabled, enter the **write memory** command on the designated MSFC and reload the nondesignated MSFC (do not save the configuration on the nondesignated MSFC at the reload prompt).

Tip We recommend that you use the second method where the nondesignated MSFC starts with the same configuration as the designated MSFC. This method lessens the chance of encountering any unforeseen problems.

Note If the nondesignated MSFC startup configuration had configuration items that were not present in the designated MSFC configuration during the system boot, the MSFCs will be inconsistent with their run-time configuration state. Both procedures ensure that the nondesignated MSFC always has the same run-time configuration as the designated MSFC.

Configuring Single Router Mode Redundancy on Supervisor Engine 1 or Supervisor Engine 2

To configure SRM redundancy, perform these steps:

Caution Before going from dual router mode to SRM redundancy, we recommend that you use the **copy running-config** command on the MSFCs to save the non-SRM configuration to bootflash. When going to SRM redundancy, the alternative configuration (the configuration following the **alt** keyword) is lost. Before enabling SRM redundancy, save the dual router mode configuration to bootflash by entering the **copy running-config bootflash:***nosrm_dual_router_config* command on both MSFCs.

See the "Getting Out of Single Router Mode" section on page 23-49 for additional information.

Note This procedure assumes that the designated router is the MSFC2 in slot 1 and the nondesignated router is the MSFC2 in slot 2; the active supervisor engine is in slot 1 and the standby supervisor engine is in slot 2.

Step 1 Enter the **show version** command to ensure that both supervisor engines are running supervisor engine software release 6.3(1) or later releases.

Step 2 Enter the **set system highavailability enable** command to enable high availability on the active supervisor engine. Enter the **show system highavailability** command to verify that high availability is enabled.

Step 3 If you have a console connection, enter the **switch console** command to access the designated router. If connected through a Telnet session, enter the **session** *mod* command to access the designated router.

Step 4 Copy the Cisco IOS Release 12.1(8a)E2 or later image to the bootflash of the designated router and nondesignated router.

Step 5 Set the boot image and configuration register on the designated router and nondesignated router to boot the new image on a reload.

For the designated router, enter the **boot system flash bootflash:***image_name* command and ensure that this image is the first image in the boot list. Clear any existing **boot system** commands that appear in the running configuration (**show running-config**) using the **no** form of the **boot system** command.

For the nondesignated router, set the configuration register to auto boot by entering the **config-register 0x102** command.

Note If you already have SRM-capable Cisco IOS images loaded, you do not need to perform Step 6.

Step 6 Enter the **reload** command to reload the designated router and nondesignated router.

Step 7 Disable the configuration synchronization (**config-sync**) on the designated router by entering the **no** form of the command. Enter the **write memory** command. This step lets you access the configuration mode on both the designated and nondesignated routers.

Step 8 Enable SRM on the designated router first, and then enable SRM on the nondesignated router as follows:

```
Router(config)# redundancy
Router(config-r)# high-availability
Router(config-r-ha)# single-router-mode
```

Note With Cisco IOS Release 12.1(11b)E and later releases, you can specify the transition time that the newly designated router waits before downloading the new Layer 3 switching information to the supervisor engine switch processor. For configuration details, see the "Specifying the Transition Time on the Newly Designated Active Router" section on page 23-47.

Step 9 Enter the **write memory** command on the designated router to ensure that the nondesignated router's startup configuration has SRM enabled.

Step 10 Enter the **show startup-config** command on the nondesignated router to ensure that the nondesignated router has the following configuration statements:

```
redundancy
 high-availability
 single-router-mode
```

Step 11 Enter the **show redundancy** command on the designated router and nondesignated router to ensure that both routers have the following configuration statement:

```
Single Router Mode RuntimeStatus: enabled
```

If not, repeat Steps 9 and 10 and allow sufficient time between steps.

Step 12 Enter the **reload** command to reload the nondesignated router. When asked whether the configuration should be saved, enter **no**.

This display summarizes the configuration commands that were used on the designated router and the nondesignated router to enable SRM redundancy:

```
Time   Designated Router                      Nondesignated Router
  ----   ---                                  ----
  t0:    conf t->red->hi->no config-sync
  t1:                                         conf t->red->hi->no config-sync
  t2:    conf t->red->hi->single-router-mode
  t3:                                         conf t->red->hi->single-router-m
  t4:    write mem
  t5:                                         reload
```

Specifying the Transition Time on the Newly Designated Active Router

With Cisco IOS releases prior to Release 12.1(11b)E, the transition time was 120 seconds and was not configurable. Because of the differences in the routing convergence times, 120 seconds might not be long enough. The older Layer 3 switching entries might be erased, and the newly downloaded Layer 3 switching information might be incomplete.

With Cisco IOS Release 12.1(11b)E and later releases, you can specify the transition time that the newly designated router waits before downloading the new Layer 3 switching information to the switch processor. On a switchover, the old Layer 3 switching information is used for a configurable number of seconds before the new Layer 3 switching information is downloaded to the switch processor.

If nonstop forwarding is required, we do not recommend setting the transition time to a value that is lower than the default (120 seconds). At a minimum, it takes 30 to 60 seconds for routes to converge.

To specify the transition time, enter these commands (in this example, the transition time is set to 240 seconds):

```
Router(config)# redundancy
Router(config-r)# high-availability
Router(config-r-ha)# single-router-mode
Router(config-r-ha)# single-router-mode failover ?
   table-update-delay  Adjust for routing convergence time
Router(config-r-ha)# single-router-mode failover table-update-delay ?
   <0-4294967295>  Delay in seconds between switch over detection and h/w FIB reload
Router(config-r-ha)# single-router-mode failover table-update-delay 240
Router(config-r-ha)#
```

To set the transition time to the 2-minute default, use the **no** form of the command as follows:

```
Router(config-r-ha)# no single-router-mode failover table-update-delay
```

Display the transition time as follows:

```
Router-16# show redundancy
 Designated Router: 2 Non-designated Router: 1

 Redundancy Status: designated

 Config Sync AdminStatus  : enabled

 Config Sync RuntimeStatus: enabled

 Single Router Mode AdminStatus  : enabled

 Single Router Mode RuntimeStatus: enabled

 Single Router Mode transition timer : 240 seconds    <---- transition time

Router-16#
```

Upgrading Images with Single Router Mode Enabled

This section describes how to upgrade the Cisco IOS image on the active and standby MSFC when SRM is running. The new image name is c6msfc2-jsv-mz.9E. The standby MSFC cannot load an image using TFTP, but it can load an image from the supervisor engine Flash PC card (sup-slot0:).

Note This procedure impacts the data traffic. We recommend that you perform this procedure during a scheduled maintenance window.

To upgrade the images, perform these steps:

Step 1 On the active supervisor engine, enter the **copy tftp sup-slot0:** command and follow the prompts to load the new (c6msfc2-jsv-mz.9E) image onto the supervisor engine Flash PC card.

Step 2 If you have a console connection, enter the **switch console** command to access the active MSFC. If you are connected through a Telnet session, enter the **session** *mod* command to access the active MSFC.

Step 3 On the active MSFC, copy the new image from the supervisor engine Flash PC card to the MSFC bootflash as follows:

```
copy sup-slot0:c6msfc2-jsv-mz.9E bootflash:c6msfc2-jsv-mz.9E
```

Step 4 Access the standby MSFC by entering the **switch supervisor** command and then the **switch console** command on the active supervisor engine.

Note The standby MSFC does not appear in the **show module** command display that is issued from the active supervisor engine.

Step 5 On the standby MSFC, copy the new image from the supervisor engine Flash PC card to the MSFC bootflash as follows:

```
copy sup-slot0:c6msfc2-jsv-mz.9E bootflash:c6msfc2-jsv-mz.9E
```

Step 6 On the active MSFC, specify that the new image is booted when the MSFC is reloaded as follows:

```
boot system flash bootflash:c6msfc2-jsv-mz.9E
```

Step 7 On the active MSFC, enter the **write memory** command to ensure that the standby MSFC startup configuration gets the boot information.

Step 8 Enter the **reload** command to reload the standby MSFC.

Step 9 Enter the **show redundancy** command on the active and standby MSFCs to ensure that both have the following configuration statement:

```
Single Router Mode RuntimeStatus: enabled
```

Step 10 Enter the **reload** command to reload the active MSFC.

Both MSFCs are now running the c6msfc2-jsv-mz.9E image.

Getting Out of Single Router Mode

Note If you saved a copy of the running configuration that was used in dual router mode before configuring SRM redundancy, you do not need to use the procedure in this section. To get out of SRM redundancy and back to dual router mode, enter the **copy bootflash:***nosrm_dual_router_config startup-config* command on both MSFCs. After the configurations are copied, reload the MSFCs using the **reload** command.

To get out of SRM, perform these steps:

Step 1 On the designated router, disable SRM by entering the **no** form of the command as follows:

```
Router(config)# redundancy
Router(config-r)# high-availability
Router(config-r-ha)# no single-router-mode
```

Step 2 Enter the **write memory** command on the designated router and nondesignated router.

Step 3 Enter the **show startup-config** command on the designated and nondesignated routers to ensure that "single-router mode" is not in the startup configuration.

Step 4 Enter the **reload** command to reload the designated router and nondesignated router.

SRM is now disabled on the designated router and nondesignated router.

Manual-Mode MSFC Redundancy

Note Manual-mode MSFC redundancy will be supported until December 2002 due to the release of supervisor engine software release 6.3(1), which contains the feature SRM. We recommend that you use SRM rather than manual-mode MSFC redundancy to attain the automatic Layer-3 failover capabilities in addition to unlimited support of the feature.

These sections describe how to configure the redundant MSFCs with one MSFC active and the other MSFC in ROM-monitor mode:

Hardware and Software Requirements

To configure Layer 3 redundancy, you must have at least one of the following configurations:

- A single chassis with two identical supervisor engine daughter card configurations:
  - Supervisor Engine 1 with Policy Feature Card (PFC) and MSFC or MSFC2 (both supervisor engines must have the same type of MSFC)
  - Supervisor Engine 2 with PFC2 and MSFC2
- Two chassis with a supervisor engine in each—You must have at least one supervisor engine in each chassis. Each supervisor engine must be equipped with a PFC and an MSFC.
- Manual-mode MSFC redundancy requires the following software:
  - Supervisor engine software release 6.1(3) or later releases and Cisco IOS Release 12.1(7)E or later releases
  - Supervisor engine software release 5.5.8 or later releases and Cisco IOS Release 12.1(7a)E1 or later releases

Note Each MSFC must run the same release of Cisco IOS software.

Manual-Mode MSFC Redundancy Guidelines

This section describes the guidelines for configuring the manual-mode MSFC redundancy:

- Because the MSFC switchover is manual, we recommend that you have this feature only in environments where the externally redundant routers are present and where either HSRP is used or some form of gateway discovery is implemented on the hosts.
- Ensure that the configuration register on the active MSFC (MSFC-15) is set to 0x2102 and that the configuration register on the MSFC in ROM-monitor mode (MSFC-16) is set to 0x0. This setting prevents both MSFCs from becoming active at the same time and allows the active MSFC to come online after a reset. See the "Setting the MSFC Configuration Register" section on page 23-52 for details on setting the configuration register.

Note Setting both MSFCs to 0x0 is a supported option but requires user intervention if the switch is reset.

- To conserve the IP address space and reduce the overall Layer 3 complexity, ensure that configuration synchronization is disabled on both MSFCs and that all "alt" addresses are removed. If the alt addresses are used, the IP address space is not conserved and in cases where link-level peering is present (such as BGP), the Layer 3 complexity is increased.
- When the MSFC in ROM-monitor mode is brought up during a maintenance window, ensure that it has the exact configuration as the active MSFC. Follow the configuration guidelines in Table 23-2 on page 23-22.
- During manual-mode MSFC redundancy, you should enable high availability on the supervisor engine to keep the Layer 2 downtime to a minimum when doing an MSFC switchover. Since high availability is not compatible with protocol filtering, port security, Dynamic VLAN (DVLAN), or Generic Attribute Registration Protocol (GARP) VLAN Registration Protocol (GVRP), we recommend that you disable these features when using manual-mode MSFC redundancy.
- Ensure that the console port on both supervisor engines is accessible to the operations personnel (out-of-band access through the terminal server or modem).

Note The procedures in this section use the **switch console** command to access the MSFC from the supervisor engine. The **switch console** command is not supported on Telnet sessions.

Accessing the Standby MSFC

To access the standby MSFC, enter the **switch supervisor** command and then the **switch console** command.

Note The standby MSFC does not appear in the **show module** command display that is issued from the active supervisor engine.

Manually Booting the MSFC

If the configuration register on both MSFCs is set to 0x0, then MSFC manual mode requires that you manually boot the MSFC each time that the switch is reset. To boot the MSFC manually, perform these steps:

Step 1 Enter the **switch console** command to gain access to the MSFC ROMMON prompt.

Step 2 Enter the **boot bootflash:***image* command.

Step 3 Once the MSFC boots, press **Ctrl-C** three times at the Router> prompt to return to the supervisor engine prompt. Enter the **session** command to access the MSFC.

Setting the MSFC Configuration Register

For manual-mode MSFC redundancy, set the configuration registers as follows:

Step 1 From Cisco IOS configuration mode on the active MSFC (MSFC-15), enter the **config-register 0x2102** command.

Step 2 On the MSFC in ROM-monitor mode (MSFC-16), enter the **config-register 0x0** command.

Note We recommend that **boot** system commands in both MSFC configurations point to a valid image on bootflash and that you do not set the configuration registers to ignore these **boot** commands.

MSFC Recovery Procedures

This section describes how to recover from the temporary or permanent MSFC failures.

A temporary failure of the active MSFC results in an MSFC reboot because the configuration register is set to 0x2102.

You need to verify a suspected permanent failure of the active MSFC. Enter the **reset 15** command from the active supervisor engine's console port and see if the active MSFC reboots without problems. If it does not, you have these two options to switch over to the standby MSFC:

Option 1: If You Have Physical Access to the Switch

If you have physical access to the switch, use this option. You can remove the active supervisor engine with the problematic MSFC, so that the redundant supervisor engine will take over. From the redundant supervisor engine's physical console port, perform these steps:

Step 1 Enter the **switch console** command.

Step 2 From the ROMMON prompt, enter the **boot bootflash:***image* command.

Step 3 After the standby MSFC has booted, enter the **config-register 0x2102** command from Cisco IOS configuration mode, to ensure that the MSFC will boot when the switch is reset.

Option 2: If You Have Only Remote Access to the Switch

If you have only remote access to the switch, use this option. From the active supervisor engine with the problematic MSFC, perform these steps:

Note If the problematic MSFC is on the standby supervisor engine, enter the **switch supervisor** command.

Step 1 Enter the **switch console** command.

Step 2 Send a Break signal to get into the problematic MSFC's ROMMON (the break will work if the MSFC is continually rebooting). You need to time the break so that it is issued after the system bootstrap message, but before the main Cisco IOS image is decompressed (see the two arrows in the following display output):

```
System Bootstrap, Version 12.0(3)XE, RELEASE SOFTWARE
Copyright (c) 1998 by cisco Systems, Inc.
Cat6k-MSFC platform with 131072 Kbytes of main memory <======= ISSUE BREAK AFTER THIS
POINT


Self decompressing the image :
################################################################################################
[OK]

<==========BUT BEFORE THIS POINT

Self decompressing the image :
################################################################################################
################################################################################################
################################################################################################
################################################################################################
################################################################################################
################################################################################################
################################################################################################
################################################################################################
### [OK]
```

Step 3 At the ROMMON prompt, enter the **confreg** command:

a. Enter **y** at the "do you wish to change the configuration? y/n [n]:" prompt.

b. Press **Enter** to accept the default for all questions until you reach this prompt: "change the boot characteristics? y/n [n]:"

c. Enter **y**.

d. Enter **0** to select the "0 = ROM Monitor" option at the next prompt.

e. Review the Configuration Summary to ensure the following value: boot: the ROM Monitor.

f. You are again prompted with: "do you wish to change the configuration? y/n [n]:"

g. Enter **n**.

h. You are returned to the ROMMON prompt.

Step 4 Enter the **reset** command and verify that the MSFC boots into ROMMON. This step ensures that this MSFC and the active MSFC will not boot concurrently.

Step 5 Press **Ctrl-C** three times to return to the supervisor engine prompt.

Step 6 Ensure that high availability has synchronized the supervisor engine state by entering the **show system highavailability** command and verifying that high availability "Operational-status" is ON.

Step 7 Enter the **switch supervisor** command.

Step 8 Enter the **switch console** command.

Step 9 From the standby MSFC's ROMMON prompt, repeat Step 3 but in Step 3d, select option 2 "boot system" as follows:

```
change the boot characteristics? y/n  [n]:  y
enter to boot:
 0 = ROM Monitor
 1 = the boot helper image
 2-15 = boot system
    [2]:  2   <========================


    Configuration Summary
enabled are:
load rom after netboot fails
console baud: 9600
boot: the ROM Monitor

do you wish to change the configuration? y/n  [n]:  n


You must reset or power cycle for new config to take effect
rommon 2 >
```

Step 10 Enter the **reset** command at the ROMMON prompt to boot the system.

Step 11 Once the MSFC has booted, enter the **config-register 0x2102** command from Cisco IOS configuration mode on the newly active MSFC's console port.

Step 12 Press **Ctrl-C** three times to return to the supervisor engine prompt.

CHAPTER 24

Configuring NSF with SSO MSFC Redundancy

This chapter describes how to configure MSFC redundancy using Cisco nonstop forwarding (NSF) with stateful switchover (SSO) on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series MSFC Cisco IOS Command Reference*.

Note The term *MSFC* is used throughout this chapter to refer to MSFC2, MSFC2A, and MSFC3 except where specifically differentiated.

Note Except where specifically differentiated, the information and procedures in this chapter apply to Supervisor Engine 32 with PFC3B/PFC3BXL, Supervisor Engine 720 with PFC3A/PFC3B/PFC3BXL, and Supervisor Engine 2 with PFC2.

This chapter consists of these sections:

Hardware and Software Requirements

This sections describes the hardware and software requirements for configuring NSF/SSO:

- Supported supervisor engines— Supervisor Engine 2, Supervisor Engine 720, and Supervisor Engine 32 (NSF/SSO is not supported with Supervisor Engine 1).
- Supported MSFCs—MSFC2, MSFC2A, and MSFC3 (the MSFC is not supported).
- The redundant supervisor engines must be the same type with the same model PFC and MSFC.
- Catalyst software release 8.5(1) and later releases.

Note If SSO is enabled on the MSFC, you must enable high availability on the supervisor engine *before* upgrading to supervisor engine software release 8.5(1) and later releases. Use the **set system highavailability enable** command to enable high availability on the supervisor engine.

- Cisco IOS Release 12.2(18)SXF and later releases.

Understanding How NSF/SSO Works

Note SSO replaces single router mode (SRM) and dual router mode (DRM). There is no support for these high-availability modes. For SRM and DRM CLI processing details, see the "Configuration Guidelines and Restrictions" section on page 24-4.

The Catalyst operating system that runs on the supervisor engine provides a Layer 2 high availability for redundant supervisor engines. Cisco IOS Release 12.2(18)SXF and later releases with NSF and SSO that run on the MSFC provide Layer 3 (and above) high availability for redundant MSFCs. MSFC SSO high-availability benefits are as follows:

- Reduced downtime.
- The ability to upgrade software without shutting down the MSFC.
- The ability to detect a failure of the active MSFC and allow the standby MSFC to take over the system with minimal drops in existing traffic flows.

When the system comes up, after the supervisor engine completes its initialization and prepares itself for operation, the supervisor engine sends an SCP inventory message to both MSFCs. The inventory message contains information about which MSFCs are present in the system and as other operational state information. From a high-availability perspective, the inventory message is important because it contains information that dictates which MSFC will be the active MSFC and which will be the standby MSFC.

During the startup of the standby MSFC, image version information is exchanged between MSFCs and one of the following occurs:

- If the image version information matches and both MSFCs are configured as SSO or have the default (SSO) configuration, the system runs in SSO mode.
- If the image version information does not match or if one of the MSFCs is configured for route processor redundancy (RPR), the system runs in RPR mode.

In NSF/SSO mode, one MSFC is active and the other MSFC is in a hot-standby mode. The hot-standby MSFC maintains a constant readiness state by receiving state information from the active MSFC. At any given moment, the standby MSFC may be called on by the supervisor engine to take over the responsibilities held by the active MSFC. The supervisor engine monitors the active MSFC and if the MSFC does not respond, the supervisor engine declares the MSFC as lost or down and proceeds to reset the MSFC. The standby MSFC has the up-to-date state information necessary to resume processing (the standby MSFC is fully initialized, but the VLANs are kept in an administrative down state until a switchover occurs).

With NSF, the switching modules and switch fabric continue to forward packets while the MSFC switchover is in progress.

Note Detected failures of hardware or CLI commands may also cause a switchover.

Note High availability on the supervisor engine operates independent of the MSFC high-availability feature. However, you must enable high availability on the supervisor engine must be enabled to ensure the correct operation of the MSFC SSO feature.

If you run the MSFC in SSO mode and fail to run the high-availability feature on the supervisor engine, any switchover that may occur will result in a nonstateful switchover and the standby MSFC will reset itself and reload at the time of the switchover. This reset/reload of the standby MSFC occurs because there is insufficient state information on the supervisor engine to support a stateful switchover of the MSFC. This reset/reload of the standby MSFC interrupts service.

RPR Overview

Note RPR+ mode is not supported.

RPR is a *cold* standby mode. When a switchover occurs, the standby MSFC must go completely through its initialization. RPR mode is used primarily for the fast software upgrade (FSU). (See the "Fast Software Upgrade" section on page 24-14.) In RPR mode, the startup configuration is synchronized to the standby MSFC, however, it is not processed in any way until the switchover occurs. The running configuration is not synchronized to the standby MSFC.

When the active MSFC boots completely, no state information is exchanged between the MSFCs. If the active MSFC fails, the standby MSFC processes its startup configuration file and begins its initialization.

If there is an image compatibility problem, the active MSFC boots fully, but the standby MSFC suspends its startup before processing the startup configuration file. If the active MSFC fails, a switchover is triggered and the suspended standby MSFC begins to initialize and become the active MSFC.

Note High availability on the supervisor engine does not have to be enabled to run RPR on the MSFC.

Types of MSFC Switchovers

The types of MSFC switchovers are as follows:

- Failover—An MSFC failover occurs when the active MSFC crashes or detects a serious system failure and ends up in ROMMON.
- Forced switchover—A forced switchover is caused by either entering a CLI command or by removing the supervisor engine with the active MSFC from the chassis. The MSFC CLI commands that force a switchover are the **redundancy force-switchover** and **reload** commands. The supervisor engine CLI command that forces a switchover is the **reset** *mod* command where *mod* is the module number of the MSFC as shown in the **show module** command display.

Configuration Guidelines and Restrictions

This section describes the configuration guidelines and restrictions for configuring NSF/SSO:

- If SSO is enabled on the MSFC, you must enable high availability on the supervisor engine *before* upgrading to supervisor engine software release 8.5(1) and later releases. Use the **set system highavailability enable** command to enable high availability on the supervisor engine.
- SSO replaces SRM and DRM. There is no support for these high-availability modes. The details are as follows:
  - SRM CLI processing—Cisco IOS Release 12.2(18)SXF and later software releases contain the SRM CLI. The CLI is accepted when entered but it is not acted on in any way. The SRM CLI was kept in Cisco IOS Release 12.2(18)SXF and later software releases to assist you in migrating to NSF/SSO. However, the SRM CLI does not cause NVRAM updates. If you have SRM CLI in your configuration and you decide to modify the SRM configuration and enter the **write mem** command, the SRM CLI commands in the configuration are lost. If you want to downgrade to an image that has SRM, your original SRM CLI configuration is lost and you will have to reconfigure SRM. For this reason, we recommend that you save your configuration before you upgrade from SRM to NSF/SSO.
  - DRM CLI processing—Unlike SRM CLI, any existing DRM CLI in the configuration file after upgrading to NSF/SSO are flagged as errors at system startup. You must reconfigure the switch and remove the DRM configuration. We recommend that you save your configuration before you upgrade from DRM to NSF/SSO.
- During a switchover, there will be traffic loss for traffic that is routed by the MSFC. NSF only applies to traffic that is hardware switched by modules and the switch fabric. New flows are not allowed until the switchover is complete.
- In cases where the MSFC has failed and is unable to notify the supervisor engine of the failure, the supervisor engine may take 30 to 40 seconds before it realizes that the MSFC has failed and a switchover is triggered. If the supervisor engine receives the failure notification, the switchover is triggered immediately.
- The Frame Relay, ATM, and PPP protocols that are not supported in SSO mode.
- WAN modules react to SSO switchovers as follows:
  - WAN modules do not reload with an SSO switchover.
  - WAN module interfaces go down and then come back up during an SSO switchover.
  - All routing protocols do not perform NSF if NSF is configured over WAN interfaces.
  - All features on the WAN interfaces resume operation after an SSO switchover.

- Standby supervisor engine/MSFC insertion—With NSF/SSO redundancy, you can hot swap the standby supervisor engine/MSFC for maintenance. When you hot insert the standby MSFC, the active MSFC detects the presence of the standby MSFC and starts to drive the standby MSFC state transition to hot-standby. When you remove the standby MSFC, the synchronization between the active and standby MSFC is stopped, any pending updates to the standby MSFC are discarded, and the system enters simplex mode. The standby MSFC state is displayed by entering the **show redundancy states** command.
- Counters and statistics—The various counters and statistics that are maintained by the MSFC are not synchronized between MSFCs.
- Not all subsystems are high-availability aware and those that are high-availability aware may have their own set of limitations.
- Some subsystems have their own high availability-specific configurations and status commands (such as the **show isis nsf** command).
- MSFC software images do not currently support the in-service software upgrade (ISSU).
- Diagnostics are not integrated into high availability. Switchovers due to failed diagnostics on the MSFC are not supported.

Using the CLI to Configure NSF/SSO

These sections describe how to configure NSF/SSO:

Configuring SSO

SSO is the default mode. By default, even if you do not configure the system explicitly as SSO, the system comes up in SSO mode. However, we recommend that you explicitly configure SSO mode.

Note The following task can also be used to configure RPR mode (use **mode rpr** instead of **mode sso**).

To configure SSO mode, perform this task:

	Task	Command
Step 1	Enter redundancy configuration mode.	`Router(config)# redundancy`
Step 2	Configure SSO. When this command is entered, the redundant MSFC is reloaded and begins to work in SSO mode.	`Router(config-red)# mode sso`
Step 3	Verify that SSO is enabled.	`Router# show running-config`
Step 4	Display the operating redundancy mode.	`Router# show redundancy states`

This example shows how to configure the system for SSO and display the redundancy state:

```
Router> enable
Router# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)# redundancy
Router(config-red)# mode sso
Router(config-red)# end
Router# show redundancy states
my state = 13 -ACTIVE
     peer state = 1  -DISABLED
           Mode = Simplex
           Unit = Primary
        Unit ID = 7
Redundancy Mode (Operational) = Stateful SwitchOver - SSO
Redundancy Mode (Configured)  = Stateful SwitchOver - SSO
Redundancy State              = Non Redundant

     Split Mode = Disabled
   Manual Swact = Disabled  Reason: Simplex mode
 Communications = Down      Reason: Simplex mode

   client count = 18
 client_notification_TMR = 30000 milliseconds
          keep_alive TMR = 9000 milliseconds
        keep_alive count = 0
    keep_alive threshold = 18
           RF debug mask = 0x0
Router#
```

Configuring CEF NSF

CEF NSF operates by default while the networking device is running in SSO mode. No configuration is necessary.

Verifying CEF NSF

To verify that CEF is NSF-capable, perform this task:

Task	Command
Verify that CEF is NSF-capable.	`Router# show cef state`

This example shows how to verify that CEF is NSF-capable:

```
router# show cef state
CEF Status [RP]
 CEF enabled/running
 dCEF enabled/running
 CEF switching enabled/running
 CEF default capabilities:
  Always CEF switching:               yes
  Always dCEF switching:              yes
  Default CEF switching:              yes
  Default dCEF switching:             yes
  Drop multicast packets:             no
  OK to punt packets:                 yes
  NVGEN CEF state:                    yes
  fastsend() used:                    no
  CEF NSF capable:                    yes
  RPR+/SSO standby capable:           yes
  IPC delayed func on SSO:            no
  FIB auto repair supported:          yes
  LCs not running at init time:       yes
  Hardware forwarding supported:      yes
  Hardware forwarding in use:         yes
  Load-sharing pr. packet supported:  no
 RRP state:
  I am standby RRP:                   no
  RF Peer Presence:                   no
  RF PeerComm reached:                no
  Config Redundancy mode:             Stateful SwitchOver - SSO(7)
  Operating Redundancy mode:          Stateful SwitchOver - SSO(7)
  CEF NSF:                            enabled/not running
 RP state:
  Expanded LC ipc memory:             0 Kbytes
  Linecard reloader type:             aggressive (Default)
  Linecard dFIB structures:           initialized
Router#
```

Configuring BGP NSF

Note You must configure BGP graceful restart on all peer devices that participate in BGP NSF.

To configure BGP for NSF, perform this task (repeat this procedure on each of the BGP NSF peer devices):

	Purpose	Command
Step 1	Enter global configuration mode.	`Router# configure terminal`
Step 2	Enable a BGP routing process, which places the router in router configuration mode.	`Router(config)# router bgp as-number`
Step 3	Enable the BGP graceful restart capability, starting BGP NSF. If you enter this command after the BGP session has been established, you must restart the session for the capability to be exchanged with the BGP neighbor. Use this command on the restarting router and all of its peers.	`Router(config-router)# bgp graceful-restart`

Verifying BGP NSF

To verify BGP NSF, you must check that the graceful restart function is configured on the SSO-enabled networking device and on the neighbor devices. To verify, follow these steps:

Step 1 Verify that "bgp graceful-restart" appears in the BGP configuration of the SSO-enabled router by entering the **show running-config** command:

```
Router# show running-config

.
.
.
router bgp 120
.
.
.
bgp graceful-restart
 neighbor 10.2.2.2 remote-as 300
.
.
.
```

Step 2 Repeat step 1 on each of the BGP neighbors.

Step 3 On the SSO device and the neighbor device, verify that the graceful restart function is shown as both advertised and received, and confirm the address families that have the graceful restart capability.

Note If no address families are listed, then BGP NSF also will not occur.

```
Router# show ip bgp neighbors x.x.x.x

BGP neighbor is 192.168.2.2,  remote AS YY, external link
  BGP version 4, remote router ID 192.168.2.2
  BGP state = Established, up for 00:01:18
  Last read 00:00:17, hold time is 180, keepalive interval is 60 seconds
  Neighbor capabilities:
    Route refresh:advertised and received(new)
    Address family IPv4 Unicast:advertised and received
    Address famiiy IPv4 Multicast:advertised and received
    Graceful Restart Capabilty:advertised and received
      Remote Restart timer is 120 seconds
      Address families preserved by peer:
        IPv4 Unicast, IPv4 Multicast
  Received 1539 messages, 0 notifications, 0 in queue
  Sent 1544 messages, 0 notifications, 0 in queue
  Default minimum time between advertisement runs is 30 seconds
```

Configuring OSPF NSF

Note All peer devices that participate in OSPF NSF must be made OSPF NSF-aware, which happens automatically once you install an NSF software image on the device.

To configure OSPF NSF, perform this task:

	Purpose	Command
Step 1	Enter global configuration mode.	`Router# configure terminal`
Step 2	Enable an OSPF routing process, which places the router in router configuration mode.	`Router(config)# router ospf processID`
Step 3	Enable NSF operations for OSPF.	`Router(config-router)# nsf`

Verifying OSPF NSF

To verify OSPF NSF, you must check that the NSF function is configured on the SSO-enabled networking device. To verify OSPF NSF, follow these steps:

Step 1 Verify that "nsf"' appears in the OSPF configuration of the SSO-enabled device by entering the **show running-config** command.

```
Router# show running-config

router ospf 120
log-adjacency-changes
nsf
network 192.168.20.0 0.0.0.255 area 0
network 192.168.30.0 0.0.0.255 area 1
network 192.168.40.0 0.0.0.255 area 2
.
.
.
```

Step 2 Verify that NSF is enabled on the device by entering the **show ip ospf** command.

```
router> show ip ospf

Routing Process "ospf 1" with ID 192.168.2.1 and Domain ID 0.0.0.1
Supports only single TOS(TOS0) routes
Supports opaque LSA
SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs
Number of external LSA 0. Checksum Sum 0x0
Number of opaque AS LSA 0. Checksum Sum 0x0
Number of DCbitless external and opaque AS LSA 0
Number of DoNotAge external and opaque AS LSA 0
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
External flood list length 0
Non-Stop Forwarding enabled, last NSF restart 00:02:06 ago (took 44 secs)
Area BACKBONE(0)
Number of interfaces in this area is 1 (0 loopback)
 Area has no authentication
 SPF algorithm executed 3 times
```

Configuring IS-IS NSF

To configure IS-IS NSF, perform this task:

	Purpose	Command
Step 1	Enter global configuration mode.	`Router# configure terminal`
Step 2	Enable an IS-IS routing process, which places the router in router configuration mode.	`Router(config)# router isis` [*tag*]

	Purpose	Command
Step 3	Enable NSF operation for IS-IS. Enter the **ietf** keyword to enable IS-IS in a homogeneous network where adjacencies with networking devices supporting IETF draft-based restartability is guaranteed. Enter the **cisco** keyword to run IS-IS in heterogeneous networks that might not have adjacencies with NSF-aware networking devices.	`Router(config-router)# nsf [cisco | ietf]`
Step 4	(Optional) Specify the minimum time between NSF restart attempts. The default time between *consecutive* NSF restart attempts is 5 minutes.	`Router(config-router)# nsf interval [minutes]`
Step 5	(Optional) Specify the time that IS-IS will wait for the IS-IS database to synchronize before generating overloaded link-state information for itself and flooding that information out to its neighbors. The **t3** keyword applies only if you selected IETF operation. When you specify **adjacency**, the router that is restarting obtains its wait time from neighboring devices.	`Router(config-router)# nsf t3 {manual [seconds] | adjacency}`
Step 6	(Optional) Specify how long an IS-IS NSF restart will wait for all interfaces with IS-IS adjacencies to come up before completing the restart. The default is 10 seconds.	`Router(config-router)# nsf interface wait seconds`

Verifying IS-IS NSF

To verify IS-IS NSF, you must check that the NSF function is configured on the SSO-enabled networking device. To verify IS-IS NSF, perform these steps:

Step 1 Verify that "nsf" appears in the IS-IS configuration of the SSO-enabled device by entering the **show running-config** command. The display will show either the Cisco IS-IS or the IETF IS-IS configuration. This example indicates that the device uses the Cisco implementation of IS-IS NSF:

```
Router# show running-config
<...Output Truncated...>
router isis
nsf cisco
<...Output Truncated...>
```

Step 2 If the NSF configuration is set to **cisco**, enter the **show isis nsf** command to verify that NSF is enabled on the device. Using the Cisco configuration, the display output will be different on the active and redundant MSFCs (RPs). This example shows the sample output for the Cisco configuration on the active MSFC (RP). In this example, note the presence of "NSF restart enabled":

```
router# show isis nsf

NSF is ENABLED, mode 'cisco'

RP is ACTIVE, standby ready, bulk sync complete
NSF interval timer expired (NSF restart enabled)
Checkpointing enabled, no errors
```

```
Local state:ACTIVE,  Peer state:STANDBY HOT,  Mode:SSO
```

This example shows the sample output for the Cisco configuration on the standby RP. In this example, note the presence of "NSF restart enabled":

```
router# show isis nsf

NSF enabled, mode 'cisco'
RP is STANDBY, chkpt msg receive count:ADJ 2, LSP 7
NSF interval timer notification received (NSF restart enabled)
Checkpointing enabled, no errors
Local state:STANDBY HOT,  Peer state:ACTIVE,  Mode:SSO
```

Step 3 If the NSF configuration is set to **ietf**, enter the **show isis nsf** command to verify that NSF is enabled on the device. This example shows the sample output for the IETF IS-IS configuration on the networking device:

```
router# show isis nsf

NSF is ENABLED, mode IETF
NSF pdb state:Inactive
NSF L1 active interfaces:0
NSF L1 active LSPs:0
NSF interfaces awaiting L1 CSNP:0
Awaiting L1 LSPs:
NSF L2 active interfaces:0
NSF L2 active LSPs:0
NSF interfaces awaiting L2 CSNP:0
Awaiting L2 LSPs:
Interface:Serial3/0/2
    NSF L1 Restart state:Running
    NSF p2p Restart retransmissions:0
    Maximum L1 NSF Restart retransmissions:3
    L1 NSF ACK requested:FALSE
    L1 NSF CSNP requested:FALSE
    NSF L2 Restart state:Running
    NSF p2p Restart retransmissions:0
    Maximum L2 NSF Restart retransmissions:3
    L2 NSF ACK requested:FALSE
Interface:GigabitEthernet2/0/0
    NSF L1 Restart state:Running
    NSF L1 Restart retransmissions:0
    Maximum L1 NSF Restart retransmissions:3
    L1 NSF ACK requested:FALSE
    L1 NSF CSNP requested:FALSE
    NSF L2 Restart state:Running
    NSF L2 Restart retransmissions:0
    Maximum L2 NSF Restart retransmissions:3
    L2 NSF ACK requested:FALSE
    L2 NSF CSNP requested:FALSE
Interface:Loopback1
    NSF L1 Restart state:Running
    NSF L1 Restart retransmissions:0
    Maximum L1 NSF Restart retransmissions:3
    L1 NSF ACK requested:FALSE
    L1 NSF CSNP requested:FALSE
    NSF L2 Restart state:Running
    NSF L2 Restart retransmissions:0
    Maximum L2 NSF Restart retransmissions:3
    L2 NSF ACK requested:FALSE
    L2 NSF CSNP requested:FALSE
```

Displaying Redundancy-Related Information

Use the **show redundancy** [qualifier] command to display redundancy-related information. The supported qualifiers are as follows:

```
Router# show redundancy ?
  clients          Redundancy Facility (RF) client list
  counters         Redundancy Facility (RF) operational counters
  events           Redundancy Facility (RF) events list
  history          Redundancy Facility (RF) history
  linecard-group   Line card redundancy group information
  states           Redundancy Facility (RF) states
  switchover       Redundancy Facility (RF) switchover
  |                Output modifiers
  <cr>

Router#
```

Performing an MSFC Switchover

Use the **redundancy switch-activity** [**force**] command to switch over to the standby MSFC. The force **keyword** overrides any restrictions.

Performing an MSFC Software Reload

Use the **redundancy reload** {**peer** | **shelf**} command to reload the standby MSFC (**peer** keyword) or all modules in the chassis (**shelf** keyword).

Using Redundancy-Related Debug Commands

Use the **debug redundancy** [qualifier] command to display redundancy-related debug information. The supported qualifiers are as follows:

```
Router# debug redundancy ?
  config-sync  HA config sync debug option
  ehsa         Redundancy Facility (RF) EHSA
  errors       Redundancy Facility (RF) Errors
  fsm          Redundancy Facility (RF) FSM events
  kpa          Redundancy Facility (RF) keep alive
  msg          Redundancy Facility (RF) Messaging events
  progression  Redundancy Facility (RF) Progression events
  status       Redundancy Facility (RF) Status events
  timer        Redundancy Facility (RF) Timer events

Router#
```

Use the **debug hybrid-ha** [qualifier] command to display NSF/SSO-specific redundancy information. The supported qualifiers are as follows:

```
Router# debug hybrid-ha ?
  all     All Hybrid HA SSO/NSF platform specific debugging messages
  errors  Hybrid HA SSO/NSF platform specific warnings and errors
  events  Hybrid HA SSO/NSF platform specific events
  ipc     Hybrid HA SSO/NSF platform specific IPC related events
  kpa     Hybrid HA SSO/NSF platform specific Keep-Alive related events

Router#
```

Upgrading Software

These sections describe how to upgrade the MSFC software:

Note Before performing any software upgrade procedure, see the "Configuration Guidelines and Restrictions" section on page 24-4.

Fast Software Upgrade

Note Because the system is in RPR mode during the fast software upgrade, service is interrupted. The switchover is not stateful; interfaces go down but come back up as the MSFC initializes and comes up in RPR mode. Additionally, any configuration changes that are not saved are lost.

Note This procedure requires that the Cisco IOS Release on both MSFCs supports RPR (at a minimum), and both MSFCs must be running the same software version. The active MSFC checks the standby image version when the standby MSFC is coming up, and if the standby image version does not match the active image version, the redundancy mode falls back to RPR.

Note This procedure does not work with SRM and DRM images.

Note The redundant supervisor engines must be the same type with the same model PFC and MSFC.

The fast software upgrade allows you to reduce planned downtime for software upgrades or downgrades. The fast software upgrade procedure consists of loading a new image onto both the standby MSFC and the active MSFC, and then rebooting the standby MSFC. The new image running on the standby MSFC is incompatible with the image currently running on the active MSFC, so the standby MSFC comes up in RPR mode. The fast software upgrade is done in RPR mode to avoid image incompatibility during the upgrade process.

To bring the new images into service, the standby MSFC is forced to switch over by taking the active MSFC out of service; the standby MSFC now becomes the active MSFC. Next, the out-of-service MSFC is allowed to boot; it becomes the standby MSFC but runs the newly upgraded image. The new image is running on both MSFCs, and the standby MSFC comes up in the hot-standby state. At this point, the system is now running in SSO mode because both MSFCs are running the same image version.

Note You may restore the original roles of the MSFCs (their active and passive status) by forcing another switchover in which the standby MSFC becomes the active MSFC.

To perform fast software upgrade procedure, perform these steps:

Step 1 Copy the new image to both MSFCs.

Step 2 Set the boot variables and save the configuration by entering the **write memory** command.

Step 3 Reset the standby MSFC and bring it back online, running the new image. Ensure the standby MSFC is fully online by entering the **show redundancy states** command.

Step 4 Do a manual switchover by entering the **redundancy force-switchover** command. The standby MSFC becomes the newly active MSFC running the new image. Because the system was in RPR mode before the switchover, the installed modules are reset and redownloaded with the new software during the switchover.

Step 5 After the new standby MSFC reboots and comes back online, both MSFCs and installed modules are running the new version of the software.

Upgrading to SSO from Single Router or Dual Router Modes

Note This upgrade interrupts service. The actual downtime varies based on the configuration of the switch, but it will not be longer than the time it takes to boot the system and come online.

Note When upgrading to SSO from SRM or DRM, you must save your configuration before performing the upgrade. DRM configurations generate parse errors when the system reloads the new image. After the upgrade, DRM configurations need to be reconfigured for use with SSO.

Cisco IOS software prior to Cisco IOS Release 12.2(18)SXF is either SRM and/or DRM capable but does not support upgrading to SSO. These software images cannot be upgraded using the fast software upgrade procedure. To upgrade this software, you are required to load the new images on each of the MSFCs, and then simultaneously boot both MSFCs.

After the new images have been loaded on the MSFCs, you must reboot the system to load the new images. During this boot time, the switch is offline and you will not see the benefits of SSO until the new images are loaded.

Mixed-Mode Operation

If you make a mistake when upgrading the software, it may result in a mixed-mode situation in which an SSO-based image is running on one MSFC and an SRM and/or DRM based image is running on the other MSFC. This situation could lead to system stability problems.

In this mixed mode, if the SSO-based image is running on the active MSFC, the active MSFC will boot completely, coming up in a simplex (nonredundant) state. The SRM and/or DRM based image will also boot but will remain in a standby state.

Another example of a mixed-mode upgrade scenario is when the SRM and/or DRM image is running on the active MSFC and the SSO-based image is running on the standby MSFC. In this mode, the active MSFC running the SRM and/or DRM image will boot completely, but the SSO-based image running on the standby MSFC will incorrectly determine that it is the active MSFC and will try to boot as the active MSFC. When the inventory message is received from the supervisor engine indicating it should be the standby MSFC, it will report an MSFC role mismatch error and reload itself. This problem can happen whenever an SRM, DRM, or boothelper image is running on the active MSFC and you try to load an SSO capable image on the standby MSFC.

To correct both scenarios, you must either upgrade the SRM and/or DRM software to the same level as the SSO-based software, or downgrade the SSO-based software to the level of the SRM and/or DRM image.

CHAPTER 25

Modifying the Switch Boot Configuration

This chapter describes how to modify the switch boot configuration on the Catalyst 6500 series switches, including the BOOT environment variable, the CONFIG_FILE environment variable, and the configuration register.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Understanding How the Switch Boot Configuration Works

These sections describe how the boot configuration works:

Understanding the Boot Process

The boot process involves two software images: ROM monitor and supervisor engine system code. When you power up or reset the switch, the ROM-monitor code is executed. Depending on the nonvolatile RAM (NVRAM) configuration, the switch either stays in ROM-monitor mode or loads the supervisor engine system code.

Two user-configurable parameters determine how the switch boots: the configuration register and the BOOT environment variable. The configuration register is described in the "Understanding the Configuration Register" section on page 25-2. The BOOT environment variable is described in the "Understanding the BOOT Environment Variable" section on page 25-3.

Understanding the ROM Monitor

The ROM-monitor code executes upon switch power up, reset, or when a fatal exception occurs. The system enters ROM-monitor mode if the switch does not find a valid system image, if the NVRAM configuration is corrupted, or if the configuration register is set to enter ROM-monitor mode. From ROM-monitor mode, you can manually load a system image from flash memory, from a network server file, or from bootflash.

You can enter ROM-monitor mode by restarting the switch and then pressing the **Break** key during the first 60 seconds of startup. If you are connected through a terminal server, you can escape to the Telnet prompt and enter the **send break** command to enter ROM-monitor mode.

Note The **Break** key is always enabled for 60 seconds after rebooting the system, regardless of whether the configuration-register setting has the **Break** key disabled.

The following functionality is built into the ROM monitor:

- Power-on confidence test
- Hardware initialization
- Boot capability (allows manual boot and autoboot)
- Debug utility and crash analysis
- Monitor call interface (EMT calls—the ROM monitor provides information and some functionality to the running system images through EMT calls)
- File system (the ROM monitor knows the simple file system and supports the newly developed file system through the dynamic linked file system library [MONLIB])
- Exception handling

Understanding the Configuration Register

The configuration register determines whether the switch loads an operating system image and where the system image is stored. The configuration register boot field determines if and how the ROM monitor loads a supervisor engine system image at startup. You can modify the boot field to force the switch to boot a particular system image at startup instead of using the default system image.

The lowest four bits (bits 3, 2, 1, and 0) of the 16-bit configuration register form the boot field. The default boot field value is 0x10F. The possible configuration register boot field settings are as follows:

- When the boot field equals 0000, the switch does not load a system image. Instead, it enters ROM-monitor mode from which you can enter ROM-monitor commands to load a system image manually.
- When the boot field equals 0001, the switch loads the first valid system image found in onboard flash memory.
- When the boot field equals a value between 0010 and 1111, the switch loads the system image that is specified by the **boot system** commands in the NVRAM configuration. It attempts to boot the image in the order in which you entered the **boot system** commands. If it cannot boot any image in the BOOT environment variable list, the switch remains in ROM-monitor mode. The exact booting sequence is defined by the ROM monitor.

The other bits in the configuration register function as follows when set:

- Bit 5 (0x0020)—Enables CONFIG_FILE recurrence.
- Bit 6 (0x0040)—Causes system software to clear NVRAM contents.
- Bit 7 (0x0080)—Enables OEM bit (not used).
- Bit 8 (0x0100)—Disables break.
- Bit 9 (0x0200)—Uses secondary bootstrap (not used by the ROM monitor).
- Bit 10 (0x0400)—Provides IP broadcast with all zeros (not used).
- Bits 11/12 (0x0800/0x1000)—Provide console line speed: 0/0=9600, 0/1=1200, 1/0=4800, 1/1=2400 (default is 9600).
- Bit 13 (0x2000)—Boots default flash software if network boot fails (not used).
- Bit 14 (0x4000)—IP broadcasts do not have network numbers (not used).
- Bit 15 (0x8000)—Enables diagnostic messages and ignores NVRAM contents (not used).

Understanding the BOOT Environment Variable

The BOOT environment variable specifies a list of image files on the various devices from which the switch can boot at startup.

You can add several images to the BOOT environment variable to provide a fail-safe boot configuration. If the first file fails to boot the switch, subsequent images that are specified in the BOOT environment variable are tried until the switch boots or there are no additional images to attempt to boot. If there is no valid image to boot, the system enters ROM-monitor mode where you can manually specify an image to boot.

The system stores and executes images in the order in which you added them to the BOOT environment variable. If you want to change the order in which the images are tried at startup, you can either prepend and clear images from the BOOT environment variable to attain the desired order or you can clear the entire BOOT environment variable and then redefine the list in the desired order.

Understanding the CONFIG_FILE Environment Variable

You can use the CONFIG_FILE environment variable to specify a list of configuration files (auto-config files) on the various devices to use to configure the switch at startup. You can specify the following functions:

- Nonrecurring—When you add a list of configuration files to the CONFIG_FILE environment variable, the next time that the switch is restarted, the system erases the configuration in NVRAM and uses the specified files to configure the switch. The CONFIG_FILE environment variable is cleared before the switch is configured. Nonrecurring is the default setting.
- Recurring—When you add a list of configuration files to the CONFIG_FILE environment variable, the list is stored indefinitely in NVRAM. Each time that the switch is restarted, the system erases the configuration in NVRAM and configures the switch using the configuration files that are specified. The CONFIG_FILE environment variable is not cleared.

  For information on specifying recurrence or nonrecurrence, see the "Setting CONFIG_FILE Recurrence" section on page 25-7.
- Overwrite—When you add a list of configuration files to the CONFIG_FILE environment variable, overwriting means that the NVRAM configuration will be cleared before executing the configuration files. Overwrite is the default setting.
- Append—Append means that the configuration files will be executed without first clearing NVRAM.

  For information on specifying overwriting or appending, see the "Setting CONFIG_FILE Overwrite" section on page 25-8.
- Sync enable—Enables synchronization to force the configuration files to synchronize automatically to the standby supervisor engine. The file(s) are kept consistent with what is on the active supervisor engine.
- Sync disable—Disables synchronization.

  For information on specifying synchronization, see the "Setting CONFIG_FILE Synchronization" section on page 25-8.

Tip You can alter the CONFIG_FILE environment variable or change its other properties by using the commands in the configuration files that configure the switch at startup.

You can add multiple configuration files to the CONFIG_FILE environment variable. The specified files can be any valid configuration file that is stored on a local flash device (bootflash: or slot0:).

When the switch boots up, if any of the files that are specified in the CONFIG_FILE environment variable are valid configuration files, the configuration in NVRAM is erased and the system uses the specified configuration file to configure the switch. If multiple valid configuration files are specified, each configuration file is executed in the order in which it appears in the CONFIG_FILE environment variable.

If any specified file is not a valid configuration file, the entry is skipped and subsequent files are tried until there are no additional, specified images. If no valid configuration file is specified, the system retains the last configuration that is stored in NVRAM.

Default Switch Boot Configuration

Table 25-1 shows the default switch boot configuration.

Table 25-1 Default Switch Boot Configuration

Feature	Default Configuration
Configuration register value	0x10f
Boot method	System boots from the image that is specified in the BOOT environment variable
ROM-monitor console port baud rate	9600 baud
ignore-config parameter	Disabled
BOOT environment variable	Empty
CONFIG_FILE environment variable	slot0:switch.cfg
CONFIG_FILE recurrence configuration register parameter	Nonrecurring
CONFIG_FILE overwrite configuration register parameter	Overwrite
CONFIG_FILE synchronization configuration register parameter	Synchronization disabled

Setting the Configuration Register

Note The configuration register settings are not copied automatically to a redundant supervisor engine. You must set the configuration register separately for each supervisor engine in the switch.

These sections describe how to modify the configuration register:

- Setting the Boot Field in the Configuration Register, page 25-6
- Setting the ROM-Monitor Console-Port Baud Rate, page 25-6
- Setting CONFIG_FILE Recurrence, page 25-7
- Setting CONFIG_FILE Overwrite, page 25-8
- Setting CONFIG_FILE Synchronization, page 25-8
- Setting the Switch to Ignore the NVRAM Configuration, page 25-9
- Setting the Configuration Register Value, page 25-10

Setting the Boot Field in the Configuration Register

You can determine the boot method that the switch will use at the next startup by setting the boot field in the configuration register. This command affects only the configuration register bits that control the boot field and leaves the remaining bits unaltered. The following boot methods are supported:

- ROM monitor—Enter the **rommon** keyword to force the switch to remain in ROM-monitor mode at startup.
- Bootflash—Enter the **bootflash** keyword to cause the switch to boot from the first image that is stored in the onboard flash memory.
- System—Enter the **system** keyword to boot from the image that is specified in the BOOT environment variable (the default).

Note We recommend that you use only the **rommon** and **system** keywords with the **set boot config-register boot** command.

To set the configuration register boot field, perform this task in privileged mode:

Task	Command
Set the boot field in the configuration register.	**set boot config-register boot** {**rommon** | **bootflash** | **system**} [*mod*]

This example shows how to set the boot field in the configuration register:

```
Console> (enable) set boot config-register boot rommon
Configuration register is 0x0
ignore-config: disabled
auto-config: non-recurring
console baud: 9600
boot: the ROM monitor
Console> (enable)
```

Setting the ROM-Monitor Console-Port Baud Rate

You can set the console-port baud rate that is used by the ROM monitor. The new baud rate is used the next time that the switch is restarted. This command affects only the configuration register bits that control the baud rate and leaves the remaining bits unaltered.

Note The baud rate that is specified in the configuration register is used by the ROM monitor only and is different from the baud rate that is specified by the **set system baud** command.

To set the ROM-monitor console-port baud rate in the configuration register, perform this task in privileged mode:

Task	Command
Set the ROM-monitor console-port baud rate in the configuration register.	**set boot config-register baud** {**1200** | **2400** | **4800** | **9600**} [*mod*]

This example shows how to set the ROM-monitor console-port baud rate in the configuration register to 2400:

```
Console> (enable) set boot config-register baud 2400
Configuration register is 0x1800
ignore-config: disabled
auto-config: non-recurring
console baud: 2400
boot: the ROM monitor
Console> (enable)
```

Setting CONFIG_FILE Recurrence

By default, when you set the CONFIG_FILE environment variable, the list of configuration files to use at startup is retained only until the next time that the switch is restarted.

You can cause the system software to retain the CONFIG_FILE environment variable settings indefinitely so that each time that the switch is restarted, the specified configuration files are used to configure the switch.

This command affects only the configuration register bit that controls whether the CONFIG_FILE environment variable settings are recurring or nonrecurring. The remaining configuration register bits are unaltered.

Caution With the CONFIG_FILE environment variable set to **recurring**, the current configuration in NVRAM is erased each time that the switch is restarted and the switch is configured using the specified configuration files. With the CONFIG_FILE environment variable set to **non-recurring**, the current configuration in NVRAM is erased at the next restart and the switch is configured using the specified configuration files. The NVRAM configuration is retained after subsequent restarts (unless you again set the CONFIG_FILE variable).

To set the switch to retain the current CONFIG_FILE environment variable indefinitely, perform this task in privileged mode:

Task	Command
Set the switch to retain the current CONFIG_FILE environment variable indefinitely.	**set boot config-register auto-config** {**recurring** | **non-recurring**}

This example shows how to set the switch to retain the current CONFIG_FILE environment variable indefinitely:

```
Console> (enable) set boot config-register auto-config recurring
Configuration register is 0x1820
ignore-config: disabled
auto-config: recurring, overwrite, sync disabled
console baud: 2400
boot: the ROM monitor
Console> (enable)
```

Setting CONFIG_FILE Overwrite

This command allows you to specify if the auto-config file should be used to overwrite the NVRAM configuration or if the file configuration should be appended to what is currently in NVRAM. Overwriting means that the NVRAM configuration will be cleared before executing the auto-config file; appending means that the auto-config file will be executed without first clearing NVRAM. The default is **overwrite**.

To specify if the auto-config file should be used to overwrite the NVRAM configuration or if the file configuration should be appended to what is currently in NVRAM, perform this task in privileged mode:

Task	Command
Specify if the auto-config file should be used to overwrite the NVRAM configuration or if the file configuration should be appended to what is currently in NVRAM.	**set boot config-register auto-config** {**overwrite** | **append**}

This example shows how to specify that the auto-config file is used to overwrite the NVRAM configuration:

```
Console> (enable) set boot config-register auto-config overwrite
Configuration register is 0x12F
ignore-config: disabled
auto-config: recurring, overwrite, sync disabled
console baud: 9600
boot: image specified by the boot system commands
Console> (enable)
```

This example shows how to specify that the auto-config file is appended to what is currently in NVRAM:

```
Console> (enable) set boot config-register auto-config append
Configuration register is 0x12F
ignore-config: disabled
auto-config: recurring, append, sync disabled
console baud: 9600
boot: image specified by the boot system commands
Console> (enable)
```

Setting CONFIG_FILE Synchronization

The **set boot config-register auto-config sync** command allows you to enable synchronization to force the auto-config file(s) to synchronize automatically to the standby supervisor engine. The file(s) are kept consistent with what is on the active supervisor engine. The default is **disabled**. These events can trigger a synchronization check and a synchronization (if necessary):

- Changing the auto-config file(s) on either supervisor engine (if the file is deleted on the active supervisor engine, it is also deleted on the standby supervisor engine)
- Changing the boot string CONFIG_FILE variable setting
- Inserting a new supervisor engine
- System startup

The CONFIG_FILE variable from the active supervisor engine is made identical on the standby supervisor engine. Each auto-config file on the active supervisor engine is compared against each corresponding auto-config file on the standby supervisor engine. Two files are considered identical if their lengths and cyclic redundancy check (CRC) are the same. If a file on the standby supervisor engine is not identical to the file on the active supervisor engine, a new file is generated on the standby supervisor engine with the name of the file on the active supervisor engine. If a file with that name already exists on the standby supervisor engine, it is overwritten.

To enable or disable synchronization, perform this task in privileged mode:

Task	Command
Specify if synchronization should be enabled or disabled.	**set boot config-register auto-config sync** {**enable** | **disable**}

This example shows how to enable synchronization:

```
Console> (enable) set boot config-register auto-config sync enable
Configuration register is 0x12F
ignore-config: disabled
auto-config: recurring, append, sync enabled
console baud: 9600
boot: image specified by the boot system commands
Console> (enable)
```

This example shows how to disable synchronization:

```
Console> (enable) set boot config-register auto-config sync disable
Configuration register is 0x12F
ignore-config: disabled
auto-config: recurring, append, sync disabled
console baud: 9600
boot: image specified by the boot system commands
Console> (enable)
```

Setting the Switch to Ignore the NVRAM Configuration

You can cause the system software to ignore the configuration information that is stored in NVRAM the next time that the switch is restarted. The **set boot config-register ignore-config enable** command affects only the configuration register bits that control whether the switch ignores the NVRAM configuration and leaves the remaining bits unaltered. This command affects the next system restart only.

Caution Enabling the **ignore-config** parameter is the same as entering the **clear config all** command; that is, it clears the entire configuration that is stored in NVRAM the next time that the switch is restarted.

To set the switch to ignore the NVRAM configuration at the next startup, perform this task in privileged mode:

Task	Command
Set the switch to ignore the contents of NVRAM at startup.	**set boot config-register ignore-config enable**

This example shows how to set the switch to ignore the NVRAM configuration at the next startup:

```
Console> (enable) set boot config-register ignore-config enable
Configuration register is 0x1860
ignore-config: enabled
auto-config: recurring
console baud: 2400
boot: the ROM monitor
Console> (enable)
```

Setting the Configuration Register Value

To set the configuration register value, perform this task in privileged mode:

Task	Command
Set the configuration register.	**set boot config-register 0x***value* [*mod*]

This example shows how to set the configuration register value to 0x90f:

```
Console> (enable) set boot config-register 0x90f
Configuration register is 0x90f
ignore-config: disabled
auto-config: non-recurring
console baud: 4800
boot: image specified by the boot system commands
Console> (enable)
```

Setting the BOOT Environment Variable

Note The BOOT environment variable settings are not copied automatically to a redundant supervisor engine (if present). You must set the BOOT variable separately for each supervisor engine in the switch.

These sections describe how to modify the BOOT environment variable:

- Setting the BOOT Environment Variable, page 25-10
- Clearing the BOOT Environment Variable Settings, page 25-11

Setting the BOOT Environment Variable

To set the BOOT environment variable, perform this task in privileged mode:

Task	Command
Set the BOOT environment variable.	**set boot system flash** *device:*[*filename*] [**prepend**] [*mod*]

This example shows how to set the BOOT environment variable:

```
Console> (enable) set boot system flash bootflash:cat6000-sup.5-5-1.bin
BOOT variable = bootflash:cat6000-sup.5-5-1.bin,1;
Console> (enable) set boot system flash bootflash:cat6000-sup.4-5-2.bin
BOOT variable = bootflash:cat6000-sup.5-1-1.bin,1;bootflash:cat6000-sup.4-5-2.
bin,1;
Console> (enable) set boot system flash bootflash:cat6000-sup.5-2-1.bin prepend
BOOT variable = bootflash:cat6000-sup.5-2-1.bin,1;bootflash:cat6000-sup.5-5-1.
bin,1;bootflash:cat6000-sup.4-5-2.bin,1;
Console> (enable)
```

Clearing the BOOT Environment Variable Settings

To clear the entries from the BOOT environment variable, perform one of these tasks in privileged mode:

Task	Command
Clear a specific image from the BOOT environment variable.	**clear boot system flash** *device:*[*filename*] [*mod*]
Clear the entire BOOT environment variable.	**clear boot system all** [*mod*]

This example shows how to clear a specific entry from the BOOT environment variable:

```
Console> (enable) clear boot system flash bootflash:cat6000-sup.5-1-1.bin
BOOT variable = bootflash:cat6000-sup.5-2-1.bin,1;bootflash:cat6000-sup.4-5-2.
bin,1;
Console> (enable)
```

This example shows how to clear the entire BOOT environment variable:

```
Console> (enable) clear boot system all
BOOT variable =
Console> (enable)
```

Setting the CONFIG_FILE Environment Variable

These sections describe how to modify the CONFIG_FILE environment variable:

Setting the CONFIG_FILE Environment Variable

You can specify multiple configuration files with the **set boot auto-config** command by separating them with a semicolon (;). You must specify both the device name and the filename for each configuration file.

Note You cannot prepend or append the configuration files to the CONFIG_FILE environment variable. Entering the **set boot auto-config** command erases any list of configuration files that were previously specified using the **set boot auto-config** command.

To set the CONFIG_FILE environment variable, perform this task in privileged mode:

Task	Command
Set the CONFIG_FILE environment variable.	**set boot auto-config** *device:filename*[**;***device:filename*...]

This example shows how to set the CONFIG_FILE environment variable:

```
Console> (enable) set boot auto-config bootflash:generic.cfg;bootflash:6509_1_noc.cfg
CONFIG_FILE variable = bootflash:generic.cfg;bootflash:6509_1_noc.cfg
WARNING: nvram configuration may be lost during next bootup,
         and re-configured using the file(s) specified.
Console> (enable)
```

Clearing the CONFIG_FILE Environment Variable Settings

To clear the entries from the CONFIG_FILE environment variable, perform this task in privileged mode:

Task	Command
Clear the entries in the CONFIG_FILE environment variable.	**clear boot auto-config**

This example shows how to clear the entries in the CONFIG_FILE environment variable:

```
Console> (enable) clear boot auto-config
CONFIG_FILE variable =
Console> (enable)
```

Displaying the Switch Boot Configuration

To display the current configuration register, the BOOT environment variable, and the CONFIG_FILE environment variable settings, perform this task:

Task	Command
Display the current configuration register, the BOOT environment variable, and the CONFIG_FILE environment variable settings.	**show boot** [*mod*]

This example shows how to display the current configuration register, the BOOT environment variable, and the CONFIG_FILE environment variable settings:

```
Console> (enable) show boot
BOOT variable = bootflash:cat6000-sup.5-2-1.bin,1;
CONFIG_FILE variable = bootflash:generic.cfg;bootflash:6509_1_noc.cfg

Configuration register is 0x12f
ignore-config: disabled
auto-config: recurring
console baud: 9600
boot: image specified by the boot system commands

Console> (enable)
```

CHAPTER 26

Working With the Flash File System

This chapter describes how to use the flash file system on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Understanding How the Flash File System Works

The flash file system on a Catalyst 6500 series supervisor engine provides a number of useful commands to help you manage the software image and configuration files. The flash file system on the supervisor engine consists of the flash devices on which you can store the files:

- Supervisor Engine 1 and Supervisor Engine 2
  - **bootflash:** Onboard flash memory
  - **slot0:** Linear Flash PC card (PCMCIA slot)
  - **disk0:** ATA Flash PC card (PCMCIA slot)
- Supervisor Engine 720
  - **bootflash:** Onboard flash memory
  - **disk0:** CompactFlash Type II card only (disk 0 slot)
  - **disk1:** CompactFlash Type II card (disk 1 slot)
- Supervisor Engine 32
  - **bootdisk:** Onboard flash memory
  - **disk0:** CompactFlash Type II card only (disk 0 slot)

Working with the Flash File System on the Switch

These sections describe how to work with the flash file system:

Setting the Default Flash Device

When you set the default flash device for the switch, the default device is assumed when you enter a flash file system command without specifying the flash device.

To set the default flash device, perform this task:

	Task	Command
Step 1	Set the default flash device for the switch.	**cd** [[*m/*][**bootflash:** | **slot0:**]]
Step 2	Verify the default flash device for the switch.	**pwd** [*mod*]

This example shows how to change the default flash device to **slot0:** and verify the default device:

```
Console> (enable) cd slot0:
Console> (enable) pwd
slot0
Console> (enable)
```

Setting the Text File Configuration Mode

When you use text file configuration mode, the switch stores its configuration as a text file in nonvolatile storage, either in NVRAM or flash memory. This text file consists of the commands that are entered by you to configure the various features. For example, if you disable a port, the command to disable that port will be in the text configuration file.

Because the text file only contains the commands that you have used to configure your switch, it typically uses less NVRAM or flash memory space than binary configuration mode. Because the text file in most cases requires less space, NVRAM is a good place to store the file. If the text file exceeds NVRAM space, it can also be saved to flash memory.

When operating in text file configuration mode, most user settings are not immediately saved to NVRAM; the configuration changes are only written to DRAM. You will need to enter the **write memory** command to store the configuration in nonvolatile storage.

Note The VLAN commands are not saved as part of the configuration file when the switch is operating in text mode with the VTP mode set to server.

To set the text file configuration mode, perform this task:

	Task	Command
Step 1	Set the file configuration mode for the system to text.	**set config mode text** {**nvram** | *device:file-id*}
Step 2	Verify the file configuration mode for the system.	**show config mode**
Step 3	Save the text file configuration.	**write memory**
Step 4	Display the current run-time configuration.	**show running-config all**
Step 5	Display the startup configuration that will be used after the next reset.	**show config**

This example shows how to configure the system to save its configuration as a text file in NVRAM, verify the configuration mode, and display the current run-time configuration:

```
Console> (enable) set config mode text nvram
Console> (enable) show config mode
Console> (enable) show running-config all
Console> (enable) show config
Console> (enable)
```

Setting the Text File Configuration Mode to Auto-Save

Use the **set config mode text auto-save** command to save the text configuration in NVRAM automatically. Use the **interval** keyword to set the time interval between the occurrences of saving the text configuration in NVRAM. You can specify the time interval between the occurrences of saving the text configuration in NVRAM even if the system is in binary mode. If you do not specify the number of minutes after entering the **interval** keyword, the interval is set to the default of 30 minutes.

Note In software release 8.4(1) and earlier releases, valid values for the *mins* argument are from 30 minutes to 35000 minutes. In release 8.4(2) and subsequent releases, valid values for the *mins* argument are from 1 minute to 35000 minutes.

The text configuration is not saved automatically in NVRAM unless auto-save is enabled. To enable auto-save, you must first set the system configuration mode to text and configure the system to save the text configuration in NVRAM. If the system configuration mode is set to binary mode, you cannot enable auto-save.

To set the text file configuration mode to auto-save, perform this task:

	Task	Command
Step 1	Set the file configuration mode for the system to text.	**set config mode text** {**nvram** | *device:file-id*}
Step 2	Specify the **auto-save** keyword.	**set config mode text auto-save** {**enable** | **disable**}
Step 3	(Optional) Configure the **interval** keyword.	**set config mode text auto-save interval** *mins*
Step 4	Verify the file configuration mode for the system.	**show config mode**
Step 5	Save the text file configuration.	**write memory**
Step 6	Display the current run-time configuration.	**show running-config all**
Step 7	Display the startup configuration that will be used after the next reset.	**show config**

This example shows how to set the configuration mode to text and designate the location and filename for saving the text configuration file:

```
Console> (enable) set config mode text bootflash:switch.cfg
Binary system configuration has been deleted from NVRAM. Configuration mode set to text.
Use the write memory command to save configuration
changes. System configuration file set to: bootflash:switch.cfg
The file specified will be used for configuration during the next bootup.
Console> (enable)
```

This example shows how to enable auto-save when the configuration is set to text mode and the system is configured to save the text configuration in NVRAM:

```
Console> (enable) set config mode text auto-save enable
auto-save feature has been enabled
auto-save feature has started
Please do a write mem manually if you plan to reboot the switch or any card before first
expiry of the timer
Console> (enable)
```

This example shows the message that is displayed if you attempt to enable auto-save when the configuration is not set to text mode and the system is not configured to save the text configuration in NVRAM:

```
Console> (enable) set config mode text auto-save enable
auto-save cannot be enabled unless config mode is set to text and config file is stored in
nvram.
Use the 'set config mode text nvram' command to enable automatic saving of the system
configuration to nvram
Console> (enable)
```

This example shows how to set the interval between the saves to 2880 minutes:

```
Console> (enable) set config mode text auto-save interval 2880
auto-save interval set to 2880 minutes
Console> (enable)
```

This example shows how to set the interval between the saves to the default setting of 30 minutes:

```
Console> (enable) set config mode text auto-save interval
auto-save interval set to 30 minutes
Console> (enable)
```

Listing the Files on a Flash Device

To list the files on a flash device, perform one of these tasks:

Task	Command
Display a list of files on a flash device.	**dir** [[*m*/]*device:*][*filename*]
Display a list of the deleted files on a flash device.	**dir** [[*m*/]*device:*][*filename*] **deleted**
Display a list of all files on a flash device, including the deleted files.	**dir** [[*m*/]*device:*][*filename*] **all**
Display a detailed list of files on a flash device.	**dir** [[*m*/]*device:*][*filename*] **long**

This example shows how to list the files on the default flash device:

```
Console> (enable) dir
-#- -length- -----date/time------ name
  4  3134688 Mar 15 1999 08:27:01 cat6000-sup.5-2-1-CSX.bin
  5  3231989 Jan 24 1999 12:04:40 cat6000-sup.5-1-1-CSX.bin
  6      135 Feb 17 1999 11:30:05 dns_config.cfg

1213952 bytes available (6388224 bytes used)
Console> (enable)
```

This example shows how to list the files on a flash device other than the default device:

```
Console> (enable) dir slot0:
-#- -length- -----date/time------ name
  1  3209261 Jun 16 1998 13:18:19 cat6000-sup.5-2-1-CSX.bin
  2      135 Jul 17 1998 11:32:53 dns-config.cfg
  3  3231989 Jul 17 1998 16:54:23 cat5000-sup3.4-1-2.bin
  4     8589 Jul 17 1998 17:02:52 6000_config.cfg

9933504 bytes available (6450496 bytes used)
Console> (enable)
```

This example shows how to list the deleted files on the default flash device:

```
Console> (enable) dir deleted
-#- ED --type-- --crc--- -seek-- nlen -length- -----date/time------ name
  1 .D ffffffff 81a027ca   41bdc   22     7004 Apr 01 1998 15:27:45 5002.config.
4.1.98.cfg
  2 .D ffffffff ccce97a3   43644   23     6630 Apr 01 1998 15:36:47 5002.default
.config.cfg
  3 .D ffffffff 81a027ca   45220   15     7004 Apr 19 1998 10:05:59 5002_config.
cfg

1213952 bytes available (6388224 bytes used)
Console> (enable)
```

Copying Files

To copy a file, perform one of these tasks in privileged mode:

Task	Command
Copy a flash file to a TFTP server, rcp server, flash memory, another flash device, or to the running configuration.	**copy** *file-id* {**tftp** | **rcp** | **flash** | *file-id* | **config**}
Copy a file from a TFTP server, rcp server to flash memory, to a flash device, or to the running configuration.	**copy** {**tftp** | **rcp**} {**flash** | *file-id* | **config**}
Copy a file from flash memory to a TFTP server, rcp server, to a flash device, or to the running configuration.	**copy flash** {**tftp** | **rcp** | *file-id* | **config**}
Copy the running configuration to flash memory, another flash device, to a TFTP server, or rcp server.	**copy config** {**flash** | *file-id* | **tftp** | **rcp**}

This example shows how to copy a file from the default flash device to another flash device:

```
Console> (enable) copy cat6000-sup.5-2-1-CSX.bin slot0:

13174216 bytes available on device slot0, proceed (y/n) [n]? y
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCcccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
ccccccccccccCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
File has been copied successfully.
Console> (enable)
```

This example shows how to copy a file from a TFTP server to the running configuration:

```
Console> (enable) copy tftp config
IP address or name of remote host []? 172.20.52.3
Name of file to copy from []? dns_config.cfg

Configure using tftp:dns_config.cfg (y/n) [n]? y
/
Finished network download.  (135 bytes)
>>
>> set ip dns server 172.16.10.70 primary
172.16.10.70 added to DNS server table as primary server.
>> set ip dns server 172.16.10.140
172.16.10.140 added to DNS server table as backup server.
>> set ip dns enable
DNS is enabled
>> set ip dns domain corp.com
Default DNS domain name set to corp.com
Console> (enable)
```

This example shows how to download a configuration file from a TFTP server for storage on a flash device:

```
Console> (enable) copy tftp flash
IP address or name of remote host []? 172.20.52.3
Name of file to copy from []? dns-config.cfg
Flash device [slot0]?
Name of file to copy to [dns-config.cfg]?

9932056 bytes available on device slot0, proceed (y/n) [n]? y
/
File has been copied successfully.
Console> (enable)
```

This example shows how to copy the running configuration to flash memory:

```
Console> (enable) copy config flash
Flash device [bootflash]? slot0:
Name of file to copy to []? 6000_config.cfg

Upload configuration to slot0:6000_config.cfg
9942096 bytes available on device slot0, proceed (y/n) [n]? y
.....
..........
.......

..........
...........
..

Configuration has been copied successfully.
Console> (enable)
```

This example shows how to upload a configuration file on a flash device to a TFTP server:

```
Console> (enable) copy slot0:6000_config.cfg tftp
IP address or name of remote host []? 172.20.52.3
Name of file to copy to [6000_config.cfg]?
/
File has been copied successfully.
Console> (enable)
```

This example shows how to upload an image from a remote host into flash using rcp:

```
Console> (enable) copy rcp flash
IP address or name of remote host []? 172.20.52.3
Name of file to copy from []? 6000_config.cfg
Flash device [bootflash]?
Name of file to copy to [6000_config.cfg]?

4369664 bytes available on device bootflash, proceed (y/n) [n]? y
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCC
File has been copied successfully.
Console> (enable)
```

Deleting Files

Caution If you enter the **squeeze** command on a flash device, you cannot restore the files that were deleted prior to the **squeeze** command.

To delete the files on a flash device, perform this task in privileged mode:

	Task	Command
Step 1	Delete a file on a flash device.	**delete** [[*m*/]*device:*]*filename*
Step 2	If desired, permanently remove all deleted files on the flash device (this operation can take several minutes to complete).	**squeeze** [*m*/]*device:*
Step 3	Verify that the files are deleted.	**dir** [[*m*/]*device***:**][*filename*]

This example shows how to delete a file from a flash device:

```
Console> (enable) delete dns_config.cfg
Console> (enable)
```

This example shows how to remove all deleted files from a flash device permanently:

```
Console> (enable) squeeze slot0:
All deleted files will be removed, proceed (y/n) [n]? y
Squeeze operation may take a while, proceed (y/n) [n]? y
Erasing squeeze log
Console> (enable)
```

Restoring Deleted Files

You must specify the index number of a deleted file to identify the file to undelete. The index number for each file appears in the first column of the **dir** command output. A file cannot be undeleted if a valid file with the same name already exists. Instead, you must delete the existing file and then undelete the desired file. A file can be deleted and undeleted up to 15 times.

To restore the deleted files on a flash device, perform this task in privileged mode:

	Task	Command
Step 1	Identify the index number of the deleted files on the flash device.	**dir** [[*m*/]*device***:**][*filename*] **deleted**
Step 2	Restore a file on a flash device.	**undelete** *index* [[*m*/]*device*:]
Step 3	Verify that the file is restored.	**dir** [[*m*/]*device***:**][*filename*]

This example shows how to restore a deleted file:

```
Console> (enable) dir deleted
-#- ED --type-- --crc--- -seek-- nlen -length- -----date/time------ name
6 .D ffffffff 42da7f71  657a00   14      135 Jul 17 1998 11:30:05 dns_config.cfg

1213952 bytes available (6388224 bytes used)
Console> (enable) undelete 6
```

```
Console> (enable) dir
-#- -length- -----date/time------ name
  4  3134688 Apr 27 1998 08:27:01 cat6000-sup.5-2-1.bin
  5  3231989 Jun 24 1998 12:04:40 cat6000-sup.5-2-1.bin
  6      135 Jul 17 1998 11:30:05 dns_config.cfg

1213952 bytes available (6388224 bytes used)
Console> (enable)
```

Verifying a File Checksum

To verify the checksum of a file on a flash device, perform this task in privileged mode:

Task	Command
Verify the checksum of a file on a flash device.	**verify** [[*m*/]*device:*] *filename*

This example shows how to verify the checksum of a file:

```
Console> (enable) verify cat6000-sup.5-2-1-CSX.bin
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCC
File bootflash:cat6000-sup.5-2-1-CSX.bin verified OK
Console> (enable)
```

Formatting a Flash Device

Some flash devices require formatting before they can be used. You can reserve up to 16 spare sectors for use when other sectors fail (by default, none are reserved). If you do not reserve spare sectors and later some sectors fail, you will have to reformat the entire flash memory, which erases all existing data.

Note Supervisor Engine 2 and Supervisor Engine 1 do not support the same Flash PC card format. To use a Flash PC card with Supervisor Engine 2, format the card with Supervisor Engine 2. To use a Flash PC card with Supervisor Engine 1, format the card with Supervisor Engine 1.

Note The Flash PC cards that are formatted on Supervisor Engine 1 or on a route-switch processor (RSP)-based Cisco 7500 series router are interchangeable if the router is running software at least at the same level as the supervisor engine. You cannot use the Flash PC cards that are formatted on a route processor (RP)-based Cisco 7000 series router without reformatting.

When you format a flash device, you can specify the *monlib* file (the ROM monitor library), which the ROM monitor uses to access the files in the flash file system. The monlib file is also compiled into the software image.

In the **format** command syntax, use the *device2* argument to specify the device that contains the monlib file to use. If you omit the entire *device2* argument, the switch formats the device using the monlib file that is bundled with the software. If you omit just the device name (*device2*) from the [[*device2:*][*monlib-filename*]] argument, the switch formats the device using the named monlib file from the default flash device. If you omit the *monlib-filename* from the [[*device2:*][*monlib-filename*]] argument, the switch formats the device using the monlib file from *device2*. If you specify the entire [[*device2:*][*monlib-filename*]] argument, the switch formats the device using the specified monlib file from the specified device. If the switch cannot find a monlib file, it terminates the formatting process.

Note If the flash device has a volume ID, you must provide the volume ID to format the device. The volume ID is displayed using the **show flash** *m/device*: **filesys** command.

To format a flash device, perform this task in privileged mode:

Task	Command
Format a flash device.	**format** [**spare** *spare-number*] [*m/*]*device1:* [[*device2:*] [*monlib-filename*]]

This example shows how to format the flash device in slot0:

```
Console> (enable) format slot0:
All sectors will be erased, proceed (y/n) [n]?y
Enter volume id (up to 31 characters):
Formatting sector 1
Format device slot0 completed.
Console> (enable)
```

CHAPTER 27

Working with System Software Images

This chapter describes how to work with system software image files on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Software Image Naming Conventions

The software images on the Catalyst 6500 series switches use the following naming conventions (software release 7.3(1) images for a Supervisor Engine 2 are used in the examples):

- 7.3(1) flash image (standard)—cat6000-sup2k8.7-3-1.bin
- 7.3(1) flash image (CiscoView)—cat6000-sup2cvk8.7-3-1.bin
- 7.3(1) flash image (Secure Shell)—cat6000-sup2k9.7-3-1.bin
- 7.3(1) flash image (Secure Shell and CiscoView)—cat6000-sup2cvk9.7-3-1.bin

Note The sup2cvk8, sup2k9, and sup2cvk9 designations are as follows: sup2cvk8 is a CiscoView image, sup2k9 is a Secure Shell image, and sup2cvk9 is a Secure Shell and CiscoView image.

Upgrading the EPLD Images

Note The supervisor engine EPLD upgrades are supported only on Supervisor Engine 2 and Supervisor Engine 720. The nonsupervisor engine module (switching modules and service modules) EPLD upgrades are supported using Supervisor Engine 1, Supervisor Engine 2, or Supervisor Engine 720.

The EPLD image for Supervisor Engine 2 and Supervisor Engine 720 is included in the Catalyst supervisor engine software image. The EPLD image for the nonsupervisor engine modules is provided in a separate downloadable image.

Upgrading the Supervisor Engine EPLD Image

The supervisor engine EPLD upgrade is performed automatically when you reset or power cycle the switch. You can use the **set system supervisor-update** command to modify the EPLD upgrade process. By default, the supervisor engine EPLD upgrade is disabled. In the **automatic** mode, the system checks the version level of the bundled EPLD image and performs the upgrade if the bundled EPLD image version is greater than the existing version. If you specify the **force** keyword, the system upgrades the existing EPLD image with the bundled EPLD image regardless of the version level. After a forced upgrade, the configuration reverts back to the **automatic** default setting. The **disable** keyword disables the automatic EPLD upgrade process.

To upgrade the supervisor engine EPLD image, perform this task in privileged mode:

	Task	Command
Step 1	Upgrade the supervisor engine EPLD image.	**set system supervisor-update** {**automatic** | **disable** | **force**}
Step 2	Verify the supervisor engine EPLD image upgrade.	**show system supervisor-update**

This example shows how to specify the **automatic** keyword for the EPLD upgrades:

```
Console> (enable) set system supervisor-update automatic
Down-rev supervisor EPLD's will be re-programmed next reset.
Console> (enable)
```

This example shows how to specify the **force** keyword for the EPLD upgrades:

```
Console> (enable) set system supervisor-update force
Supervisor EPLD's will synchronize to the image bundle during the next reset.
Console> (enable)
```

This example shows how to disable the EPLD upgrades:

```
Console> (enable) set system supervisor-update disable
Supervisor EPLD update during reset is disabled.
Console> (enable)
```

This example shows how to display the EPLD upgrade configuration:

```
Console> (enable) show system supervisor-update
Supervisor EPLD update: disabled
Console> (enable)
```

Upgrading the Nonsupervisor Engine Module EPLD Images

Caution Do not power off or reset the switch or module during the upgrade process. Powering off or resetting the switch or module could leave the module in an unusable state.

Note Before you begin the procedures in this chapter, make sure that you have downloaded the new EPLD upgrade image to the supervisor engine flash memory (bootflash: or slot0:).

You can upgrade the nonsupervisor engine module EPLD image by using the **download** command with the **epld** keyword. If you enter the **download epld** *file* command without specifying a module, the new EPLD image is downloaded to all compatible modules where the new EPLD image version is greater than the existing version on the module. If you use the **download epld** *file mod* command with the **force** keyword, the existing EPLD image on a module is upgraded with the new EPLD image regardless of the existing version level.

To upgrade the EPLD on the nonsupervisor engine modules (switching modules and service modules), perform this task in privileged mode:

	Task	Command
Step 1	Upgrade the nonsupervisor engine EPLD image.	**download epld** *file* **download epld** *file mod* [**force**]
Step 2	Verify the EPLD upgrade process configuration.	**show version epld** *mod*

This example shows how to upgrade the EPLD image on the module in slot 5:

```
Console> (enable) download epld aq_cr128_art.bin 5 force
CCCCCC
Device found requiring upgrade in slot 5.

##########################################################
#                   W A R N I N G                        #
#                                                        #
# Any disruptions to the module during programming may   #
# leave the module or system in an inconsistent state.   #
# Please ensure that the system or module does not get   #
# switched off or reset during the programming process.  #
# Programming may take a minute or two, depending on     #
# the number of devices updated.  Please wait for the    #
# module to come back online before continuing.          #
#                                                        #
#                   W A R N I N G                        #
##########################################################
This command may reset module 5.
Updating fabric modules may significantly affect system performance while the update is
occurring.

Do you wish to update the devices in slot 5 (y/n) [n]? y

Updating programmable devices in slot 5. This may take a minute...
Programming successful, updating EPLD revisions.
2002 Aug 09 06:32:22 %SYS-4-NVLOG:EpldUpdate:Module 5 EPLD A updated from rev 1 to rev 1
Waiting for module to come online.
..........2002 Aug 09 06:32:33 %SYS-5-MOD_OK:Module 5 is online
.

##############################################################################

               E P L D   P R O G R A M M I N G   C O M P L E T E

    Found 1 devices requiring upgrades, 1 attempted, 1 updated, 0 failed

##############################################################################
Console> (enable) 2002 Aug 09 06:32:34 %SYS-4-NVLOG:EpldUpdate:Module 5 EPLD A s
prom updated to rev 1
Console> (enable)
```

Comparing File Transfer Protocols

Table 27-1 compares the supported file transfer protocols.

Table 27-1 Comparison of File Transfer Protocols

Requirement	TFTP	RCP	FTP	SCP	SFTP
Username needed	No	Yes	Yes	Yes	Yes
Password needed	No	No	Yes	Yes[1]	Yes
Can run as a client	Yes	Yes	Yes	Yes	Yes
Can run as a server	Yes	No	No	No	No
Secure authentication	N/A	No	No	Yes	Yes
Secure file transfer	No	No	No	Yes	Yes
Available in the standard flash images	Yes	Yes	Yes	No	No
Available in crypto images	Yes	Yes	Yes	Yes	Yes

1. SCP authentication through ".shosts" can be used to avoid login but most SSH publications recommend not using it due to security concerns.

Downloading the Software Images Using FTP or TFTP

These sections describe how to download the system software images to the switch supervisor engine and to the intelligent modules:

- Understanding How FTP and TFTP Software Image Downloads Work, page 27-5
- Specifying the FTP Username and Password, page 27-6
- Preparing to Download an Image Using FTP or TFTP, page 27-7
- Downloading the Supervisor Engine Images Using FTP or TFTP, page 27-7
- Downloading the Switching Module Images Using FTP or TFTP, page 27-8
- FTP and TFTP Download Procedures Example, page 27-9

Understanding How FTP and TFTP Software Image Downloads Work

You can download the system software images to the switch using the File Transfer Protocol (FTP) or Trivial File Transfer Protocol (TFTP). TFTP allows you to download the system image files over the network from a TFTP server. FTP allows you to download the system image files over the network from a FTP server.

Some modules, such as the ATM modules, have their own onboard flash memory. When you download a software image file, the switch checks the header of the image file to determine the type of software image.

Depending on the type of software image that you are downloading, one of the following occurs:

- Supervisor engine software image—The image file is downloaded to the supervisor engine flash memory. You can store multiple image files on the flash memory system devices (such as boot flash and Flash PC cards).
- Intelligent module software images—If you specified a module number, the image file is downloaded to the specified module only (if the image file is designed for the specified module type). If you do not specify a module number, the image file is downloaded to every module of the appropriate type. The file is relayed packet by packet to the appropriate modules using the Inter-Process Communications protocol that is internal to the system, with communication taking place across the switching bus. Downloading a software image to multiple modules significantly speeds up the process of updating the software on multiple modules of the same type.

Note For more information on working with the system software image files on the flash file system, see Chapter 26, "Working With the Flash File System."

Specifying the FTP Username and Password

FTP allows you to specify a username and password to be used for the FTP connection.

To specify the username and password, perform these steps:

Step 1 Enter the **set ftp username** *new_ftp_username* command.

Step 2 Enter the **set ftp password** command.

This example shows how to set the FTP username:

```
Console> (enable) set ftp username doc_people
ftp username set to doc_people
```

This example shows how to set the FTP password:

```
Console> (enable) set ftp password
Enter password for User 'doc_people':
Retype password for User 'doc_people':
ftp password set.
```

This example shows how to clear the FTP username:

```
Console> (enable) clear ftp username
```

This example shows how to clear the FTP password:

```
Console> (enable) clear ftp password
```

You can also connect to an FTP server using passive mode. In passive mode, the client initiates the connection to the server. To use passive mode, enter the **set ftp passive** command.

Preparing to Download an Image Using FTP or TFTP

Before you begin downloading a software image using FTP or TFTP, do the following:

- Verify that the workstation acting as the TFTP server is configured properly. When using TFTP on a Sun workstation, verify that the /etc/inetd.conf file contains this line:

  ```
  tftp dgram udp wait root /usr/etc/in.tftpd in.tftpd -p -s /tftpboot
  ```

  Verify that the /etc/services file contains this line:

  ```
  tftp 69/udp
  ```

  When using FTP on a Sun workstation, verify that the /etc/inetd.conf file contains this line:

  ```
  ftp stream tcp nowait root /usr/sbin/tcpd in.ftpd
  ```

  Verify that the /etc/services file contains this line:

  ```
  ftp 21/udp
  ```

  Note You must restart the inetd daemon after modifying the /etc/inetd.conf and /etc/services files. To restart the daemon, either stop the inetd process and restart it, or enter a **fastboot** command (on the SunOS 4.x) or a **reboot** command (on Solaris 2.x or SunOS 5.x). Refer to the documentation for your workstation for more information on using the FTP or TFTP daemon.

- Verify that the switch has a route to the FTP or TFTP server. The switch and the FTP or TFTP server must be in the same subnetwork if you do not have a router to route the traffic between the subnets. Check connectivity to the FTP or TFTP server by entering the **ping** command.
- Verify that the software image to be downloaded is in the correct directory on the FTP or TFTP server.
- Verify that the permissions on the file are set correctly. The permissions on the file should be set to world-read.
- Note that a power interruption (or other problem) during the download procedure can corrupt the flash code. If the flash code is corrupted, you can connect to the switch through the console port and boot from an uncorrupted system image on a Flash PC card.

Downloading the Supervisor Engine Images Using FTP or TFTP

Note If you have a redundant supervisor engine, you cannot download a system image directly from an FTP or TFTP server to the flash memory on the standby supervisor engine. When you download the image to the active supervisor engine, the standby supervisor engine synchronizes automatically with the new image. In addition, you cannot copy an image from the standby supervisor engine to the active supervisor engine.

To download a supervisor engine software image to the switch from an FTP or TFTP server, perform these steps:

Step 1 Copy the software image file to the appropriate FTP or TFTP directory on the workstation.

Step 2 Log into the switch through the console port or through a Telnet session. If you log in using Telnet, your Telnet session disconnects when you reset the switch to run the new software.

Step 3 Enter the **copy ftp flash** or **copy tftp flash** command. When prompted, enter the IP address or host name of the TFTP server and the name of the file to download. On those platforms that support the flash file system, you are also prompted for the flash device to which to copy the file and the destination filename.

The switch downloads the image file from the FTP or TFTP server to the specified flash device.

Note The switch remains operational while the image downloads.

Step 4 Modify the BOOT environment variable using the **set boot system flash** *device***:***filename* **prepend** command, so that the new image boots when you reset the switch. Specify the flash device (*device***:**) and the filename of the downloaded image (*filename*).

Step 5 Reset the switch by entering the **reset system** command. If you are connected to the switch through Telnet, your Telnet session disconnects.

During startup, the flash memory on the supervisor engine is reprogrammed with the new flash code.

Step 6 When the switch reboots, enter the **show version** command to check the version of the code on the switch.

Note For examples that show the complete FTP or TFTP download procedures for the various supervisor engine and switch types, see the "FTP and TFTP Download Procedures Example" section on page 27-9.

Downloading the Switching Module Images Using FTP or TFTP

To download a software image to an intelligent module, perform these steps:

Step 1 Copy the software image file to the appropriate FTP or TFTP directory on the workstation.

Step 2 Log into the switch through the console port or a Telnet session. If you log in using Telnet, your Telnet session might disconnect when you reset the modules to run the new software.

Step 3 If there is only one module of the type that is appropriate for the image, or if there are multiple modules of the same type and you want to update the image on all of them, enter the **copy ftp flash** or **copy tftp flash** command. When prompted, enter the IP address or the host name of the TFTP server, the name of the file to download, the flash device to which to copy the file, and the destination filename.

Step 4 If there are multiple modules of the type that is appropriate for the image but you only want to update a single module, enter the **copy ftp** *m*/**bootflash:** or **copy tftp** *m*/**bootflash:** command, where *m* is the number of the module to which to download the software image.

Note If you do not specify a module number, the switch examines the header of the image file to determine to which modules the software is downloaded. The image is then downloaded to all the modules of that type.

The switch downloads the image file, erases the flash memory on the appropriate modules, and reprograms the flash memory with the downloaded flash code.

Note All modules in the switch remain operational while the image downloads.

Step 5 Reset the appropriate modules by entering the **reset** *mod* command. If you are connected through Telnet, your Telnet session disconnects if you reset the module through which your connection was made.

Step 6 When the upgraded modules come online, enter the **show version** [*mod*] command to check the version of the code on the switch.

Note For examples that show the complete procedures on FTP and TFTP downloads to the intelligent modules, see the "Single Module Image Download Example" section on page 27-12 and the "Multiple Module Image Download Example" section on page 27-13.

FTP and TFTP Download Procedures Example

These sections show example TFP and TFTP download procedures:

Supervisor Engine Image Download Example

Note For a procedure on downloading a supervisor engine software image from an FTP or TFTP server, see the "Downloading the Supervisor Engine Images Using FTP or TFTP" section on page 27-7.

This example shows a complete TFTP download procedure of a supervisor engine software image to a Catalyst 6500 series switch:

```
Console> (enable) copy tftp flash
IP address or name of remote host []? 172.20.52.3
Name of file to copy from []? cat6000-sup.5-2-1-CSX.bin
Flash device [bootflash]?
Name of file to copy to [cat6000-sup.5-2-1-CSX.bin]?

4369664 bytes available on device bootflash, proceed (y/n) [n]? y
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCC
File has been copied successfully.

Console> (enable) set boot system flash bootflash:cat6000-sup.5-2-1-CSX.bin
BOOT variable = bootflash:cat6000-sup.5-2-1-CSX.bin,1;
Console> (enable) reset system
This command will reset the system.
Do you want to continue (y/n) [n]? y
Console> (enable) 07/21/1998,13:51:39:SYS-5:System reset from Console//

System Bootstrap, Version 4.2
Copyright (c) 1994-1998 by cisco Systems, Inc.
c6k_sup1 processor with 32768 Kbytes of main memory

Autoboot executing command: "boot bootflash:cat6000-sup.5-2-1-CSX.bin"
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCC
Uncompressing file:  ###########################################################
################################################################################
################################################################################
################################################################################
################################################################################
################################################################################
################################################################################
################################################################################
################################################################################
################################################################################
#############

System Power On Diagnostics
DRAM Size ...................32 MB
Testing DRAM.................Passed
Verifying Text segment .......Passed
NVRAM Size ..................512 KB
Saving NVRAM ................
Testing NVRAM ................Passed
Restoring NVRAM..............
Level2 Cache .................Present
Level2 Cache test.............Passed

Leaving power_on_diags

Cafe Daughter Present.

EOBC link up

Boot image: bootflash:cat6000-sup.5-2-1-CSX.bin
Flash Size = 0X1000000, num_flash_sectors = 64
readCafe2Version: 0x00000001
RIn Local Test Mode, Pinnacle Synch Retries: 2
Running System Diagnostics from this Supervisor (Module 1)
This may take up to 2 minutes....please wait
```

```
Cisco Systems Console



Enter password:
07/21/1998,13:52:51:SYS-5:Module 1 is online
07/21/1998,13:53:11:SYS-5:Module 4 is online
07/21/1998,13:53:11:SYS-5:Module 5 is online
07/21/1998,13:53:14:PAGP-5:Port 1/1 joined bridge port 1/1.
07/21/1998,13:53:14:PAGP-5:Port 1/2 joined bridge port 1/2.
07/21/1998,13:53:40:SYS-5:Module 2 is online
07/21/1998,13:53:45:SYS-5:Module 3 is online

Console>
```

This example shows a complete FTP download procedure of a supervisor engine software image to a Catalyst 6500 series switch:

```
Console> (enable) copy ftp flash
IP address or name of remote host []? 172.20.52.3
Name of file to copy from []? cat6000-sup2k8.7-7-1.bin
Flash device [bootflash]?
Name of file to copy to [cat6000-sup2k8.7-7-1.bin ]?

4369664 bytes available on device bootflash, proceed (y/n) [n]? y
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCC
File has been copied successfully.
Console> (enable) set boot system flash bootflash: cat6000-sup2k8.7-7-1.bin
BOOT variable = bootflash:cat6000-sup2k8.7-7-1.bin,1;
Console> (enable) reset system
This command will reset the system.
Do you want to continue (y/n) [n]? y
Console> (enable) 04/29/2003,13:51:39:SYS-5:System reset from Console//

System Bootstrap, Version 4.2
Copyright (c) 1994-1998 by cisco Systems, Inc.
c6k_sup1 processor with 32768 Kbytes of main memory

Autoboot executing command: "boot bootflash:cat6000-sup2k8.7-7-1.bin,1"
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCC
Uncompressing file:  ###########################################################
################################################################################
################################################################################
################################################################################
################################################################################
################################################################################
################################################################################
################################################################################
################################################################################
################################################################################
#############

System Power On Diagnostics
DRAM Size ....................32 MB
Testing DRAM..................Passed
Verifying Text segment .......Passed
NVRAM Size ...................512 KB
Saving NVRAM .................
Testing NVRAM ................Passed
Restoring NVRAM...............
Level2 Cache ..................Present
Level2 Cache test.............Passed
```

```
Leaving power_on_diags

Cafe Daughter Present.

EOBC link up

Boot image: bootflash:cat6000-sup2k8.7-7-1.bin,1
Flash Size = 0X1000000, num_flash_sectors = 64
readCafe2Version: 0x00000001
RIn Local Test Mode, Pinnacle Synch Retries: 2
Running System Diagnostics from this Supervisor (Module 1)
This may take up to 2 minutes....please wait

Cisco Systems Console



Enter password:
07/21/1998,13:52:51:SYS-5:Module 1 is online
07/21/1998,13:53:11:SYS-5:Module 4 is online
07/21/1998,13:53:11:SYS-5:Module 5 is online
07/21/1998,13:53:14:PAGP-5:Port 1/1 joined bridge port 1/1.
07/21/1998,13:53:14:PAGP-5:Port 1/2 joined bridge port 1/2.
07/21/1998,13:53:40:SYS-5:Module 2 is online
07/21/1998,13:53:45:SYS-5:Module 3 is online
Console>
```

Single Module Image Download Example

Note For a procedure on downloading the software images to the intelligent modules, see the "Downloading the Switching Module Images Using FTP or TFTP" section on page 27-8.

This example shows a complete TFTP download procedure of an ATM software image to a single ATM module:

```
Console> (enable) show version 4
Mod Port Model      Serial #  Versions
--- ---- ---------- --------- -----------------------------------------
4   1    WS-X6101   003414855 Hw : 1.2
                               Fw : 1.3
                               Sw : 3.2(6)

Console> (enable) copy tftp 4/flash
IP address or name of remote host []? 172.20.52.3
Name of file to copy from []? cat6000-atm.3-2-7.bin
Download image tftp:cat6000-atm.3-2-7.bin to Module 4 FLASH (y/n) [n]? y
This command will reset Download Module(s) you selected.

Do you wish to continue download flash (y/n) [n]? y
-
Download done for module 4, please wait for it to come online

File has been copied successfully.
```

```
Console> (enable) 07/21/1998,13:13:54:SYS-5:Module 4 is online

Console> (enable) show version 4
Mod Port Model      Serial #  Versions
--- ---- ---------- --------- -----------------------------------------
4   1    WS-X6101   003414855 Hw : 1.2
                               Fw : 1.3
                               Sw : 3.2(7)
Console> (enable)
```

This example shows a complete FTP download procedure of an ATM software image to a single ATM module:

```
Console> (enable) show version 4
Mod Port Model      Serial #  Versions
--- ---- ---------- --------- -----------------------------------------
4   1    WS-X6101   003414855 Hw : 1.2
                               Fw : 1.3
                               Sw : 3.2(6)

Console> (enable) copy ftp 4/flash
IP address or name of remote host []? 172.20.52.3
Name of file to copy from []? c6atm-lc-mz.121-14.E1.bin
Download image tftp:c6atm-lc-mz.121-14.E1.bin to Module 4 FLASH (y/n) [n]? y
This command will reset Download Module(s) you selected.

Do you wish to continue download flash (y/n) [n]? y
-
Download done for module 4, please wait for it to come online

File has been copied successfully.
Console> (enable) 04/29/2003,13:13:54:SYS-5:Module 4 is online

Console> (enable) show version 4
Mod Port Model      Serial #  Versions
--- ---- ---------- --------- -----------------------------------------
4   1    WS-X6101   003414855 Hw : 1.2
                               Fw : 1.3
                               Sw : 3.2(7)
Console> (enable)
```

Multiple Module Image Download Example

Note For a procedure on downloading the software images to the intelligent modules, see the "Downloading the Switching Module Images Using FTP or TFTP" section on page 27-8.

This example shows a complete TFTP download procedure of an ATM software image to multiple ATM modules:

```
Console> (enable) show version 4
Mod Port Model      Serial #  Versions
--- ---- ---------- --------- -----------------------------------------
4   1    WS-X6101   003414855 Hw : 1.2
                               Fw : 1.3
                               Sw : 3.2(6)
```

```
Console> (enable) show version 5
Mod Port Model      Serial #  Versions
--- ---- ---------- --------- -----------------------------------------
5   1    WS-X6101   003414463 Hw : 1.2
                               Fw : 1.3
                               Sw : 3.2(6)
Console> (enable) copy tftp flash
IP address or name of remote host []? 172.20.52.3
Name of file to copy from []? cat6000-atm.3-2-7.bin
Download image tftp:cat6000-atm.3-2-7.bin to Module 4 FLASH (y/n) [n]? y
Download image tftp:cat6000-atm.3-2-7.bin to Module 5 FLASH (y/n) [n]? y
This command will reset Download Module(s) you selected.

Do you wish to continue download flash (y/n) [n]? y
-
Download done for module 4, please wait for it to come online

Download done for module 5, please wait for it to come online

File has been copied successfully.
Console> (enable) 07/21/1998,12:25:10:SYS-5:Module 4 is online
07/21/1998,12:25:10:SYS-5:Module 5 is online

Console> (enable) show version 4
Mod Port Model      Serial #  Versions
--- ---- ---------- --------- -----------------------------------------
4   1    WS-X6101   003414855 Hw : 1.2
                               Fw : 1.3
                               Sw : 3.2(7)
Console> (enable) show version 5
Mod Port Model      Serial #  Versions
--- ---- ---------- --------- -----------------------------------------
5   1    WS-X6101   003414463 Hw : 1.2
                               Fw : 1.3
                               Sw : 3.2(7)
Console> (enable)
```

Uploading the System Software Images to an FTP or TFTP Server

These sections describe how to upload the system software images from a switch to an FTP or TFTP server:

- Preparing to Upload an Image to an FTP or TFTP Server, page 27-15
- Uploading the Software Images to an FTP or TFTP Server, page 27-15

Note For more information on working with the system software image files on the flash file system, see Chapter 26, "Working With the Flash File System."

Preparing to Upload an Image to an FTP or TFTP Server

Before you attempt to upload a software image to an FTP or TFTP server, do the following:

- Verify that the workstation acting as the FTP or TFTP server is configured properly. When using FTP on a Sun workstation, make sure that the /etc/inetd.conf file contains this line:

  ```
  ftp stream tcp nowait root /usr/sbin/tcpd in.ftpd
  ```

  Verify that the /etc/services file contains this line:

  ```
  ftp 21/udp
  ```

  When using TFTP on a Sun workstation, verify that the /etc/inetd.conf file contains this line:

  ```
  tftp dgram udp wait root /usr/etc/in.tftpd in.tftpd -p -s /tftpboot
  ```

  Verify that the /etc/services file contains this line:

  ```
  tftp 69/udp
  ```

  Note You must restart the inetd daemon after modifying the /etc/inetd.conf and /etc/services files. To restart the daemon, either stop the inetd process and restart it, or enter a **fastboot** command (on the SunOS 4.x) or a **reboot** command (on Solaris 2.x or SunOS 5.x). Refer to the documentation for your workstation for more information on using the TFTP daemon.

- Verify that the switch has a route to the FTP or TFTP server. The switch and the FTP or TFTP server must be in the same subnetwork if you do not have a router to route the traffic between the subnets. Check the connectivity to the FTP or TFTP server by entering the **ping** command.
- Note that you might need to create an empty file on the FTP or TFTP server before uploading the image. To create an empty file, enter the **touch** *filename* command, where *filename* is the name of the file that you will use when uploading the image to the server.
- If you are overwriting an existing file (including an empty file, if you had to create one), verify that the permissions on the file are set correctly. The permissions on the file should be world-write.

Uploading the Software Images to an FTP or TFTP Server

To upload a software image on a switch to an FTP or TFTP server for storage, perform these steps:

Step 1 Log into the switch through the console port or a Telnet session.

Step 2 Upload the software image to the FTP or TFTP server with the **copy flash ftp** or **copy flash tftp** command. When prompted, specify the FTP or TFTP server address and destination filename. On those platforms that support the flash file systems, you are first prompted for the flash device and the source filename. If desired, you can enter the **copy** *file-id* **ftp** or **copy** *file-id* **tftp** command on these platforms.

The software image is uploaded to the FTP or TFTP server.

This example shows how to upload the supervisor engine software image using TFTP:

```
Console> (enable) copy flash tftp
Flash device [bootflash]? slot0:
Name of file to copy from []? cat6000-sup.5-4-1.bin
IP address or name of remote host [172.20.52.3]? 172.20.52.10
Name of file to copy to [cat6000-sup.5-4-1.bin]?
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC|
File has been copied successfully.
Console> (enable)
```

Downloading the System Software Images Using rcp

These sections describe how to download the system software images to the switch supervisor engine and to the intelligent modules:

- Preparing to Download an Image Using rcp, page 27-16
- Downloading the Supervisor Engine Images Using rcp, page 27-16
- Downloading the Switching Module Images Using rcp, page 27-17
- Example rcp Download Procedures, page 27-18

Preparing to Download an Image Using rcp

Before you begin downloading a software image using rcp, do the following:

- Verify that the workstation acting as the rcp server supports the remote shell (rsh).
- Verify that the switch has a route to the rcp server. The switch and the rcp server must be in the same subnetwork if you do not have a router to route the traffic between the subnets. Check the connectivity to the rcp server by entering the **ping** command.
- If you are accessing the switch through the console or a Telnet session without a valid username, make sure that the current rcp username is the one that you want to use for the rcp download. You can enter the **show users** command to view the current valid username. If you do not want to use the current username, create a new rcp username by entering the **set rcp username** command. The new username will be stored in NVRAM. If you are accessing the switch through a Telnet session with a valid username, this username will be used and there is no need to set the rcp username.
- A power interruption (or other problem) during the download procedure can corrupt the flash code. If the flash code is corrupted, you can connect to the switch through the console port and boot from an uncorrupted system image on a Flash PC card.

Downloading the Supervisor Engine Images Using rcp

To download a supervisor engine software image to the switch from an rcp server, perform these steps:

Step 1 Copy the software image file to the appropriate rcp directory on the workstation.

Step 2 Log into the switch through the console port or through a Telnet session. If you log in using Telnet, your Telnet session disconnects when you reset the switch to run the new software.

Step 3 Download the software image from the rcp server by entering the **copy rcp flash** command. When prompted, enter the IP address or host name of the rcp server and the name of the file to download. On those platforms that support the flash file system, you are also prompted for the flash device to which to copy the file and the destination filename.

The switch downloads the image file from the rcp server.

Note The switch remains operational while the image downloads.

Step 4 Modify the BOOT environment variable by entering the **set boot system flash** *device***:***filename* **prepend** command, so that the new image boots when you reset the switch. Specify the flash device (*device***:**) and the filename of the downloaded image (*filename*).

Step 5 Reset the switch by entering the **reset system** command. If you are connected to the switch through Telnet, your Telnet session disconnects.

During startup, the flash memory on the supervisor engine is reprogrammed with the new flash code.

Step 6 When the switch reboots, enter the **show version** command to check the version of the code on the switch.

Downloading the Switching Module Images Using rcp

To download a software image to an intelligent module on a Catalyst 6500 series switch, perform these steps:

Step 1 Copy the software image file to the appropriate rcp directory on the workstation.

Step 2 Log into the switch through the console port or a Telnet session. If you log in using Telnet, your Telnet session might disconnect when you reset the modules to run the new software.

Step 3 Enter the command that is appropriate for your switch and supervisor engine to download the software image from the rcp server:

- If there is only one module of the type that is appropriate for the image, or if there are multiple modules of the same type and you want to update the image on all of them, enter the **copy rcp flash** command. When prompted, enter the IP address or host name of the rcp server, the name of the file to download, the flash device to which to copy the file, and the destination filename.
- If there are multiple modules of the type that is appropriate for the image but you only want to update a single module, enter the **copy rcp** | *m***/bootflash:** command, where *m* is the number of the module to which to download the software image. If you do not specify the module, all the modules of the same type will be updated.

Note If you do not specify a module number, the switch examines the header of the image file to determine to which modules the software is downloaded. The image is then downloaded to all the modules of that type.

The switch downloads the image file, erases the flash memory on the appropriate modules, and reprograms the flash memory with the downloaded flash code.

Note All the modules in the switch remain operational while the image downloads.

Step 4 Reset the appropriate modules using the **reset** *mod* command. If you are connected through Telnet, your Telnet session disconnects if you reset the module through which your connection was made.

Step 5 When the upgraded modules come online, enter the **show version** [*mod*] command to check the version of the code on the switch.

Example rcp Download Procedures

These sections show example rcp download procedures:

- Supervisor Engine Image rcp Download Example, page 27-18
- Single Module Image rcp Download Example, page 27-20
- Multiple Module Image rcp Download Example, page 27-20

Supervisor Engine Image rcp Download Example

Note For a procedure on downloading a supervisor engine software image from an rcp server, see the "Downloading the Supervisor Engine Images Using rcp" section on page 27-16.

This example shows a complete rcp download procedure of a supervisor engine software image to a Catalyst 6500 series switch:

```
Console> (enable) copy rcp flash
IP address or name of remote host []? 172.20.52.3
Name of file to copy from []? cat6000-sup.5-2-1-csx.bin
Flash device [bootflash]?
Name of file to copy to [cat6000-sup.5-2-1-csx.bin]?

4369664 bytes available on device bootflash, proceed (y/n) [n]? y
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCC
File has been copied successfully.
Console> (enable) set boot system flash bootflash:cat6000-sup.5-2-1-csx.bin prepend
BOOT variable = bootflash:cat6000-sup.5-2-1-csx.bin,1;bootflash:cat6000-sup.5-2-
1-csx.bin,1;
Console> (enable) reset system
This command will reset the system.
Do you want to continue (y/n) [n]? y
Console> (enable) 09/2/1999,13:51:39:SYS-5:System reset from Console//
```

```
System Bootstrap, Version 4.2
Copyright (c) 1994-1999 by cisco Systems, Inc.
Presto processor with 32768 Kbytes of main memory

Autoboot executing command: "boot bootflash:cat6000-sup.5-2-1-csx.bin"
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCC
Uncompressing file:  ###########################################################
################################################################################
################################################################################
################################################################################
################################################################################
################################################################################
################################################################################
################################################################################
################################################################################
################################################################################
#############


System Power On Diagnostics
DRAM Size ...................32 MB
Testing DRAM.................Passed
Verifying Text segment .......Passed
NVRAM Size ..................512 KB
Saving NVRAM ................
Testing NVRAM ...............Passed
Restoring NVRAM..............
Level2 Cache .................Present
Level2 Cache test.............Passed

Leaving power_on_diags

Cafe Daughter Present.

EOBC link up

Boot image: bootflash:cat6000-sup.5-2-1-CSX.bin
Flash Size = 0X1000000, num_flash_sectors = 64
readCafe2Version: 0x00000001
RIn Local Test Mode, Pinnacle Synch Retries: 2
Running System Diagnostics from this Supervisor (Module 1)
This may take up to 2 minutes....please wait

Cisco Systems Console

Enter password:
09/2/1999,13:52:51:SYS-5:Module 1 is online
09/2/1999,13:53:11:SYS-5:Module 4 is online
09/2/1999,13:53:11:SYS-5:Module 5 is online
09/2/1999,13:53:14:PAGP-5:Port 1/1 joined bridge port 1/1.
09/2/1999,13:53:14:PAGP-5:Port 1/2 joined bridge port 1/2.
09/2/1999,13:53:40:SYS-5:Module 2 is online
09/2/1999,13:53:45:SYS-5:Module 3 is online
Console> (enable)
```

Single Module Image rcp Download Example

Note For a procedure on downloading the software images to the intelligent modules, see the "Downloading the Switching Module Images Using rcp" section on page 27-17.

This example shows a complete rcp download procedure of an ATM software image to a single ATM module:

```
Console> (enable) show version 4
Mod Port Model      Serial #  Versions
--- ---- ---------- --------- -----------------------------------------
4   1    WS-X6101   003414855 Hw : 1.2
                               Fw : 1.3
                               Sw : 3.2(6)
Console> (enable) copy rcp 4/flash
IP address or name of remote host []? 172.20.52.3
Name of file to copy from []? cat6000-atm.3-2-7.bin
Download image rcp:cat6000-atm.3-2-7.bin to Module 4 FLASH (y/n) [n]? y
This command will reset Download Module(s) you selected.

Do you wish to continue download flash (y/n) [n]? y

Download done for module 4, please wait for it to come online

File has been copied successfully.
Console> (enable) 09/2/1999,13:13:54:SYS-5:Module 4 is online

Console> (enable) show version 4
Mod Port Model      Serial #  Versions
--- ---- ---------- --------- -----------------------------------------
4   1    WS-X6101   003414855 Hw : 1.2
                               Fw : 1.3
                               Sw : 3.2(7)
Console> (enable)
```

Multiple Module Image rcp Download Example

Note For a procedure on downloading the software images to the intelligent modules, see the "Downloading the Switching Module Images Using rcp" section on page 27-17.

This example shows a complete rcp download procedure of an ATM software image to multiple ATM modules:

```
Console> (enable) show version 4
Mod Port Model      Serial #  Versions
--- ---- ---------- --------- -----------------------------------------
4   1    WS-X6101   003414855 Hw : 1.2
                               Fw : 1.3
                               Sw : 3.2(6)
Console> (enable) show version 5
Mod Port Model      Serial #  Versions
--- ---- ---------- --------- -----------------------------------------
5   1    WS-X6101   003414463 Hw : 1.2
                               Fw : 1.3
                               Sw : 3.2(6)
Console> (enable) copy rcp flash
IP address or name of remote host []? 172.20.52.3
Name of file to copy from []? cat6000-atm.3-2-7.bin
Download image rcp:cat6000-atm.3-2-7.bin to Module 4 FLASH (y/n) [n]? y
Download image rcp:cat6000-atm.3-2-7.bin to Module 5 FLASH (y/n) [n]? y
This command will reset Download Module(s) you selected.

Do you wish to continue download flash (y/n) [n]? y
-
Download done for module 4, please wait for it to come online
Download done for module 5, please wait for it to come online

File has been copied successfully.
Console> (enable) 09/2/1999,12:25:10:SYS-5:Module 4 is online
09/2/1999,12:25:10:SYS-5:Module 5 is online

Console> (enable) show version 4
Mod Port Model      Serial #  Versions
--- ---- ---------- --------- -----------------------------------------
4   1    WS-X6101   003414855 Hw : 1.2
                               Fw : 1.3
                               Sw : 3.2(7)
Console> (enable) show version 5
Mod Port Model      Serial #  Versions
--- ---- ---------- --------- -----------------------------------------
5   1    WS-X6101   003414463 Hw : 1.2
                               Fw : 1.3
                               Sw : 3.2(7)
Console> (enable)
```

Uploading the System Software Images to an rcp Server

These sections describe how to upload the system software images from a switch to an rcp server:

- Preparing to Upload an Image to an rcp Server, page 27-22
- Uploading the Software Images to an rcp Server, page 27-22

Note For more information on working with the system software image files on the flash file system, see Chapter 26, "Working With the Flash File System."

Preparing to Upload an Image to an rcp Server

Before you attempt to upload a software image to an rcp server, do the following:

- Verify that the workstation acting as the rcp server is configured properly.
- Verify that the switch has a route to the rcp server. The switch and the rcp server must be in the same subnetwork if you do not have a router to route the traffic between the subnets. Check the connectivity to the rcp server by entering the **ping** command.
- If you are overwriting an existing file (including an empty file, if you had to create one), verify that the permissions on the file are set correctly. The permissions on the file should be set to write for the specific username.

Uploading the Software Images to an rcp Server

To upload a software image on a switch to an rcp server for storage, perform these steps:

Step 1 Log into the switch through the console port or a Telnet session.

Step 2 Upload the software image to the rcp server using the **copy flash rcp** command. When prompted, specify the rcp server address and destination filename. On those platforms that support the flash file systems, you are first prompted for the flash device and source filename. If desired, you can use the **copy** *file-id* **rcp** command on these platforms.

The software image is uploaded to the rcp server.

This example shows how to upload the supervisor engine software image to an rcp server:

```
Console> (enable) copy flash rcp
Flash device [bootflash]? slot0:
Name of file to copy from []? cat6000-sup.5-3-1.bin
IP address or name of remote host [172.20.52.3]? 172.20.52.10
Name of file to copy to [cat6000-sup.5-3-1.bin]?
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC|
File has been copied successfully.
Console> (enable)
```

Downloading the Crypto Images Using SCP

The Secure Copy (SCP) provides a secure and authenticated method for copying the crypto image files. SCP relies on Secure Shell (SSH) and requires that AAA authorization be configured so that the system can determine whether the user has the correct privilege level.

SCP allows a user who has appropriate authorization to copy a crypto file to and from the system by using the **copy** command. An authorized network administrator may also perform this action from a workstation.

Because SCP relies on SSH for its secure transport, the system must have an RSA key pair. You must configure and enable SSH and configure authentication and authorization correctly before you can enable SCP. For information on configuring AAA, see Chapter 39, "Configuring the Switch Access Using AAA."

These sections describe how to download the system software crypto images to the switch supervisor engine:

Preparing to Download an Image Using SCP

Before you begin downloading a software image using SCP, do the following:

- Verify that the workstation acting as the SCP server supports the secure shell (SSH).
- Verify that the server supports a command shell that has an SSH v1 or SSH v2-compatible **scp** command available.

Note With software release 8.6(1) and later releases, SCP supports SSH v2.

- Verify that the switch has a route to the SCP server. The switch and the SCP server must be in the same subnetwork if you do not have a router to route the traffic between the subnets. Check the connectivity to the SCP server using the **ping** command.
- A power interruption (or other problem) during the download procedure can corrupt the flash code. If the flash code is corrupted, you can connect to the switch through the console port and boot from an uncorrupted system image on a Flash PC card.

Downloading the Crypto Images Using SCP

To download a supervisor engine software image to the switch from an SCP server, perform these steps:

Step 1 Copy the software image file to the appropriate SCP directory on the workstation.

Step 2 Log into the switch through the console port or through an SSH session. If you log in using Telnet, your Telnet session disconnects when you reset the switch to run the new software.

Step 3 Download the software image from the SCP server by entering the **copy scp flash** command. When prompted, enter the IP address or host name of the SCP server and the name of the file to download. On those platforms that support the flash file system, you are also prompted for the flash device to which to copy the file and the destination filename.

The switch downloads the image file from the SCP server.

Note The switch remains operational while the image downloads.

Step 4 Modify the BOOT environment variable by entering the **set boot system flash** *device***:***filename* **prepend** command, so that the new image boots when you reset the switch. Specify the flash device (*device***:**) and the filename of the downloaded image (*filename*).

Step 5 Reset the switch by entering the **reset system** command. If you are connected to the switch through Telnet, your Telnet session disconnects.

During startup, the flash memory on the supervisor engine is reprogrammed with the new flash code.

Step 6 When the switch reboots, enter the **show version** command to check the version of the code on the switch.

Example SCP Download Procedure

This example shows a complete SCP download procedure of a crypto image to a Catalyst 6500 series switch:

```
Console> (enable) copy scp flash
IP address or name of remote host []? 172.20.52.3
Name of file to copy from []? cat6000-sup720cvk9.8-3-1.bin
Username for scp[bob]?
Password for User bob[]:
Flash device [bootflash]?
Name of file to copy to [cat6000-sup720cvk9.8-3-1.bin]?

4369664 bytes available on device bootflash, proceed (y/n) [n]? y
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCC
File has been copied successfully.
Console> (enable) set boot system flash bootflash:cat6000-sup720cvk9.8-3-1.bin prepend
BOOT variable =
bootflash:cat6000-sup720cvk9.8-3-1.bin,1;bootflash:cat6000-sup720cvk9.8-3-1.bin
1-csx.bin,1;
Console> (enable) reset system
This command will reset the system.
Do you want to continue (y/n) [n]? y
Console> (enable) 11/25/2003,13:51:39:SYS-5:System reset from Console//

System Bootstrap, Version 4.2
Copyright (c) 1994-2003 by cisco Systems, Inc.
Presto processor with 32768 Kbytes of main memory

Autoboot executing command: "boot bootflash:cat6000-sup720cvk9.8-3-1.bin"
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCC
Uncompressing file:  ###########################################################
################################################################################
################################################################################
################################################################################
################################################################################
################################################################################
################################################################################
################################################################################
################################################################################
################################################################################
#############


System Power On Diagnostics
DRAM Size ...................32 MB
Testing DRAM.................Passed
Verifying Text segment .......Passed
NVRAM Size ..................512 KB
Saving NVRAM ................
Testing NVRAM ................Passed
Restoring NVRAM..............
Level2 Cache .................Present
Level2 Cache test.............Passed
```

```
Leaving power_on_diags

Cafe Daughter Present.

EOBC link up

Boot image: cat6000-sup720cvk9.8-3-1.bin
Flash Size = 0X1000000, num_flash_sectors = 64
readCafe2Version: 0x00000001
RIn Local Test Mode, Pinnacle Synch Retries: 2
Running System Diagnostics from this Supervisor (Module 1)
This may take up to 2 minutes....please wait

Cisco Systems Console

Enter password:
11/25/2003,13:52:51:SYS-5:Module 1 is online
11/25/2003,13:53:11:SYS-5:Module 4 is online
11/25/2003,13:53:11:SYS-5:Module 5 is online
11/25/2003,13:53:14:PAGP-5:Port 1/1 joined bridge port 1/1.
11/25/2003,13:53:14:PAGP-5:Port 1/2 joined bridge port 1/2.
11/25/2003,13:53:40:SYS-5:Module 2 is online
11/25/2003,13:53:45:SYS-5:Module 3 is online
Console> (enable)
```

Uploading the Crypto Images to an SCP Server

These sections describe how to upload the system software images from a switch to an SCP server:

- Preparing to Upload an Image to an SCP Server, page 27-25
- Uploading the Crypto Images to an SCP Server, page 27-26

Note For more information on working with the system software image files on the flash file system, see Chapter 26, "Working With the Flash File System."

Preparing to Upload an Image to an SCP Server

Before you attempt to upload a software image to an SCP server, do the following:

- Verify that the workstation acting as the SCP server is configured properly.
- Verify that the switch has a route to the SCP server. The switch and the SCP server must be in the same subnetwork if you do not have a router to route the traffic between the subnets. Check the connectivity to the rcp server by entering the **ping** command.
- If you are overwriting an existing file (including an empty file, if you had to create one), verify that the permissions on the file are set correctly. The permissions on the file should be set to write for the specific username.

Uploading the Crypto Images to an SCP Server

To upload a crypto image on a switch to an SCP server for storage, perform these steps:

Step 1 Log into the switch through the console port or an SSH session.

Step 2 Upload the software image to the rcp server by entering the **copy flash scp** command. When prompted, specify the SCP server address and destination filename. On those platforms that support the flash file systems, you are first prompted for the flash device and source filename. If desired, you can enter the **copy** *file-id* **scp** command on these platforms.

The image is uploaded to the SCP server.

This example shows how to upload the crypto image to an SCP server:

```
Console> (enable) copy bootflash scp
Flash device [bootflash]? slot0:
Name of file to copy from []? cat6000-sup720cvk9.8-3-1.bin
Username for scp[bob]?
Password for User bob[]:
IP address or name of remote host [172.20.52.3]? 172.20.52.10
Name of file to copy to [cat6000-sup720cvk9.8-3-1.bin]?
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC|
File has been copied successfully.
Console> (enable) .
```

Downloading the Crypto Images Using SFTP

Note The Secure File Transfer Protocol (SFTP) is available only in crypto images.

FTP provides a file transfer capability, but with FTP, passwords and data files are transferred in plain text. SFTP provides a secure encrypted channel for passwords and data transmission across the network.

SFTP uses the SSH protocol for establishing a secure channel between the client and the server. SFTP is supported only with SSHv2. SFTP with SSHv1 is not supported.

SFTP client functionality is supported. SFTP server functionality is not supported.

To download a supervisor engine crypto software image to the switch from an SFTP server, perform these steps:

Step 1 Verify that the switch has a route to the SFTP server. The switch and the SFTP server must be in the same subnetwork if you do not have a router to route the traffic between the subnets. Check the connectivity to the SFTP server by entering the **ping** command.

Step 2 Copy the software image file to the appropriate SFTP directory on the workstation.

Step 3 Log into the switch through the console port or through a Telnet session. If you log in using Telnet, your Telnet session disconnects when you reset the switch to run the new software.

Step 4 Enter the **copy sftp** *destination* command. When prompted, enter the IP address or hostname of the SFTP server and the name of the file to download. You are also prompted for the flash device to which to copy the file and the destination filename. Enter your username and password. The switch downloads the image file from the SFTP server to the specified flash device.

Note The switch remains operational while the image downloads.

Step 5 Modify the BOOT environment variable by entering the **set boot system flash** *device***:***filename* **prepend** command, so that the new image boots when you reset the switch. Specify the flash device (*device***:**) and the filename of the downloaded image (*filename*).

Step 6 Reset the switch by entering the **reset system** command. If you are connected to the switch through Telnet, your Telnet session disconnects.

During startup, the flash memory on the supervisor engine is reprogrammed with the new flash code.

Step 7 When the switch reboots, enter the **show version** command to check the version of the code on the switch.

Uploading the Crypto Images to an SFTP Server

To upload a supervisor engine crypto software image from the switch to an SFTP server, perform these steps:

Step 1 Log into the switch through the console port or a Telnet session.

Step 2 Upload the software image to the SFTP server with the **copy** *source* **sftp** command. When prompted, specify the SFTP server address and destination filename. You are first prompted for the flash device and the source filename. Enter your username and password. The switch uploads the image file from the flash device on the switch to the SFTP server.

Note In the examples below, you can stop the copy process by entering Control+C at any time.

This example shows how to download a software image from an SFTP server to the switch:

```
Console> (enable) copy sftp <switch name>
IP address or name of remote host [10.6.1.10]?
Name of file to copy from [/tmp/bob/test2]?
Username for sftp[]? <username>
Password for User bob[]: <password>
37562980 bytes available on device bootflash, proceed (y/n) [n]? y

File has been copied successfully.
Console> (enable) copy sftp bootflash:
IP address or name of remote host []? <IP address>
Username for sftp[bob]?
Password for User bob[]:
Name of file to copy from []? <filename>

Can not open source file scp:/tmp/tin/test2 (SCP authentication error)
Copy from switch to SFTP Server
copy <source> sftp
```

Copying a file to an SFTP server is similar. You will be asked for the destination host and pathname and the copy process will occur without additional confirmation.

```
Console> (enable) copy bootflash:test2 sftp
IP address or name of remote host [10.6.1.10]?
Name of file to copy to [/tmp/bob/test2]?
Username for sftp[bob]?
Password for User bob[]:
CCC/
File has been copied successfully.
```

Downloading the Software Images Over a Serial Connection on the Console Port

These sections describe how to perform a serial download of the software images over the supervisor engine console port using Kermit, which is a popular file-transfer and terminal-emulation software program:

Preparing to Download an Image Using Kermit

Before you begin a serial download of a software image using Kermit, do the following:

- On a UNIX workstation, verify that your shell window is local (not an **rlogin** window to a different workstation).
- Verify that the supervisor engine console port is connected to a serial port on your PC or workstation with a serial cable.
- Verify that the Kermit software is installed on your PC or workstation.
- Verify that the line speed settings are the same on the PC or workstation and on the switch:
  - On the switch, you can change the console port speed by entering the **set system baud** *rate* command. The default baud rate is 9600 baud.
  - On the PC or workstation, you can change the baud rate of the serial port by entering the **set speed** *rate* command at the Kermit> prompt.

Caution To prevent communication problems, do not use a speed greater than 19,200 baud.

- Ensure that Kermit is using the proper serial port by doing the following:
  - On a PC, specify the serial port by entering the **set port com***x* command, where *x* is the PC serial port number (1 through 8) that you connected to the switch.
  - On a UNIX workstation, specify the serial port by entering the **set port /dev/tty***x* command, where *x* is the serial port (a or b) that you connected to the switch.

Downloading the Software Images Using Kermit (PC Procedure)

Note This procedure applies to the PC serial downloads only. For information on performing a serial download on a UNIX workstation, see the "Downloading the Software Images Using Kermit (UNIX Procedure)" section on page 27-30.

To perform a serial download of a software image over the supervisor engine console port, perform these steps:

Step 1 Copy the software image file to the directory where Kermit is loaded.

Step 2 Start Kermit on the PC.

Note Before continuing, ensure that the line speed is correct and that you have selected the proper serial line, as described in the "Preparing to Download an Image Using Kermit" section on page 27-28.

Step 3 At the Kermit> prompt, enter the **connect** command to connect to the switch. If your line and speed are set correctly, the switch Console> prompt appears.

Step 4 Enter the **enable** command to enter privileged mode.

Step 5 Enter the **download serial** command. The file is downloaded to module 1 by default.

Step 6 When prompted, confirm the download.

Step 7 Enter the escape sequence **Ctrl-]-c** by holding down the **Ctrl** key while you press **]**, and then press **c**.

Step 8 At the Kermit> prompt, enter the **send** *filename* command to send the file to the switch.

The switch downloads the image file, erases the flash memory on the supervisor engine or the appropriate module, and reprograms the flash memory with the downloaded flash code.

Note The switch remains operational while the image downloads.

Step 9 When the Kermit> prompt reappears, enter the **connect** command to return to the switch Console> prompt. You will see the status information as the switch erases and reprograms the flash memory.

Note If you enter the **connect** command more than 2 minutes after the Kermit> prompt reappears, you might see only a Console> prompt instead of the status information about erasing and programming flash code.

Step 10 Reset the switch using the **reset system** command.

Step 11 When the switch reboots, enter the **show version** [*mod*] command to check the version of the code on the switch.

Note For an example that shows a complete serial download procedure using Kermit on a PC, see the "PC Serial Download Procedure Example" section on page 27-31.

Downloading the Software Images Using Kermit (UNIX Procedure)

Note This procedure applies to the UNIX serial downloads only. For information on performing a serial download on a PC, see the "Downloading the Software Images Using Kermit (PC Procedure)" section on page 27-29.

Use this procedure to perform a serial download of a software image over the supervisor engine console port.

To copy the software to the workstation, log in as root, and perform these steps:

Step 1 Copy the software image file to your home directory.

Step 2 At the UNIX command prompt, start Kermit by entering the **kermit** command (make sure that the directory where Kermit is installed is included in the $PATH environment variable on the workstation).

Note Before continuing, ensure that the line speed is correct and that you have selected the proper serial line, as described in the "Preparing to Download an Image Using Kermit" section on page 27-28.

Step 3 At the C-Kermit> prompt, enter the **connect** command to connect to the switch. If your line and speed are set correctly, the switch Console> prompt appears.

Step 4 Enter the **enable** command to enter privileged mode.

Step 5 Enter the **download serial** command. The file downloads to module 1 by default.

Step 6 When prompted, confirm the download.

Step 7 Enter the escape sequence **Ctrl-\-c** by holding down the **Ctrl** key while you press **\,** and then press **c**.

Step 8 At the Kermit> prompt, enter the **send** *filename* command to send the file to the switch.

You can monitor the progress of the download by pressing the **a** key at any time during the Kermit download. A dot appears onscreen for every four packets that are transferred. If there is a problem transferring the file, one or more of the following letter codes appear:

- T—Kermit timed out.
- N—Kermit is not acknowledging the switch download process.
- E—Kermit detected an error in the progress of the transaction.

The switch downloads the image file, erases the flash memory on the supervisor engine or the appropriate module, and reprograms the flash memory with the downloaded flash code.

Note The switch remains operational while the image downloads.

Step 9 Press **Return** to return to the C-Kermit> prompt. When the Kermit> prompt reappears, enter the **connect** command to return to the switch Console> prompt. You will see the status information as the switch erases and reprograms the flash memory.

Note If you enter the **connect** command more than 2 minutes after the Kermit> prompt reappears, you might see only a Console> prompt instead of the status information about erasing and programming flash code.

Step 10 Reset the switch by entering the **reset system** command.

Step 11 When the switch reboots, enter the **show version** [*mod*] command to check the version of the code on the switch.

Note For an example that shows a complete serial download procedure using Kermit on a UNIX workstation, see the "UNIX Workstation Serial Download Procedure Example" section on page 27-32.

Example Serial Software Image Download Procedures

These sections show the example serial download procedures over the supervisor engine console port using Kermit:

PC Serial Download Procedure Example

This screen output shows an example of a complete serial download procedure on a PC:

```
C:\ copy A:\*.*
copying c6509_xx.bin
C:\ kermit
Kermit, 4C(057) 06 Apr 98, 4.2 BSD
Type ? for help
Kermit> set port com1
Kermit> set speed 9600
Kermit> connect
Connecting to com1,speed 9600.
The escape character is ^] (ASCII 29).
Type the escape character followed by C to get back,
or followed by ? to see other options

Console> enable
Console> (enable) download serial
Download CBI image via console port (y/n) [n]? y

Waiting for DOWNLOAD!
Return to your local Machine by typing its escape sequence
Issue Kermit send command from there[ Send `Filename`]

<CONTROL-] c to return to Local Machine>

Kermit> send c6509_xx.bin


          File name: c6509_xx.bin
 KBytes transferred: xxxx
```

```
Percent transferred: 100%
            Sending: Complete

  Number of Packets: xxxx
  Number of retries: None
         Last error: None
       Last warning: None
Kermit> connect

Finished network download.  (1136844 bytes)
Flash erase in progress ... Erase done
Programming Flash:  Flash Programming Complete
Flash erase in progress ... Erase done
Programming Flash:  Flash Programming Complete
Flash erase in progress ... Erase done
Programming Flash:  Flash Programming Complete
Flash erase in progress ... Erase done
Programming Flash:  Flash Programming Complete
Flash erase in progress ... Erase done
Programming Flash:  Flash Programming Complete
Flash erase in progress ... Erase done
Programming Flash:  Flash Programming Complete
Flash erase in progress ... Erase done
Programming Flash:  Flash Programming Complete
The system needs to be reset to run the new image.

Cisco Systems Console
Enter password:
Mon Apr 06, 1998, 14:35:08
Console>
```

UNIX Workstation Serial Download Procedure Example

This screen output shows an example of a complete serial download procedure on a UNIX workstation:

```
workstation% cd /tmp
workstation% tar -xvfp /dev/rfd0
c5009_xx.bin, 1156046 bytes, 2258 tape blocks
workstation% ls -la
total 1150
drwxrwsrwt  5 bin           512 Sep 28 04:15 .
drwxr-xr-x 18 root         1536 Sep 27 15:41 ..
-r--r--r--  1 60000     1156046 Jul 18 10:32 c5009_xx.bin
workstation% kermit
C-Kermit, 4E(072) 06 Apr 98, SUNOS 4.x
Type ? for help
C-Kermit> set line /dev/ttya
C-Kermit> set speed 9600
/dev/ttya: 9600 baud
C-Kermit> connect
Connecting thru /dev/ttya, speed 9600.
The escape character is CTRL-\ (28).

Type the escape character followed by C to get back,
or followed by ? to see other options.

Console> enable
Console> (enable) download serial c5009_xx.bin

Download CBI image via console port (y/n) [n]? y

Waiting for DOWNLOAD!
Return to your local Machine by typing its escape sequence
```

```
Issue Kermit send command from there[ Send `Filename`]
[Back at Local System]
C-Kermit> send c5009_xx.bin
SF
c5009_xx.bin => c5009_xx.bin, Size: 1156046

CTRL-F to cancel file,  CTRL-R to resend current packet
CTRL-B to cancel batch, CTRL-A for status report:
.....................................................................................
.....................................
*** Display Truncated ***
.............................................................
..................................... [OK]
ZB?
C-Kermit> connect
Connecting thru /dev/ttya, speed 9600.
The escape character is CTRL-\ (28).
Type the escape character followed by C to get back,
or followed by ? to see other options.

Download OK
Initializing Flash
Programming Flash
Base....Code....Length....Time....Done

Cisco Systems Console
Enter password:
Mon Apr 06, 1998, 17:35:08
Console>
```

Downloading a System Image Using Xmodem or Ymodem

When you need a system image on the switch, but the switch does not have network access and you do not have a software image on a Flash PC card, you can download an image from a local or remote computer (such as a PC, UNIX workstation, or Macintosh) through the console port using the Xmodem or Ymodem protocol.

The Xmodem and Ymodem protocols are used to transfer files and are included in applications such as Windows 3.1 (TERMINAL.EXE), Windows 95 (HyperTerminal), Windows NT 3.5x (TERMINAL.EXE), Windows NT 4.0 (HyperTerminal), and Linux UNIX freeware (minicom).

The Xmodem and Ymodem downloads are slow. Use them only when the switch does not have network access. You can speed up the transfer by setting the console port speed to 38400 bps.

The Xmodem and Ymodem file transfers are performed from the ROM monitor with this command:

xmodem [**-y**] [**-c**] [**-s** *data-rate*]

where **-y** uses the Ymodem protocol, **-c** provides CRC-16 checksumming, and **-s** sets the console port data rate.

The computer from which you transfer the supervisor engine software image must run terminal emulation software that supports the Xmodem or Ymodem protocol.

This procedure shows a file transfer using the Xmodem protocol. To use the Ymodem protocol, include the **-y** keyword with the **xmodem** command.

Caution A modem connection from the telephone network to your console port can introduce security issues that you should consider before enabling the connection. For example, the remote users can dial into your modem and access the switch configuration settings.

Caution If you have redundant supervisor engines, you must remove the second (redundant) supervisor engine before you perform this procedure. The image that is downloaded through Xmodem is not saved to memory; therefore, after the download if you have two supervisor engines that are installed and attempt to reboot the active supervisor engine with the downloaded image, the redundant supervisor engine will take over and synchronize with the active supervisor engine. The downloaded image will not be booted.

Step 1 Place a supervisor engine software image on the computer's hard drive. You can download an image from Cisco.com (see the "Preface"section for details).

Step 2 To download from a local computer, connect the console port (port mode switch in the *in* position) to a serial port on the computer using a null-modem cable. The console port speed must match the speed that is configured on the local computer.

Note If you are transferring from a local computer, you may need to configure the terminal emulation program to ignore the RTS/DTR signals.

Step 3 To download from a remote computer, do the following:

a. Connect a modem to the console port and to the telephone network.

b. Note that the modem and console port must communicate at the same speed, which can be from 1200 to 38400 bps, depending on the speed that is supported by your modem. Enter the **confreg** ROM monitor command to configure the console port transmission speed.

c. Connect a modem to the remote computer and to the telephone network and configure it for the same speed as the supervisor engine.

d. Dial the number of the supervisor engine modem from the remote computer.

Step 4 Enter the **xmodem** command at the ROM-monitor prompt in the terminal emulation window:

```
rommon > xmodem -s 38400 -c
```

Step 5 Start an Xmodem or Ymodem send operation with the computer's terminal emulation software. The computer downloads the system image to the supervisor engine. See your terminal emulation software application manual for instructions on how to execute a Xmodem or Ymodem file transfer.

After the new image is completely downloaded, the ROM monitor boots it.

Note Downloading an image through the console port does not create an image file on any of the flash devices. The downloaded image resides only in memory. You cannot save the image in memory as a file.

Step 6 After the download, the console port returns to 9600, which is the default baud rate. If the download took place at other than 9600 baud, you must change the remote computer's baud rate back to 9600 baud.

Step 7 Establish network connectivity to the switch to copy an image file from a TFTP server to one of the flash devices.

Verifying the Software Images

Note This feature is not supported on Supervisor Engine 1.

Because a software image goes through a sequence of transfers before it is copied into the memory of the switch, the integrity of the image is at risk each time that it is downloaded from Cisco.com. The image size and checksum are automatically checked when the image is copied, but these types of checks do not ensure that the downloaded image has not been corrupted. To ensure the integrity of any images that you download, use the **set image-verification** command. You can set image verification to work when booting, after the image has been copied, or before a system reset.

To enable the image verification, perform this task in privileged mode:

	Task	Command
Step 1	Enable the image verification.	**set image-verification** [**boot** | **copy** | **reset**] **enable**
Step 2	Verify the image verification setting.	**show image-verification**

This example shows how to enable the image verification upon a switch reset:

```
Console> (enable) set image-verification reset enable
Console> (enable)
```

This example shows how to verify the image verification settings:

```
Console> (enable) show image-verification
Image Verification Status:
Boot:  Disable
Copy:  Disable
Reset: Enable
Console> (enable)
```

CHAPTER 28

Working with Configuration Files

This chapter describes how to work with the switch configuration files on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

Note The flash device names (such as **slot0:**) differ depending on the type of supervisor engine. See the "Understanding How the Flash File System Works" section on page 26-1 for details.

This chapter consists of these sections:

Working with the Configuration Files on the Switch

These sections describe how to work with the configuration files on the switch:

Note For more information on working with the configuration files on the flash file system, see Chapter 26, "Working With the Flash File System."

Creating and Using Configuration File Guidelines

Creating configuration files can help you configure your switch. The configuration files can contain some or all the commands that are needed to configure one or more switches. For example, you might want to download the same configuration file to several switches that have the same hardware configuration so that they have identical module and port configurations.

This section describes the guidelines for creating a configuration file:

- We recommend that you connect through the console port when using the configuration files to configure the switch. If you configure the switch from a Telnet session, the IP addresses are not changed, and the ports and the modules are not disabled.
- If no passwords have been set on the switch, you must set them on each switch by entering the **set password** and **set enablepass** commands. Enter a blank line after the **set password** and **set enablepass** commands. The passwords are saved in the configuration file as clear text.

  If the passwords already exist, you cannot enter the **set password** and **set enablepass** commands because the password verification will fail. If you enter the passwords in the configuration file, the switch mistakenly attempts to execute the passwords as commands as it executes the file.
- Certain commands must be followed by a blank line in the configuration file. The blank line is necessary; without the blank line, these commands might disconnect your Telnet session. Before disconnecting a session, the switch prompts you for confirmation. The blank line acts as a carriage return, which indicates a negative response to the prompt and retains the Telnet session.

  Include a blank line after each occurrence of these commands in a configuration file:
  - **set interface sc0** *ip_addr netmask*
  - **set interface sc0 disable**
  - **set module disable** *mod*
  - **set port disable** *mod/port*

Creating a Configuration File

When creating a configuration file, you must list the commands in a logical way so that the system can respond appropriately. To create a configuration file, perform these steps:

Step 1 Download an existing configuration from a switch.

Step 2 Open the configuration file in a text editor, such as vi or emacs on UNIX or Notepad on a PC.

Step 3 Extract the portion of the configuration file with the desired commands and save it in a new file. Make sure that the file begins with the word **begin** on a line by itself and ends with the word **end** on a line by itself.

Step 4 Copy the configuration file to the appropriate TFTP directory on the workstation (/tftpboot on a UNIX workstation).

Step 5 Make sure that the permissions on the file are set to world-read.

This example shows an example configuration file. This file could be used to set the Domain Name System (DNS) configuration on multiple switches.

```
begin

!
#dns
set ip dns server 172.16.10.70 primary
set ip dns server 172.16.10.140
set ip dns enable
set ip dns domain corp.com
end
```

Downloading the Configuration Files to the Switch Using TFTP

You can configure the switch using the configuration files that you create or download from another switch. In addition, you can store the configuration files on the flash devices on the hardware that supports the flash file system, and you can configure the switch using a configuration that is stored on a flash device.

These sections describe how to configure the switch using the configuration files that are downloaded from a TFTP server or that are stored on a flash device:

- Preparing to Download a Configuration File Using TFTP, page 28-3
- Configuring the Switch Using a File on a TFTP Server, page 28-4
- Configuring the Switch Using a File on a Flash Device, page 28-4

Preparing to Download a Configuration File Using TFTP

Before you begin downloading a configuration file using TFTP, do the following:

- Ensure that the workstation acting as the TFTP server is configured properly. On a Sun workstation, make sure that the /etc/inetd.conf file contains this line:

  ```
  tftp dgram udp wait root /usr/etc/in.tftpd in.tftpd -p -s /tftpboot
  ```

  Make sure that the /etc/services file contains this line:

  ```
  tftp 69/udp
  ```

  Note You must restart the inetd daemon after modifying the /etc/inetd.conf and /etc/services files. To restart the daemon, either stop the inetd process and restart it, or enter a **fastboot** command (on the SunOS 4.x) or a **reboot** command (on Solaris 2.x or SunOS 5.x). Refer to the documentation for your workstation for more information on using the TFTP daemon.

- Ensure that the switch has a route to the TFTP server. The switch and the TFTP server must be in the same subnetwork if you do not have a router to route the traffic between the subnets. Check the connectivity to the TFTP server using the **ping** command.
- Ensure that the configuration file to be downloaded is in the correct directory on the TFTP server (/tftpboot on a UNIX workstation).
- Ensure that the permissions on the file are set correctly. The permissions on the file should be set to world-read.

Configuring the Switch Using a File on a TFTP Server

To configure the switch using a configuration file that is downloaded from a TFTP server, perform these steps:

Step 1 Copy the configuration file to the appropriate TFTP directory on the workstation.

Step 2 Log into the switch through the console port or a Telnet session.

Step 3 Configure the switch using the configuration file that is downloaded from the TFTP server with the **copy tftp config** command. Specify the IP address or host name of the TFTP server and the name of the file to download.

The configuration file downloads and the commands are executed as the file is parsed line by line.

This example shows how to configure the switch using a configuration file that is downloaded from a TFTP server:

```
Console> (enable) copy tftp config
IP address or name of remote host []? 172.20.52.3
Name of file to copy from []? dns-config.cfg

Configure using tftp:dns-config.cfg (y/n) [n]? y
/
Finished network download.  (134 bytes)
>>
>> set ip dns server 172.16.10.70 primary
172.16.10.70 added to DNS server table as primary server.
>> set ip dns server 172.16.10.140
172.16.10.140 added to DNS server table as backup server.
>> set ip dns enable
DNS is enabled
>> set ip dns domain corp.com
Default DNS domain name set to corp.com
Console> (enable)
```

Configuring the Switch Using a File on a Flash Device

To configure a switch using a configuration file that is stored on a flash device in the flash file system, perform these steps:

Step 1 Log into the switch through the console port or a Telnet session.

Step 2 Locate the configuration file using the **cd** and **dir** commands (for more information, see Chapter 26, "Working With the Flash File System").

Step 3 Configure the switch using the configuration file that is stored on the flash device with the **copy** *file-id* **config** command.

The commands are executed as the file is parsed line by line.

This example shows how to configure the switch using a configuration file that is stored on a flash device:

```
Console> (enable) copy slot0:dns-config.cfg config

Configure using slot0:dns-config.cfg (y/n) [n]? y

Finished network download.  (134 bytes)
>>
>> set ip dns server 172.16.10.70 primary
172.16.10.70 added to DNS server table as primary server.
>> set ip dns server 172.16.10.140
172.16.10.140 added to DNS server table as backup server.
>> set ip dns enable
DNS is enabled
>> set ip dns domain corp.com
Default DNS domain name set to corp.com
Console> (enable)
```

Uploading the Configuration Files to a TFTP Server

These sections describe how to upload the running configuration or a configuration file that is stored on a flash device to a TFTP server:

- Preparing to Upload a Configuration File to a TFTP Server, page 28-5
- Uploading a Configuration File to a TFTP Server, page 28-6

Preparing to Upload a Configuration File to a TFTP Server

Before you attempt to upload a configuration file to a TFTP server, do the following:

- Ensure that the workstation acting as the TFTP server is configured properly. On a Sun workstation, make sure that the /etc/inetd.conf file contains this line:

  ```
  tftp dgram udp wait root /usr/etc/in.tftpd in.tftpd -p -s /tftpboot
  ```

  Make sure that the /etc/services file contains this line:

  ```
  tftp 69/udp
  ```

  Note You must restart the inetd daemon after modifying the /etc/inetd.conf and /etc/services files. To restart the daemon, either stop the inetd process and restart it, or enter a **fastboot** command (on the SunOS 4.x) or a **reboot** command (on Solaris 2.x or SunOS 5.x). Refer to the documentation for your workstation for more information on using the TFTP daemon.

- Ensure that the switch has a route to the TFTP server. The switch and the TFTP server must be in the same subnetwork if you do not have a router to route the traffic between the subnets. Check the connectivity to the TFTP server by entering the **ping** command.
- You might need to create an empty file on the TFTP server before uploading the configuration file. To create an empty file, enter the **touch** *filename* command, where *filename* is the name of the file that you will use when uploading the configuration to the server.
- If you are overwriting an existing file (including an empty file, if you had to create one), ensure that the permissions on the file are set correctly. The permissions on the file should be set to world-write.

Uploading a Configuration File to a TFTP Server

To upload a configuration file from a switch to a TFTP server for storage, perform these steps:

Step 1 Log into the switch through the console port or a Telnet session.

Step 2 Upload the switch configuration to the TFTP server with the **copy config tftp** command. Specify the IP address or host name of the TFTP server and the destination filename.

The file is uploaded to the TFTP server.

This example shows how to upload the running configuration to a TFTP server for storage:

```
Console> (enable) copy config tftp
IP address or name of remote host []? 172.20.52.3
Name of file to copy to []? cat6000_config.cfg

Upload configuration to tftp:cat6000_config.cfg, (y/n) [n]? y
.....
.........
.......

.........
...........
..
/
Configuration has been copied successfully.
Console> (enable)
```

Copying the Configuration Files Using SCP or rcp

This section describes how to copy the files using SCP or rcp:

- rcp Overview, page 28-6
- SCP Overview, page 28-7

rcp Overview

Remote copy protocol (rcp) provides another method of downloading, uploading, and copying the configuration files between the remote hosts and the switch. Unlike TFTP, which uses User Datagram Protocol (UDP), a connectionless protocol, rcp uses Transmission Control Protocol (TCP), which is connection oriented.

To use rcp to copy the files, the server from or to which you will be copying the files must support rcp. The rcp **copy** commands rely on the rsh server (or daemon) on the remote system. To copy the files using rcp, you do not need to create a server for file distribution, as you do with TFTP. You need only to have access to a server that supports the remote shell (rsh). (Most UNIX systems support rsh.) Because you are copying a file from one place to another, you must have read permission on the source file and write permission on the destination file. If the destination file does not exist, rcp creates it for you.

SCP Overview

The Secure Copy (SCP) provides a secure method for copying the crypto image files. SCP relies on Secure Shell (SSH) and allows you to copy a crypto file to and from the system through an encrypted channel.

Downloading the Configuration Files from an rcp or SCP Server

These sections describe how to download a configuration file from an rcp or SCP server to the running configuration or to a flash device:

- Preparing to Download a Configuration File Using rcp or SCP, page 28-7
- Configuring the Switch Using a File on an rcp or SCP Server, page 28-7

Preparing to Download a Configuration File Using rcp or SCP

Before you begin downloading a configuration file using rcp or SCP, do the following:

- Ensure that the workstation acting as the rcp server supports the remote shell (rsh).
- Ensure that the workstation acting as the SCP server supports the Secure Shell (SSH).
- Ensure that the switch has a route to the rcp or SCP server. The switch and the server must be in the same subnetwork if you do not have a router to route the traffic between the subnets. Check the connectivity to the rcp server using the **ping** command.
- If you are accessing the switch through the console or a Telnet session without a valid username, make sure that the current rcp username is the one that you want to use for the rcp download. You can enter the **show users** command to view the current valid username. If you do not want to use the current username, create a new username by entering the **set rcp username** command. The new username will be stored in NVRAM. If you are accessing the switch through a Telnet session with a valid username, this username will be used and there is no need to set the rcp username.

Configuring the Switch Using a File on an rcp or SCP Server

To configure a Catalyst 6500 series switch using a configuration file that is downloaded from an rcp or SCP server, perform these steps:

Step 1 Copy the configuration file to the appropriate directory on the workstation.

Step 2 Log into the switch through the console port or a Telnet session. If you are using SCP, log in using an SSH session.

Step 3 Configure the switch using the configuration file that is downloaded from the server by entering the **copy rcp | scp config** command. Specify the IP address or host name of the server and the name of the file to download.

The configuration file downloads and the commands are executed as the file is parsed line by line.

This example shows how to configure a Catalyst 6500 series switch using a configuration file that is downloaded from a server:

```
Console> (enable) copy rcp config
IP address or name of remote host []? 172.20.52.3
Name of file to copy from []? dns-config.cfg

Configure using rcp:dns-config.cfg (y/n) [n]? y
/
Finished network download.  (134 bytes)
>>
>> set ip dns server 172.16.10.70 primary
172.16.10.70 added to DNS server table as primary server.
>> set ip dns server 172.16.10.140
172.16.10.140 added to DNS server table as backup server.
>> set ip dns enable
DNS is enabled
>> set ip dns domain corp.com
Default DNS domain name set to corp.com
Console> (enable)
```

Uploading Configuration Files to an rcp or SCP Server

These sections describe how to upload the running configuration or a configuration file that is stored on a flash device to an rcp or SCP server:

- Preparing to Upload a Configuration File to an rcp or SCP Server, page 28-8
- Uploading a Configuration File to an rcp or SCP Server, page 28-8

Preparing to Upload a Configuration File to an rcp or SCP Server

Before you attempt to upload a configuration file to an rcp or SCP server, do the following:

- Ensure that the workstation acting as the rcp or SCP server is configured properly.
- Ensure that the switch has a route to the rcp or SCP server. The system and the server must be in the same subnetwork if you do not have a router to route the traffic between the subnets. Check the connectivity to the server by entering the **ping** command.
- If you are overwriting an existing file (including an empty file, if you had to create one), ensure that the permissions on the file are set correctly. The permissions on the file should be set to user-write.

Uploading a Configuration File to an rcp or SCP Server

To upload a configuration file from a switch to an rcp or SCP server for storage, perform these steps:

Step 1 Log into the switch through the console port or a Telnet session. If you are using SCP, log into the switch using an SSH session.

Step 2 Upload the switch configuration to the rcp server by entering the **copy config rcp | scp** command. Specify the IP address or host name of the rcp server and the destination filename.

The file is uploaded to the server.

This example shows how to upload the running configuration on a Catalyst 6500 series switch to an rcp server for storage:

```
Console> (enable) copy config rcp
IP address or name of remote host []? 172.20.52.3
Name of file to copy to []? cat6000_config.cfg

Upload configuration to rcp:cat6000_config.cfg, (y/n) [n]? y
.....
..........
.......

..........
...........
..
/
Configuration has been copied successfully.
Console> (enable)
```

This example shows how to upload the running configuration on a Catalyst 6500 series switch to an SCP server for storage:

```
Console> (enable) copy scp flash scp
IP address or name of remote host []? 172.20.52.3
Name of file to copy from []? cat6000-sup720cvk9.8-3-1.bin
Username for scp[bob]?
Password for User bob[]:
CCC/
File has been copied successfully.
```

Clearing the Configuration

To clear the configuration on the entire switch, perform this task in privileged mode:

Task	Command
Clear the switch configuration.	**clear config all**

This example shows how to clear the configuration for the entire switch:

```
Console> (enable) clear config all
This command will clear all configuration in NVRAM.
This command will cause ifIndex to be reassigned on the next system startup.
Do you want to continue (y/n) [n]? y
........
.............................

System configuration cleared.
Console> (enable)
```

To clear the configuration on an individual module, perform this task in privileged mode:

Task	Command
Clear the configuration on an individual module.	**clear config** *mod*

Note If you remove a module and replace it with a module of another type (for example, if you remove a 10/100 Ethernet module and insert a Gigabit Ethernet module), the module configuration is inconsistent. The output of the **show module** command indicates this problem. To resolve the inconsistency, clear the configuration on the problem module.

This example shows how to clear the configuration on an individual module:

```
Console> (enable) clear config 2
This command will clear module 2 configuration.
Do you want to continue (y/n) [n]? y
.............................
Module 2 configuration cleared.
Console> (enable)
```

To clear the configuration on an individual module port, perform this task in privileged mode:

Task	Command
Clear the configuration on an individual module port.	**clear config** {*mod* | *mod/port*}

Comparing the Configuration Files

You can compare the configuration files that are stored on the system to determine the differences between the configuration files or to check if changes have been made to the system configuration. To compare the configuration files, perform this task in privileged mode:

Task	Command
Compare the differences between the configuration files.	**show config differences** {**all** *file* | **context** *val* | *file* | **ignorecase**}

This example shows how to compare the differences between two different configuration files:

```
Console> (enable) show config differences 1.cfg 2.cfg
--- bootflash:1.cfg
+++ bootflash:2.cfg
@@ -8,1 +8,1 @@
-#version 8.2(0.11-Eng)DEL
+#VERSION 8.2(0.11-eNG)del
@@ -11,1 +11,1 @@
-set config mode text auto-save interval 1
+SET CONFIG MODE TEXT AUTO-SAVE INTERVAL 1
Console> (enable)
```

This example shows how to compare the differences between the configuration files but to ignore the differences in uppercase or lowercase characters:

```
Console> (enable) show config differences ignorecase 1.cfg 2.cfg
Files bootflash:1.cfg and bootflash:2.cfg are identical
Console> (enable)
```

Creating the Configuration Checkpoint Files for Configuration Rollback

You can roll back the current switch configuration file to a previously saved configuration file (also known as a "checkpoint" file) if the current file produces undesirable system results. This rollback feature provides a command to set multiple configuration "checkpoint" files. If you no longer want the current configuration file to run on the switch, you can return to one of these configuration checkpoint files quickly and with the least possible disturbance to switch functionality.

Follow these guidelines when creating the configuration checkpoint files:

- A configuration checkpoint file is identified by a name that you specify when you create the file. The configuration checkpoint filename can be no more than 15 characters. If you do not specify a name, the system generates one. The system-generated name is in the format CKPi_MMDDYYHHMM, where "i" represents a checkpoint number.
- The checkpoint file is stored either on the bootflash or on slotX/diskX. If you do not specify a device, the file is stored on the current default device.
- The configuration checkpoint file is stored as a text file that can be read and edited. We strongly advise that you do not edit the file.
- When a checkpoint filename is cleared from the system, the associated configuration checkpoint file is deleted. You should squeeze the device to reclaim space.
- You can create a maximum of five configuration checkpoint files on a system. You can roll back to any of the saved configuration checkpoint files in any order. Because these files are generated using a complete configuration, they are independent of each other.
- The checkpoint configuration is stored in NVRAM. The configuration is not cleared when you enter the **clear config all** command.
- This feature is supported on the systems with redundant supervisor engines. The checkpoint configuration and its associated files are synchronized to the standby supervisor engine.

To create a configuration checkpoint file, perform this task in privileged mode:

Task	Command
Create a configuration checkpoint file.	**set config checkpoint** [**name** *name*] [**device** *device*]
Verify the configuration checkpoint filename.	**show config checkpoints**

This example shows how to create a configuration checkpoint file and verify that it has been created:

```
Console> (enable) set config checkpoint
Configuration checkpoint CKP0_0722040905 creation successful.
Console> (enable) show config checkpoints
Checkpoint          File id                                 Date
==========          =======                                 ====
CKP0_0722040905     bootflash:CKP0_07220409058.4(0.79)COC   Thu Jul 22 2000, 09:05:31
Console> (enable)
```

To roll the current configuration file back to a previously created configuration checkpoint file, perform this task in privileged mode:

Task	Command
Roll the current configuration file back to a configuration checkpoint file.	**set config rollback** *name*

To clear all the configuration checkpoint files or a particular configuration checkpoint file, perform this task in privileged mode:

Task	Command
Clear all configuration checkpoint files or a particular configuration checkpoint file.	**clear config checkpoint** {**all** | *name*}
Verify the configuration checkpoint filenames.	**show config checkpoints**

This example shows how to clear all configuration checkpoint files and to verify that they have been cleared:

```
Console> (enable) clear config checkpoint all
All configuration checkpoints cleared.
Console> (enable) show config checkpoints
No Checkpoints defined.
Console> (enable)
```

Working with the Configuration Files on the MSFC

These sections describe how to work with the configuration files on the Multilayer Switch Feature Card (MSFC):

- Uploading the Configuration File to a TFTP Server, page 28-13
- Uploading the Configuration File to the Supervisor Engine Flash PC Card, page 28-14
- Downloading the Configuration File from a Remote Host, page 28-14
- Downloading the Configuration File from the Supervisor Engine Flash PC Card, page 28-16

The configuration information resides in two places when the MSFC is operating: the default (permanent) configuration in NVRAM and the running (temporary) memory in RAM. The default configuration always remains available; NVRAM retains the information even when the power is shut down. The current information is lost if the system power is shut down. The current configuration contains all the nondefault configuration information that you added by entering the **configure** command or the **setup** command facility, or by editing the configuration file.

The **copy running-config startup-config** command adds the current configuration to the default configuration in NVRAM, so that it is saved if power is shut down. Whenever you make changes to the system configuration, enter the **copy running-config startup-config** command to save the new configuration.

If you replace the MSFC, you need to replace the entire configuration. If you upload (copy) the configuration file to a remote server before removing the MSFC, you can retrieve it later and write it into NVRAM on the new MSFC. If you do not upload the configuration file, you need to enter the **configure** command to reenter the configuration information after you install the new MSFC.

Saving and retrieving the configuration file is not necessary if you are temporarily removing an MSFC that you are going to reinstall; the lithium batteries retain the configuration in memory. This procedure requires the privileged-level access to the EXEC command interpreter, which usually requires a password.

Uploading the Configuration File to a TFTP Server

Before you upload the running configuration to the TFTP file server, ensure the following:

- You have a connection to the MSFC either with a console terminal or remotely through a Telnet session.
- The MSFC is connected to a network supporting a file server (remote host).
- The remote host supports the TFTP application.
- You have the IP address or name of the remote host available.

To store information on a remote host, enter the privileged **write network** EXEC command. This command prompts you for the destination host address and a filename and then displays the instructions for confirmation. When you confirm the instructions, the MSFC sends a copy of the currently running configuration to the remote host. The system default is to store the configuration in a file called by the name of the MSFC with *-confg* appended. You can either accept the default filename by pressing **Return** at the prompt, or enter a different name before pressing **Return**.

To upload the currently running configuration to a remote host, perform these steps:

Step 1 Check if the system prompt displays a pound sign (#) to indicate the privileged level of the EXEC command interpreter.

Step 2 Enter the **ping** command to check the connection between the MSFC and the remote host.

Step 3 Enter the **write term** command to display the currently running configuration on the terminal and ensure that the configuration information is complete and correct. If it is not correct, enter the **configure** command to add or modify the existing configuration.

Step 4 Enter the **write net** command. The EXEC command interpreter prompts you for the name or IP address of the remote host that is to receive the configuration file. (The prompt might include the name or address of a default file server.)

```
Router# write net
Remote host []?
```

Step 5 Enter the name or IP address of the remote host. In this example, the name of the remote server is *servername*:

```
Router# write net
Remote host []? servername
Translating "servername"...domain server (1.1.1.1) [OK]
```

Step 6 Note that the EXEC command interpreter prompts you to specify a name for the file that is to hold the configuration. By default, the system appends *-confg* to the MSFC name to create the new filename. Press **Return** to accept the default filename, or enter a different name for the file before pressing **Return**. This example shows that the default is accepted:

```
Name of configuration file to write [Router-confg]?
Write file Router-confg on host 1.1.1.1? [confirm]
Writing Router-confg .....
```

Step 7 Note that before the MSFC executes the copy process, it displays the instructions that you entered for confirmation. If the instructions are not correct, enter **n** (no) and press **Return** to abort the process. To accept the instructions, press **Return** or **y** (yes) and then press **Return**, and the system begins the copy process. This example shows that the default is accepted:

```
Write file Router-confg on host 1.1.1.1? [confirm]
Writing Router-confg: !!!! [ok]
```

While the MSFC copies the configuration to the remote host, it displays a series of exclamation points (! ! !) or periods (. . .). The ! ! ! and [ok] indicate that the operation is successful. A display of . . . [timed out] or [failed] indicates a failure, which would probably be due to a network fault or the lack of a writable, readable file on the remote file server.

Step 8 Note that if the display indicates that the process was successful (with the series of ! ! ! and [ok]), the upload process is complete. The configuration is safely stored in the temporary file on the remote file server.

If the display indicates that the process failed (with the series of . . . as shown in the following example):

```
Writing Router-confg .....
```

your configuration was not saved. Repeat the preceding steps, or select a different remote file server and repeat the preceding steps.

If you are unable to copy the configuration to a remote host successfully, contact your network administrator.

Uploading the Configuration File to the Supervisor Engine Flash PC Card

To upload the configuration file to the supervisor engine Flash PC card, perform this task:

	Task	Command
Step 1	At the EXEC prompt, enter enable mode.	Router> **enable**
Step 2	Copy the startup configuration file to slot 0.	Router# **copy startup-config sup-slot0:***file_name*
Step 3	Copy the running configuration file to slot 0.	Router# **copy running-config sup-slot0:***file_name*

Downloading the Configuration File from a Remote Host

After you install the new MSFC, you can retrieve the saved configuration and copy it to NVRAM. Enter configuration mode and specify that you want to configure the MSFC from the network. The system prompts you for a host name and address, the name of the configuration file that is stored on the host, and confirmation to reboot using the remote file.

To download the currently running configuration from a remote host, perform these steps:

Step 1 Check if the system prompt displays a pound sign (#) to indicate the privileged level of the EXEC command interpreter.

Note Until you retrieve the previous configuration, the MSFC runs from the default configuration in NVRAM. Any passwords that were configured on the previous system are not valid until you retrieve the configuration.

Step 2 Enter the **ping** command to verify the connection between the router and the remote host.

Step 3 At the system prompt, enter the **configure network** command and press **Return** to enter configuration mode. Specify that you want to configure the system from a network device (instead of from the console terminal, which is the default).

```
Router# configure network
```

Step 4 Note that the system prompts you to select a host or network configuration file. The default is host; press **Return** to accept the default.

```
Host or network configuration file [host]?
```

Step 5 Note that the system prompts you for the IP address of the host. Enter the IP address or name of the remote host (the remote file server to which you uploaded the configuration file).

```
IP address of remote host [255.255.255.255]? 1.1.1.1
```

Step 6 Note that the system prompts you for the configuration filename. When uploading the file, the default is to use the name of the MSFC with the suffix *-confg* (*router-confg* in the following example). If you specified a different filename when you uploaded the configuration, enter the filename; otherwise, press **Return** to accept the default.

```
Name of configuration file [router-confg]?
```

Step 7 Note that before the system reboots with the new configuration, it displays the instructions that you entered for confirmation. If the instructions are not correct, enter **n** (no), and then press **Return** to cancel the process. To accept the instructions, press **Return**, or **y,** and then press **Return**.

```
Configure using router-confg from 1.1.1.1? [confirm]
Booting router-confg from 1.1.1.1: ! ! [OK - 874/16000 bytes]
```

While the MSFC retrieves and boots from the configuration on the remote host, the console display indicates whether or not the operation was successful. A series of !!!! and [OK] (as shown in the preceding example) indicate that the operation was successful. A series of . . . and [timed out] or [failed] indicate a failure (which would probably be due to a network fault or an incorrect server name, address, or filename). This example shows a failed attempt to boot from a remote server:

```
Booting Router-confg ..... [timed out]
```

Step 8 Proceed to the next step if the display indicates that the process was successful.

If the display indicates that the process failed, verify the name or address of the remote server and the filename, and repeat the preceding steps. If you are unable to retrieve the configuration, contact your network administrator.

Step 9 Enter the **write term** command to display the currently running configuration on the terminal. Review the display and ensure that the configuration information is complete and correct. If it is not, verify the filename and repeat the preceding steps to retrieve the correct file, or enter the **configure** command to add or modify the existing configuration. (See the appropriate software documentation for the configuration options that are available for the system, the individual interfaces, and specific configuration instructions.)

Step 10 When you verify that the currently running configuration is correct, enter the **copy running-config startup-config** command to save the retrieved configuration in NVRAM. Otherwise, you will lose the new configuration if you restart the system.

Downloading the Configuration File from the Supervisor Engine Flash PC Card

To download the configuration file from the supervisor engine Flash PC card, perform this task:

	Task	Command
Step 1	At the EXEC prompt, enter enable mode.	Router> **enable**
Step 2	Copy the stored running configuration file to the MSFC running configuration.	Router# **copy sup-slot0:** *file_name* **running-config**
Step 3	Copy the stored startup configuration file to the MSFC running configuration.	Router# **copy sup-slot0:***file_name* **startup-config**

Working with Profile Files

A profile file allows you to have a customized configuration as the default configuration on the switch. The profile file allows you to load a custom default configuration that enables or disables certain features at bootup or when a new module is installed. With the profile files, you can eliminate the features or processes that may pose security risks (for example, disabling CDP or turning off auto-trunking on a port) to your switch.

A profile file that has most of the security risks disabled is also known as a “lockdown” profile. A lockdown profile changes the functionality of the switch from enabling access to preventing access by default. When a lockdown profile is applied, you must manually enable the features that were disabled by the profile file.

Building Profile Files

The profile file format is similar to the format of a configuration file. You can either create a new profile file or edit a system-generated configuration file.

Caution We recommend that you do not create new profile files unless you are familiar with configuration files because missing or misplaced elements will cause the configuration to fail.

If you choose to create the profile files by editing a system-generated configuration file, most of the required notations will already be in the file. The keywords that are currently supported are ALL_MODULES, ALL_PORTS, ALL_MODULE_PORTS, and ALL_VLANS. Do not create a profile file using the output that results from entering the **copy config all** command as a template because the output includes default configuration information, which increases the size and processing time of the file.

To designate the system profile file that you want to use, perform this task in privileged mode:

	Task	Command
Step 1	Designate the device and name of the profile filename that you want to use.	**set system profile** *device*:*filename*
Step 2	Enable or disable the system profile files on the specified modules.	**set system profile** {**enable** | **disable**} *mod_list*

This example shows how to designate the device name and the profile filename:

```
Console> (enable) set system profile bootflash:test.cfg
System is set to be configured with profile file bootflash:test.cfg.
Console> (enable)
```

This example shows how to disable the system profile loading on a specified module:

```
Console> (enable) set system profile disable 2
System profile loading is disabled for module 2.
Console> (enable)
```

This example shows a sample lockdown profile file. You can use an exact copy of this file if you want to use what would be considered a typical lockdown profile file as your default configuration. You can also change the file and use the altered version of the file if the parameters of this lockdown profile file does not meet your needs.

```
begin
!
# ***** DEFAULT PROFILE *****
!
!
###################################
# Lockdown Profile version 1.0.3 #
###################################
!
# set system prompt (edit as needed)
set prompt locked_down>
!
# system attributes to be customized (edit as needed)
set system name locked_down
set system contact locked_down
set system location locked_down
!
# set a strong banner (edit as needed)
set banner motd ^
Access to this device or the attached networks is prohibited
without express permission from the network administrator.

Violators will be prosecuted to the fullest extent of both civil
and criminal law.
```

```
^
!
!
#
# vtp mode off, enable password and dummy domain (edit as needed)
set vtp domain locked_down
set vtp mode off
set vtp passwd locked_down
!
# default VLAN is "Quarantine" (edit as needed)
set vlan 999 name Quarantine
!
# Management VLAN is "Management" (edit as needed)
set vlan 1000 name Management
# Alternate management vlan is "OtherMgmt" (edit as needed)
set vlan 1001 name OtherMgmt
!
# sc0 and sc1 off (edit as needed)
set interface sc0 down
set interface sc0 1000
set interface sc1 down
set interface sc1 1001
!
# default port status is disabled
set port disable ALL_PORTS
!
# default cdp status is disabled
set cdp disable ALL_PORTS
!
# default STP status is with BPDU guard enabled
set spantree portfast bpdu-guard ALL_PORTS enable
!
# default PAgP/LACP status is disabled
set port channel ALL_PORTS mode off
!
# Default DTP status is disabled, no allowed vlans and dot1q-all-tagged mode on.
# Warning: A max of 128 trunks can have non-default configuration in CatOS 8.4
# Warning: Edit port list as needed.
set trunk ALL_PORTS off none
set dot1q-all-tagged enable
!
# default is CPU rate limiters enabled
set rate-limit l2pdu enable
!
# default SSH version is 2
set ssh mode v2
!
# default VLAN is "Quarantine" (edit as needed)
set vlan 999 ALL_PORTS
!
# Enable image checksum verification by default
set image-verification enable
!
# Set a more aggressive default logout timeout
set logout 10
#
#
# Anti-spoofing ACL
#
!
! Deny any packets from the RFC 1918, IANA reserved, ranges,
! multicast as a source, and loopback netblocks to block
! attacks from commonly spoofed IP addresses.
```

```
Note that the following ACLs might not be up to date.

! Refer to www.iana.org/assignments/ipv4-address-space for a current list.
!
! Bogons
!
set security acl ip Anti-spoofing deny ip 0.0.0.0 0.255.255.255  any log
set security acl ip Anti-spoofing deny ip 1.0.0.0 0.255.255.255  any log
set security acl ip Anti-spoofing deny ip 2.0.0.0 0.255.255.255  any log
set security acl ip Anti-spoofing deny ip 5.0.0.0 0.255.255.255  any log
set security acl ip Anti-spoofing deny ip 7.0.0.0 0.255.255.255  any log
set security acl ip Anti-spoofing deny ip 10.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 23.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 27.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 31.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 36.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 37.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 39.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 42.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 49.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 50.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 77.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 78.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 79.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 92.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 93.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 94.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 95.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 96.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 97.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 98.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 99.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 100.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 101.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 102.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 103.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 104.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 105.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 106.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 107.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 108.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 109.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 110.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 111.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 112.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 113.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 114.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 115.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 116.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 117.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 118.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 119.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 120.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 121.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 122.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 123.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 127.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 169.254.0.0 0.0.255.255 any log
set security acl ip Anti-spoofing deny ip 172.16.0.0 0.15.255.255 any log
set security acl ip Anti-spoofing deny ip 173.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 174.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 175.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 176.0.0.0 0.255.255.255 any log
```

```
set security acl ip Anti-spoofing deny ip 177.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 178.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 179.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 180.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 181.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 182.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 183.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 184.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 185.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 186.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 187.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 192.0.2.0 0.0.0.255     any log
set security acl ip Anti-spoofing deny ip 192.168.0.0 0.0.255.255 any log
set security acl ip Anti-spoofing deny ip 197.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 223.0.0.0 0.255.255.255 any log
set security acl ip Anti-spoofing deny ip 224.0.0.0 31.255.255.255 any log
# Add here a specific list of permits as needed
set security acl ip Anti-spoofing deny any any log
!
# Set protection to VLAN list (edit as needed)
# You can use ALL_VLANS but that will
# take some time to finish.
# Use the "show security acl" cmd to verify when
# the ACL mapping process is completed.
commit security acl all
set security acl map Anti-spoofing ALL_VLANS
!
!
end
```

CHAPTER 29

Configuring System Message Logging

This chapter describes how to configure the system message logging on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

Note For more information on the system messages, refer to the *Catalyst 6500 Series Switch System Message Guide*.

This chapter consists of these sections:

Understanding How the System Message Logging Works

The system message logging software can save messages in a log file or direct the messages to other devices. The system message logging facility has these features:

- Provides you with logging information for monitoring and troubleshooting
- Allows you to select the types of logging information that is captured
- Allows you to select the destination of the captured logging information

By default, the switch logs normal but significant system messages to its internal buffer and sends these messages to the system console. You can specify which system messages should be saved based on the type of facility (see Table 29-1) and the severity level (see Table 29-2). The messages are time-stamped to enhance real-time debugging and management.

You can access the logged system messages using the switch command-line interface (CLI) or by saving them to a properly configured syslog server. The switch software saves the syslog messages in an internal buffer that can store up to 500 messages. You can monitor the system messages remotely by accessing the switch through Telnet or the console port, or by viewing the logs on a syslog server.

If a system failure occurs, the system syslog-dump allows you to write the system messages in the syslog buffer to a flash file, capturing the pertinent syslog information before the system fails. If the system core dump is enabled, the syslog is dumped before the core.

Note The messages that are redirected to a syslog server are delayed up to 90 seconds.

Table 29-1 describes the facility types that are supported by the system message logs.

Table 29-1 System Message Log Facility Types

Facility Name	Definition
all	All facilities
acl	ACL facility
cdp	Cisco Discovery Protocol
cops	Common Open Policy Server
dtp	Dynamic Trunking Protocol
dvlan	Dynamic VLAN
earl	Enhanced Address Recognition Logic
filesys	File System
gvrp	GARP VLAN Registration Protocol
ip	Internet Protocol
kernel	Kernel
ld	ASLB facility
mcast	Multicast
mgmt	Management
mls	Multilayer Switching
pagp	Port Aggregation Protocol
protfilt	Protocol Filter
pruning	VTP pruning
privatevlan	Private VLAN facility
qos	Quality of Service
radius	Remote Access Dial-In User Service
rsvp	ReSerVation Protocol
security	Security
snmp	Simple Network Management Protocol
spantree	Spanning Tree Protocol
sys	System
tac	Terminal Access Controller
tcp	Transmission Control Protocol

Table 29-1 System Message Log Facility Types (continued)

Facility Name	Definition
telnet	Terminal Emulation Protocol
tftp	Trivial File Transfer Protocol
udld	User Datagram Protocol
vmps	VLAN Membership Policy Server
vtp	VLAN Trunking Protocol

Table 29-2 describes the severity levels that are supported by the system message logs.

Table 29-2 Severity Level Definitions

Severity Level	Description
0—emergencies	System unusable
1—alerts	Immediate action required
2—critical	Critical condition
3—errors	Error conditions
4—warnings	Warning conditions
5—notifications	Normal bug significant condition
6—informational	Informational messages
7—debugging	Debugging messages

System Log Message Format

The system log messages begin with a percent sign (%) and can contain up to 80 characters. The messages are displayed in this format:

mm/dd/yyy:hh/mm/ss:facility-severity-MNEMONIC:description

Table 29-3 describes the elements of the syslog messages.

Table 29-3 System Log Message Elements

Element	Description
mm/dd/yyy:hh/mm/ss	Date and time of the error or event. This information appears only if configured using the **set logging timestamp enable** command.
facility	Indicates the facility to which the message refers (for example, SNMP, SYS, etc.).
severity	Single-digit code from 0 to 7 that indicates the severity of the message.
MNEMONIC	Text string that uniquely describes the error message.
description	Text string containing the detailed information about the event being reported.

This example shows some typical switch system messages (at system startup):

```
1999 Apr 16 10:01:26 %MLS-5-MLSENABLED:IP Multilayer switching is enabled
1999 Apr 16 10:01:26 %MLS-5-NDEDISABLED:Netflow Data Export disabled
1999 Apr 16 10:01:26 %SYS-5-MOD_OK:Module 1 is online
1999 Apr 16 10:01:47 %SYS-5-MOD_OK:Module 3 is online
1999 Apr 16 10:01:42 %SYS-5-MOD_OK:Module 6 is online
1999 Apr 16 10:02:27 %PAGP-5-PORTTOSTP:Port 3/1 joined bridge port 3/1
1999 Apr 16 10:02:28 %PAGP-5-PORTTOSTP:Port 3/2 joined bridge port 3/2
```

Default System Message Logging Configuration

Table 29-4 describes the default system message logging configuration.

Table 29-4 Default System Message Logging Configuration

Configuration Parameter	Default Setting
System message logging to the console	Enabled
System message logging to Telnet sessions	Enabled
Logging buffer size	500 (default and maximum setting)
Logging history size	1
Logging history severity	Warnings (4)
Timestamp option	Enabled
Logging server	Disabled
Syslog server IP address	None configured
Server facility	LOCAL7
Server severity	Warnings (4)
Facility/severity level for system messages	**sys/5** **dtp/5** **pagp/5** **mgmt/5** **mls/5** **cdp/4** **udld/4** *all other facilities*/**2**
System syslog dump	Disabled
System syslog-dump device and filename specifications	flash device is slot0: Filename is sysloginfo

Configuring the System Message Logging on the Switch

These sections describe how to configure the system message logging on the switch:

Enabling and Disabling the Session Logging Settings

By default, the system logging messages are sent to the console and Telnet sessions that are based on the default logging facility and severity values. If desired, you can disable logging to the console or logging to a given Telnet session.

When you disable or enable logging to the console sessions, the enable state is applied to all future console sessions. For example, if you disable logging to the console, disconnect from the console port, and later reconnect, logging is still disabled for the console.

When you disable or enable logging to a Telnet session, the enable state is applied only to that session. If you disable logging to a Telnet session, disconnect the session, and later reconnect, logging is enabled for the new session.

Note If you enter the **set logging session** command while connected through the console port, the command has the same effect as entering the **set logging console** command. However, if you enter the **set logging console** command while you are connected through a Telnet session, the default console logging enable state is changed.

To enable or disable the logging state for the console sessions, perform this task in privileged mode:

	Task	Command
Step 1	Enable or disable the default logging state for the console sessions.	**set logging console** {**enable** | **disable**}
Step 2	Verify the logging configuration.	**show logging** [**noalias**]

This example shows how to disable logging to the current and future console sessions:

```
Console> (enable) set logging console disable
System logging messages will not be sent to the console.
Console> (enable)
```

To enable or disable the logging state for the current Telnet session, perform this task in privileged mode:

	Task	Command
Step 1	Enable or disable the logging state for a Telnet session.	**set logging session** {**enable** | **disable**}
Step 2	Verify the logging configuration.	**show logging** [**noalias**]

This example shows how to disable logging to the current Telnet session:

```
Console> (enable) set logging session disable
System logging messages will not be sent to the current login session.
Console> (enable)
```

Setting the System Message Logging Levels

You can set the severity level for each logging facility using the **set logging level** command. Enter the **all** keyword to specify all facilities. Enter the **default** keyword to make the specified severity level the default for the specified facilities. If you do not enter the **default** keyword, the specified severity level applies only to the current session.

To set the system message logging severity level setting for a logging facility, perform this task in privileged mode:

	Task	Command
Step 1	Set the severity level for the logging facilities.	**set logging level** {**all** | *facility*} *severity* [**default**]
Step 2	Verify the system message logging configuration.	**show logging** [**noalias**]

This example shows how to set the logging severity level to 5 for all the facilities (for the current session only):

```
Console> (enable) set logging level all 5
All system logging facilities for this session set to severity 5(notifications)
Console> (enable)
```

This example shows how to set the default logging severity level to 3 for the **cdp** facility:

```
Console> (enable) set logging level cdp 3 default
System logging facility <cdp> set to severity 3(errors)
Console> (enable)
```

Enabling and Disabling the Logging Time-Stamp Enable State

To enable or disable the logging time-stamp state, perform this task in privileged mode:

	Task	Command
Step 1	Enable or disable the logging time-stamp state.	**set logging timestamp** {**enable** | **disable**}
Step 2	Verify the logging time-stamp state.	**show logging** [**noalias**]

This example shows how to enable the time-stamp display on the system logging messages:

```
Console> (enable) set logging timestamp enable
System logging messages timestamp will be enabled.
Console> (enable)
```

Setting the Logging Buffer Size

To set the number of messages to log to the logging buffer, perform this task in privileged mode:

	Task	Command
Step 1	Set the number of messages to log to the logging buffer.	**set logging buffer** *buffer_size*
Step 2	Verify the system message logging configuration.	**show logging** [**noalias**]

This example shows how to set the logging buffer size to 200 messages:

```
Console> (enable) set logging buffer 200
System logging buffer size set to <200>
Console> (enable)
```

Limiting the Number of syslog Messages

You can limit the number of syslog messages that are sent to the history table and the SNMP network management station based on the severity. The default severity is set to warnings(4).

To limit the number of syslog messages, perform this task in privileged mode:

	Task	Command
Step 1	Limit the number of syslog messages.	**set logging history severity** *severity_level*
Step 2	Verify the system message logging configuration.	**show logging**

This example shows how to limit the number of syslog messages to the messages with a severity level of notifications(5):

```
Console> (enable) set logging history severity 5
System logging history set to severity <5>
Console> (enable)
```

Configuring the syslog Daemon on a UNIX syslog Server

Before you can send the system log messages to a UNIX syslog server, you must configure the syslog daemon on a UNIX server. Log in as root, and perform these steps:

Step 1 Add a line such as the following to the file /etc/syslog.conf:

```
user.debug                      /var/log/myfile.log
```

Note There must be five tab characters between **user.debug** and **/var/log/**myfile.log. Refer to the entries in the /etc/syslog.conf file for further examples.

The switch sends the messages according to the specified facility types and severity levels. The **user** keyword specifies the UNIX logging facility that is used. The messages from the switch are generated by the user processes. The **debug** keyword specifies the severity level of the condition being logged. You can set the UNIX systems to receive all the messages from the switch.

Step 2 Create the log file by entering these commands at the UNIX shell prompt:

```
$ touch /var/log/myfile.log
$ chmod 666 /var/log/myfile.log
```

Step 3 Make sure that the syslog daemon reads the new changes by entering this command:

```
$ kill -HUP `cat /etc/syslog.pid
```

Configuring the syslog Servers

Note Before you can send the system log messages to a UNIX syslog server, you must configure the syslog daemon on the UNIX server as described in the "Configuring the syslog Daemon on a UNIX syslog Server" section on page 29-8.

To configure the switch to log messages to a syslog server, perform this task in privileged mode:

	Task	Command
Step 1	Specify the IP address of one or more syslog servers[1].	**set logging server** *ip_addr*
Step 2	Set the facility and severity levels for syslog server messages.	**set logging server facility** *server_facility_parameter* **set logging server severity** *server_severity_level*
Step 3	Enable the system message logging to the configured syslog servers.	**set logging server enable**
Step 4	Verify the configuration.	**show logging** [**noalias**]

1. You can configure a maximum of three syslog servers.

This example shows how to specify a syslog server, set the facility and severity levels, and enable logging to the server:

```
Console> (enable) set logging server 10.10.10.100
10.10.10.100 added to System logging server table.
Console> (enable) set logging server facility local5
System logging server facility set to <local5>
Console> (enable) set logging server severity 5
System logging server severity set to <5>
Console> (enable) set logging server enable
System logging messages will be sent to the configured syslog servers.
Console> (enable)
```

To delete a syslog server from the syslog server table, perform this task in privileged mode:

Task	Command
Delete a syslog server from the syslog server table.	**clear logging server** *ip_addr*

This example shows how to delete a syslog server from the syslog server table:

```
Console> (enable) clear logging server 10.10.10.100
System logging server 10.10.10.100 removed from system logging server table.
Console> (enable)
```

To disable logging to the syslog server, perform this task in privileged mode:

Task	Command
Disable system message logging to the configured syslog servers.	**set logging server disable**

This example shows how to disable logging to the syslog servers:

```
Console> (enable) set logging server disable
System logging messages will not be sent to the configured syslog servers.
Console> (enable)
```

Displaying the Logging Configuration

Enter the **show logging** command to display the current system message logging configuration. Enter the **noalias** keyword to display the IP addresses instead of the host names of the configured syslog servers.

To display the current system message logging configuration, perform this task:

Task	Command
Display the current system message logging configuration.	**show logging** [**noalias**]

This example shows how to display the current system message logging configuration:

```
Console> (enable) show logging
Logging buffered size:       500
        timestamp option:    enabled
Logging history size:        1
            severity:        notifications(5)
Logging console:             enabled
Logging server:              disabled
        server facility:     LOCAL7
        server severity:     warnings(4
Current Logging Session:     enabled


Facility            Default Severity         Current Session Sever
-------------       ------------------------ ---------------------
acl                 5                        5
cdp                 4                        4
cops                3                        3
dtp                 5                        5
dvlan               2                        2
earl                2                        2
filesys             2                        2
gvrp                2                        2
ip                  2                        2
kernel              2                        2
ld                  3                        3
mcast               2                        2
mgmt                5                        5
mls                 5                        5
pagp                5                        5
protfilt            2                        2
pruning             2                        2
privatevlan         3                        3
qos                 3                        3
radius              2                        2
rsvp                3                        3
security            2                        2
snmp                2                        2
spantree            2                        2
sys                 5                        5
tac                 2                        2
tcp                 2                        2
telnet              2                        2
tftp                2                        2
udld                4                        4
vmps                2                        2
vtp                 2                        2

0(emergencies)        1(alerts)             2(critical)
3(errors)             4(warnings)           5(notifications)
6(information)        7(debugging)
Console> (enable)
```

Displaying the System Messages

Enter the **show logging buffer** command to display the messages in the switch logging buffer. If you do not specify *number_of_messages*, the default is to display the last 20 messages in the buffer (-20).

To display the messages in the switch logging buffer, perform one of these tasks:

Task	Command
Display the first *number_of_messages* messages in the buffer.	**show logging buffer** [*number_of_messages*]
Display the last *number_of_messages* messages in the buffer.	**show logging buffer -**[*number_of_messages*]

This example shows how to display the first five messages in the buffer:

```
Console> (enable) show logging buffer 5
1999 Apr 16 08:40:11 %SYS-5-MOD_OK:Module 1 is online
1999 Apr 16 08:40:14 %SYS-5-MOD_OK:Module 3 is online
1999 Apr 16 08:40:14 %SYS-5-MOD_OK:Module 2 is online
1999 Apr 16 08:41:15 %PAGP-5-PORTTOSTP:Port 2/1 joined bridge port 2/1
1999 Apr 16 08:41:15 %PAGP-5-PORTTOSTP:Port 2/2 joined bridge port 2/2
```

This example shows how to display the last five messages in the buffer:

```
Console> (enable) show logging buffer -5
%PAGP-5-PORTFROMSTP:Port 3/1 left bridge port 3/1
%SPANTREE-5-PORTDEL_SUCCESS:3/2 deleted from vlan 1 (PAgP_Group_Rx)
%PAGP-5-PORTFROMSTP:Port 3/2 left bridge port 3/2
%PAGP-5-PORTTOSTP:Port 3/1 joined bridge port 3/1-2
%PAGP-5-PORTTOSTP:Port 3/2 joined bridge port 3/1-2
Console> (enable)
```

Enabling and Disabling the System syslog Dump

If the system fails, a file containing the system messages in the syslog buffer (as displayed when entering the **show logging buffer** command) is produced.

To enable or disable the system syslog dump, perform this task in privileged mode (by default, the syslog dump is disabled):

	Task	Command
Step 1	Enable or disable the system syslog dump.	**set system syslog-dump** {**enable** | **disable**}
Step 2	Verify the status of the system syslog dump.	**show system**

This example shows how to enable the system syslog dump:

```
Console> (enable) set system syslog-dump enable
(1) In the event of a system crash, this feature will
cause a syslog file to be written out.
(2) Selected syslog file is slot0:sysloginfo
(3) Please make sure the above device has been installed,
and ready to use.
Syslog-dump enabled
Console> (enable)
```

This example shows how to disable the system syslog dump:

```
Console> (enable) set system syslog-dump disable
Syslog-dump disabled
Console> (enable)
```

This example shows how to display the status of the system syslog dump:

```
Console> (enable) show system
PS1-Status PS2-Status
---------- ----------
ok         none

Fan-Status Temp-Alarm Sys-Status Uptime d,h:m:s Logout
---------- ---------- ---------- -------------- ---------
ok         off        ok         1,00:03:18     20 min
.
.
.
Core Dump                Core File
------------------------ ----------------------
disabled                 slot0:crashinfo


Syslog Dump              Syslog File
------------------------ ----------------------
enabled                  slot0:sysloginfo
Console> (enable)
```

Specifying the System syslog Dump Flash Device and Filename

You can change the flash device and the filename when the syslog dump is enabled or disabled. If you only specify the flash device, the filename is automatically set to sysloginfo. If you do not specify the flash device or the filename, the previous filename for the syslog dump is cleared and the default flash device and filename (slot0:sysloginfo) are used.

To specify the flash device and filename for the system syslog dump, perform this task in privileged mode:

	Task	Command
Step 1	Specify the flash device and filename.	**set system syslog-file** [*device*:[*filename*]]
Step 2	Verify the flash device and filename settings.	**show system**

This example shows how to set the flash device for the syslog dump:

```
Console> (enable) set system syslog-file bootflash:
Default filename sysloginfo added to the device bootflash:
System syslog-file set.
Console> (enable)
```

This example shows how to set the flash device and the filename:

```
Console> (enable) set system syslog-file bootflash:sysmsgs1
System syslog-file set.
Console> (enable)
```

This example shows how to restore the flash device and the filename to the default settings:

```
Console> (enable) set system syslog-file
System syslog-file set to the default file.
Console> (enable)
```

Configuring CallHome

You can use the CallHome feature to set your switch to e-mail or page a syslog message of a specified severity to a specified e-mail or pager address or a set of e-mail or pager addresses.

CallHome is triggered whenever a syslog message is generated. If the severity of the generated syslog message is lower than the severity that you configure, the message is not forwarded to the destination addresses that you specified. If the severity is higher than the severity that you specified, the switch forwards the syslog message to the list of destination addresses that you entered.

CallHome is tied to the syslog messages and their severity. When you set the CallHome severity level, carefully consider the level of severity that you require for the existing **set logging level** command and the newly introduced **set logging callhome severity** command.

If you configure a very fine syslog severity level, such as for alerts (level 1), and a coarse CallHome severity level, such as for notifications (level 5), the destination addresses will receive the alerts and the emergencies only (levels 0 and 1). The destination addresses do not receive the remaining CallHome severity level notifications (levels 2, 3, and 4) that you specified. To ensure that the destination addresses receive the severity level alerts and notifications for all the levels that you want, set the CallHome severity level at the same severity level, or higher, than the level that you use to set the syslog message severity.

You can configure multiple SMTP servers so that the CallHome functionality is not disrupted if one server fails. If an SMTP server fails, the switch contacts the next configured server. If you configure multiple SMTP servers, the switch uses the first available SMTP server.

To configure CallHome on your switch, perform this task in privileged mode:

	Task	Command
Step 1	Enable CallHome.	**set logging callhome** {**enable** | **disable**}
Step 2	Specify the destination e-mail or pager address where you want to receive the syslog messages and the fragment size, if necessary.	**set logging callhome destination** *Email or Epage Address* [**fragment** *size in bytes*]
Step 3	Specify the SMTP server IP address(es) to which the switch should dispatch the syslog messages.	**set logging callhome smtp-server** *IP Address*
Step 4	Specify the CallHome severity level. **Note** By default, the severity level is set to critical messages only. (Level 2)	**set logging callhome severity** *level*

	Task	Command
Step 5	(Optional) Set the "from" e-mail address in case the SMTP server cannot forward the syslog message. **Note** The SMTP server will send a message to the "from" address for the failed deliveries.	**set logging callhome from** *Email Address*
Step 6	(Optional) Set the "reply to" e-mail address if you want the recipients to respond to a different address than the "from" address.	**set logging callhome reply-to** *Email address*
Step 7	Verify the configuration.	**show logging callhome**

This example shows how to enable CallHome:

```
Console> (enable) set logging callhome enable
Callhome functionality is enabled.
Callhome messages will be sent to the configured destination addresses.
Console> (enable)
```

This example shows how to set the following addresses to receive the CallHome messages:

- page adminjoe@epage.cisco.com using a fragment size of 128 bytes
- email adminboss@cisco.com, and adminjane@cisco.com

```
Console> (enable) set logging callhome destination adminjoe@epage.cisco fragment 128
Included adminjoe@epage.cisco in the table of callhome destination addresses.
Messages will be sent to this address in fragments of 128 bytes.
Console> (enable) set logging callhome destination adminjane@cisco.com
Included adminjane@cisco.com in the table of callhome destination addresses.
Messages will be sent to this address without fragmentation.
Console> (enable) set logging callhome destination adminboss@cisco.com
Included adminboss@cisco.com in the table of callhome destination addresses.
Messages will be sent to this address without fragmentation.
Console> (enable)
```

This example shows how to set the SMTP server with the IP address 172.16.8.19:

```
Console> (enable) set logging callhome smtp-server 172.20.8.16
Included 172.20.8.16 in the table of callhome SMTP servers.
Console> (enable)
```

This example shows how to set the severity to level 3 (critical and error messages):

```
Console> (enable) set logging callhome severity 3
Callhome severity level set to 3
Console> (enable)
```

This example shows how to set the From address to adminjoe@cisco.com:

```
Console> (enable) set logging callhome from adminjoe@cisco.com
From address of callhome messages is set to adminjoe@cisco.com
Console> (enable)
```

This example shows how to set the Reply to address to adminjane@cisco.com:

```
Console> (enable) set logging callhome reply-to adminjane@cisco.com
Reply-To address of callhome messages is set to adminjane@cisco.com
Console> (enable)
```

This example shows how to verify the configuration:

```
Console> (enable) show logging callhome
Callhome Functionality:       enabled
Callhome Severity:            LOG_ERR (3)

SMTP Server
-----------
172.20.8.16

Destination Address                                                Message Size
-------------------                                                ------------
adminboss@cisco.com                                                No Fragmentation
adminjane@cisco.com                                                No Fragmentation
adminjoe@epage.cisco                                               128 bytes

From: adminjoe@cisco.com
Reply-To: adminjane@cisco.com
Console> (enable)
```

Disabling CallHome

When you disable CallHome, you do not clear any other of the CallHome parameters that are set. You need to clear each parameter individually.

To disable CallHome on your switch, perform this task in privileged mode:

Task	Command
Disable CallHome.	**set logging callhome disable**

This example shows how to disable CallHome:

```
Console> (enable) set logging callhome disable
Callhome functionality is disabled.
Callhome messages will not be sent to the configured destination addresses.
Console> (enable)
```

To clear an address from the list of addresses that receive CallHome messages, perform this task in privileged mode:

Task	Command
Clear a destination address from the list of addresses that receive CallHome messages.	**clear logging callhome destination** *Email or Epage Address*

This example shows how to clear the destination address adminboss@cisco.com from the list of addresses that receive CallHome messages:

```
Console> (enable) clear logging callhome destination adminboss@cisco.com
Removed adminboss@cisco.com from the table of callhome destination addresses.
Console> (enable)
```

To clear the “from” address, perform this task in privileged mode:

Task	Command
Clear the “from” address.	**clear logging callhome from**

This example shows how to clear the “from” address:

```
Console> (enable) clear logging callhome from
Cleared the from address field of callhome messages.
Console> (enable)
```

To clear the “reply to” address, perform this task in privileged mode:

Task	Command
Clear the “reply to” address.	**clear logging callhome reply-to**

This example shows how to clear the “reply to” address:

```
Console> (enable) clear logging callhome reply-to
Cleared the reply-to address field of callhome messages.
Console> (enable)
```

To clear an SMTP server from the list of CallHome SMTP servers, perform this task in privileged mode:

Task	Command
Clear an SMTP server.	**clear logging callhome smtp-server** *IP Address*

This example shows how to delete the SMTP server 172.20.8.16 from the list of CallHome servers:

```
Console> (enable) clear logging callhome smtp-server 172.20.8.16
Removed 172.20.8.16 from the table of callhome SMTP servers.
Console> (enable)
```

To clear the CallHome severity level, perform this task in privileged mode:

Task	Command
Clear the CallHome severity level.	**clear logging callhome severity**

This example shows how to clear the CallHome severity level:

```
Console> (enable) clear logging callhome severity
Cleared callhome severity level to its default value of 2(LOG_CRIT).
Console> (enable)
```

CHAPTER 30

Configuring DNS

This chapter describes how to configure the Domain Name System (DNS) on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Understanding How DNS Works

DNS is a distributed database with which you can map the host names to the IP addresses through the DNS protocol from a DNS server. When you configure DNS on the switch, you can substitute the host name for the IP address with all IP commands, such as **ping**, **telnet**, **upload**, and **download**.

To use DNS, you must have a DNS name server on your network.

You can specify a primary DNS name server on the switch and two backup servers. The first server that is specified is the primary server unless you explicitly identify the primary server. The switch sends the DNS queries to the primary server first. If the query to the primary server fails, the backup servers are queried.

DNS Default Configuration

Table 30-1 shows the default DNS configuration.

Table 30-1 DNS Default Configuration

Feature	Default Value
DNS enable state	Disabled
DNS default domain name	Null
DNS servers	None specified

Configuring DNS on the Switch

These sections describe how to configure DNS:

Setting Up and Enabling DNS

To set up and enable DNS on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Specify the IP address of one or more DNS servers.	**set ip dns server** *ip_addr* [**primary**]
Step 2	Set the domain name.	**set ip dns domain** *name*
Step 3	Enable DNS.	**set ip dns enable**
Step 4	Verify the DNS configuration.	**show ip dns** [**noalias**]

This example shows how to set up and enable DNS on the switch and verify the configuration:

```
Console> (enable) set ip dns server 10.2.2.1
10.2.2.1 added to DNS server table as primary server.
Console> (enable) set ip dns server 10.2.24.54 primary
10.2.24.54 added to DNS server table as primary server.
Console> (enable) set ip dns server 10.12.12.24
10.12.12.24 added to DNS server table as backup server.
Console> (enable) set ip dns domain corp.com
Default DNS domain name set to corp.com
Console> (enable) set ip dns enable
DNS is enabled
```

```
Console> (enable) show ip dns
DNS is currently enabled.
The default DNS domain name is: corp.com

DNS name server                          status
---------------------------------------  -------
dns_serv2
dns_serv1                                primary
dns_serv3
Console> (enable)
```

Clearing a DNS Server

To clear the DNS servers from the DNS server table, perform this task in privileged mode:

	Task	Command
Step 1	Clear one or all of the DNS servers from the table.	**clear ip dns server** [*ip_addr* | **all**]
Step 2	Verify the DNS configuration.	**show ip dns** [**noalias**]

This example shows how to clear a DNS server from the DNS server table:

```
Console> (enable) clear ip dns server 10.12.12.24
10.12.12.24 cleared from DNS table
Console> (enable)
```

This example shows how to clear all of the DNS servers from the DNS server table:

```
Console> (enable) clear ip dns server all
All DNS servers cleared
Console> (enable)
```

Clearing the DNS Domain Name

To clear the default DNS domain name, perform this task in privileged mode:

	Task	Command
Step 1	Clear the default DNS domain name.	**clear ip dns domain**
Step 2	Verify the DNS configuration.	**show ip dns** [**noalias**]

This example shows how to clear the default DNS domain name:

```
Console> (enable) clear ip dns domain
Default DNS domain name cleared.
Console> (enable)
```

Disabling DNS

To disable DNS, perform this task in privileged mode:

	Task	Command
Step 1	Disable DNS on the switch.	**set ip dns disable**
Step 2	Verify the DNS configuration.	**show ip dns** [**noalias**]

This example shows how to disable DNS on the switch:

```
Console> (enable) set ip dns disable
DNS is disabled
Console> (enable)
```

CHAPTER 31

Configuring CDP

This chapter describes how to configure the Cisco Discovery Protocol (CDP) on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Understanding How CDP Works

CDP was enhanced in software release 8.1(1) to facilitate the backward compatibility with the newer, higher-powered Cisco IP phones. With this enhanced CDP, a Cisco IP phone can negotiate its power requirements to the switch within the CDP packet. The switch uses this information to ensure that it does not oversubscribe the available power.

CDP is a media- and protocol-independent protocol that runs on all the Cisco-manufactured equipment including routers, bridges, access and communication servers, and switches. Using CDP, you can view information about all the Cisco devices that are directly attached to the switch. In addition, CDP detects the native VLAN and port duplex mismatches.

The network management applications can retrieve the device type and SNMP-agent address of the neighboring Cisco devices using CDP. This feature enables the applications to send the SNMP queries to the neighboring devices. CDP allows the network management applications to discover the Cisco devices that are the neighbors of the already known devices, in particular, the neighbors running the lower-layer, transparent protocols.

CDP runs on all media that support Subnetwork Access Protocol (SNAP). CDP runs over the data link layer only.

The Cisco devices never forward the CDP packets. When new CDP information is received, the Cisco devices discard old information.

Default CDP Configuration

Table 31-1 shows the default CDP configuration.

Table 31-1 CDP Default Configuration

Feature	Default Value
CDP global enable state	Enabled
CDP port enable state	Enabled on all ports
CDP message interval	60 seconds
CDP holdtime	180 seconds

Note The CDP message interval can be configured in the range of 5–254 seconds.

Configuring CDP on the Switch

These sections describe how to configure CDP:

- Setting the CDP Global Enable and Disable States, page 31-2
- Setting the CDP Enable and Disable States on a Port, page 31-3
- Setting the CDP Message Interval, page 31-4
- Setting the CDP Holdtime, page 31-4
- Displaying CDP Neighbor Information, page 31-5

Setting the CDP Global Enable and Disable States

To set the CDP global enable state, perform this task in privileged mode:

	Task	Command
Step 1	Set the CDP global enable state on the switch.	**set cdp** {**enable** | **disable**}
Step 2	Verify the CDP configuration.	**show cdp**

This example shows how to enable CDP globally and verify the configuration:

```
Console> (enable) set cdp enable
CDP enabled globally
Console> (enable) show cdp
CDP               : enabled
Message Interval  : 60
Hold Time         : 180
Console> (enable)
```

This example shows how to disable CDP globally and verify the configuration:

```
Console> (enable) set cdp disable
CDP disabled globally
```

```
Console> (enable) show cdp
CDP               : disabled
Message Interval  : 60
Hold Time         : 180
Console> (enable)
```

Setting the CDP Enable and Disable States on a Port

You can enable or disable CDP on a per-port basis. You must enable CDP globally before the switch will transmit the CDP messages on any ports.

To set the CDP enable state on a per-port basis, perform this task in privileged mode:

	Task	Command
Step 1	Set the CDP enable state on the individual ports.	**set cdp** {**enable** | **disable**} [*mod*/*port*]
Step 2	Verify the CDP configuration.	**show cdp port** [*mod*[/*port*]]

This example shows how to enable CDP on ports 3/1-2 and verify the configuration:

```
Console> (enable) set cdp enable 3/1-2
CDP enabled on ports 3/1-2.
Console> (enable) show cdp port 3
CDP               : enabled
Message Interval  : 60
Hold Time         : 180

Port      CDP Status
--------  ----------
 3/1      enabled
 3/2      enabled
 3/3      disabled
 3/4      disabled
 3/5      disabled
 3/6      disabled
 3/7      enabled
 3/8      enabled
 3/9      enabled
 3/10     enabled
 3/11     enabled
 3/12     enabled
Console> (enable)
```

This example shows how to disable CDP on ports 3/1-6 and verify the configuration:

```
Console> (enable) set cdp disable 3/1-6
CDP disabled on ports 3/1-6.
Console> (enable) show cdp port 3
CDP               : enabled
Message Interval  : 60
Hold Time         : 180

Port      CDP Status
--------  ----------
 3/1      disabled
 3/2      disabled
 3/3      disabled
 3/4      disabled
 3/5      disabled
 3/6      disabled
```

```
 3/7      enabled
 3/8      enabled
 3/9      enabled
 3/10     enabled
 3/11     enabled
 3/12     enabled
Console> (enable)
```

Setting the CDP Message Interval

The CDP message interval specifies how often the switch will transmit the CDP messages to the directly connected Cisco devices.

To set the default CDP message interval, perform this task in privileged mode:

	Task	Command
Step 1	Set the default CDP message interval. The allowed range is 5–900 seconds.	**set cdp interval** *interval*
Step 2	Verify the CDP configuration.	**show cdp**

This example shows how to set the default CDP message interval to 100 seconds and verify the configuration:

```
Console> (enable) set cdp interval 100
CDP message interval set to 100 seconds for all ports.
Console> (enable) show cdp
CDP               : enabled
Message Interval  : 100
Hold Time         : 180
Console> (enable)
```

Setting the CDP Holdtime

The CDP holdtime specifies how much time can pass between the CDP messages from the neighboring devices before the device is no longer considered connected and the neighboring entry is aged out.

To set the default CDP holdtime, perform this task in privileged mode:

	Task	Command
Step 1	Set the default CDP holdtime. The allowed range is 10–255 seconds.	**set cdp holdtime** *interval*
Step 2	Verify the CDP configuration.	**show cdp**

This example shows how to set the default CDP holdtime to 225 seconds and verify the configuration:

```
Console> (enable) set cdp holdtime 225
CDP holdtime set to 225 seconds.
Console> (enable) show cdp
CDP               : enabled
Message Interval  : 100
Hold Time         : 225
Console> (enable)
```

Displaying CDP Neighbor Information

To display information about the directly connected Cisco devices, enter the **show cdp neighbors** command. Enter the **vlan** keyword to display the native VLAN for the connected ports. Enter the **duplex** keyword to display the duplex mode for the connected ports. Enter the **capabilities** keyword to display the device capability codes for the connected device. Enter the **detail** keyword to display the detailed information about the neighboring device.

Note If you enter the **show cdp neighbors** command for a device that supports earlier versions of CDP, "unknown" is displayed in the following fields: VTP Management Domain, Native VLAN, and Duplex.

To display information about the directly connected Cisco devices, perform this task:

Task	Command
Display information about the CDP neighbors.	**show cdp neighbors** [*mod*[*/port*]] [**vlan** | **duplex** | **capabilities** | **detail**]

This example shows how to display the CDP neighbor information for the connected Cisco devices:

```
Console> (enable) show cdp neighbors
* - indicates vlan mismatch.
# - indicates duplex mismatch.
Port     Device-ID                      Port-ID                   Platform
-------- ------------------------------ ------------------------- ------------
 2/3     JAB023807H1(2948)              2/2                       WS-C2948
 3/1     JAB023806JR(4003)              2/1                       WS-C4003
 3/2     JAB023806JR(4003)              2/2                       WS-C4003
 3/5     JAB023806JR(4003)              2/5                       WS-C4003
 3/6     JAB023806JR(4003)              2/6                       WS-C4003
Console> (enable)
```

This example shows how to display the native VLAN for each port that is connected on the neighboring device (there is a native VLAN mismatch between port 3/6 on the local switch and port 2/6 on the neighboring device, as indicated by the asterisk [*]):

```
Console> (enable) show cdp neighbors vlan
* - indicates vlan mismatch.
# - indicates duplex mismatch.
Port     Device-ID                      Port-ID                   NativeVLAN
-------- ------------------------------ ------------------------- ----------
 2/3     JAB023807H1(2948)              2/2                       522
 3/1     JAB023806JR(4003)              2/1                       100
 3/2     JAB023806JR(4003)              2/2                       100
 3/5     JAB023806JR(4003)              2/5                       1
 3/6     JAB023806JR(4003)              2/6*                      1
Console> (enable)
```

This example shows how to display detailed information about the neighboring device:

```
Console> (enable) show cdp neighbors 2/3 detail
Port (Our Port): 2/3
Device-ID: JAB023807H1(2948)
Device Addresses:
  IP Address: 172.20.52.36
Holdtime: 132 sec
Capabilities: TRANSPARENT_BRIDGE SWITCH
Version:
  WS-C2948 Software, Version McpSW: 5.1(57) NmpSW: 5.1(1)
  Copyright (c) 1995-1999 by Cisco Systems, Inc.
Platform: WS-C2948
Port-ID (Port on Neighbors's Device): 2/2
VTP Management Domain: Lab_Network
Native VLAN: 522
Duplex: full
Console> (enable)
```

CHAPTER 32

Configuring UDLD

This chapter describes how to configure the UniDirectional Link Detection (UDLD) protocol on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Understanding How UDLD Works

The UDLD protocol allows devices that are connected through the fiber-optic or copper (for example, Category 5 cabling) Ethernet cables to monitor the physical configuration of the cables and detect when a unidirectional link exists. When a unidirectional link is detected, UDLD shuts down the affected port and alerts the user. The unidirectional links can cause a variety of problems, including spanning-tree topology loops.

UDLD is a Layer 2 protocol that works with the Layer 1 mechanisms to determine the physical status of a link. At Layer 1, autonegotiation takes care of physical signaling and fault detection. UDLD performs the tasks that autonegotiation cannot perform such as detecting the identities of neighbors and shutting down the misconnected ports. When you enable both autonegotiation and UDLD, the Layer 1 and Layer 2 detections work together to prevent the physical and logical unidirectional connections and the malfunctioning of other protocols.

A unidirectional link occurs whenever the traffic that is transmitted by the local device over a link is received by the neighbor but the traffic that is transmitted from the neighbor is not received by the local device. If one of the fiber strands in a pair is disconnected, as long as autonegotiation is active, the link does not stay up. In this case, the logical link is undetermined, and UDLD does not take any action. If both fibers are working normally from a Layer 1 perspective, then UDLD at Layer 2 determines whether those fibers are connected correctly and whether the traffic is flowing bidirectionally between the right neighbors. This check cannot be performed by autonegotiation, because autonegotiation is a Layer 1 mechanism.

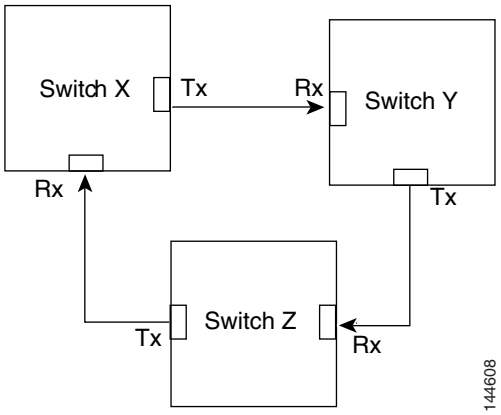
The switch periodically transmits the UDLD messages (packets) to the neighbor devices on the ports that have UDLD enabled. If the messages are echoed back to the sender within a specific time frame and they are lacking a specific acknowledgment (echo), the link is flagged as unidirectional and the port is shut down. The devices on both ends of the link must support UDLD in order for the protocol to successfully identify and disable the unidirectional links.

Note With supervisor engine software release 5.4(3) and later releases, you can specify the message interval between UDLD messages. Previously, the message interval was fixed at 60 seconds. With a configurable message interval, UDLD reacts much faster to link failures.

Note By default, UDLD is locally disabled on the copper ports to avoid sending unnecessary control traffic on this type of media since it is often used for the access ports.

Figure 32-1 shows an example of a unidirectional link condition. Each switch can send packets to a neighbor switch but is not able to receive packets from the same switch that it is sending packets to. UDLD detects and disables these one-way connections.

Figure 32-1 Unidirectional Link



Default UDLD Configuration

Table 32-1 shows the default UDLD configuration.

Table 32-1 UDLD Default Configuration

Feature	Default Value
UDLD global enable state	Globally disabled
UDLD per-port enable state for fiber-optic media	Enabled on all Ethernet fiber-optic ports
UDLD per-port enable state for twisted-pair (copper) media	Disabled on all Ethernet 10/100 and 1000BASE-TX ports
UDLD message interval	15 seconds
UDLD aggressive mode	Disabled

Configuring UDLD on the Switch

These sections describe how to configure UDLD:

Enabling UDLD Globally

To enable UDLD globally on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Enable UDLD globally on the switch.	**set udld enable**
Step 2	Verify the configuration.	**show udld**

This example shows how to enable UDLD globally and verify the configuration:

```
Console> (enable) set udld enable
UDLD enabled globally
Console> (enable) show udld
UDLD      : enabled
Console> (enable)
```

Enabling UDLD on Individual Ports

To enable UDLD on the individual ports, perform this task in privileged mode:

	Task	Command
Step 1	Enable UDLD on a specific port.	**set udld enable** *mod/port*
Step 2	Verify the configuration.	**show udld port** [*mod*[**/***port*]]

This example shows how to enable UDLD on port 4/1 and verify the configuration:

```
Console> (enable) set udld enable 4/1
UDLD enabled on port 4/1
Console> (enable) show udld port 4/1
UDLD      : enabled
Message Interval: 15 seconds
Port      Admin Status  Aggressive Mode Link State
--------  ------------  --------------- ---------
 4/1      enabled       disabled        bidirectional
Console> (enable)
```

Disabling UDLD on Individual Ports

To disable UDLD on the individual ports, perform this task in privileged mode:

	Task	Command
Step 1	Disable UDLD on a specific port.	**set udld disable** *mod/port*
Step 2	Verify the configuration.	**show udld port** [*mod*[**/***port*]]

This example shows how to disable UDLD on port 4/1:

```
Console> (enable) set udld disable 4/1
UDLD disabled on port 4/1.
Console> (enable)
```

Disabling UDLD Globally

To disable UDLD globally on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Disable UDLD globally on the switch.	**set udld disable**
Step 2	Verify the configuration.	**show udld**

This example shows how to disable UDLD globally on the switch:

```
Console> (enable) set udld disable
UDLD disabled globally
Console> (enable)
```

Specifying the UDLD Message Interval

To specify the UDLD message interval, perform this task in privileged mode:

	Task	Command
Step 1	Specify the UDLD message interval.	**set udld interval** *interval*
Step 2	Verify the configuration.	**show udld**

This example shows how to specify the UDLD message interval on the switch:

```
Console> (enable) set udld interval 20
UDLD message interval set to 20 seconds
Console> (enable)
```

This example shows how to verify the message interval on the switch:

```
Console> (enable) show udld
UDLD             : enabled
Message Interval  : 20 seconds
Console> (enable)
```

Enabling UDLD Aggressive Mode

Software release 5.4(3) and later releases have UDLD aggressive mode. UDLD aggressive mode is disabled by default and its use is recommended only for point-to-point links between the Cisco switches running software release 5.4(3) or later releases. With UDLD aggressive mode enabled, when a port on a bidirectional link that has a UDLD neighbor relationship established stops receiving the UDLD packets, UDLD tries to reestablish the connection with the neighbor. After eight failed retries, the port is put into errdisable state.

To prevent the spanning-tree loops, normal UDLD with the default interval of 15 seconds is fast enough to shut down a unidirectional link before a blocking port transitions to the forwarding state (when the default spanning-tree parameters are used).

Enabling UDLD aggressive mode provides additional benefits in the following cases:

- One side of a link has a port stuck (both Tx and Rx)
- One side of a link remains up while the other side of the link has gone down

In these cases, UDLD aggressive mode errdisables one of the ports on the link and stops discarding the traffic. Even with aggressive mode disabled, there would be no risk for a broadcast storm due to a spanning-tree loop, because one port is unable to pass the traffic in both directions.

To enable UDLD aggressive mode, perform this task in privileged mode:

	Task	Command
Step 1	Enable UDLD aggressive mode.	**set udld aggressive-mode enable** *mod/port*
Step 2	Verify the configuration.	**show udld**

This example shows how to enable UDLD aggressive mode on the switch:

```
Console> (enable) set udld aggressive-mode enable 4/1
Aggressive UDLD enabled on port 4/1.
Console> (enable)
```

This example shows how to verify that UDLD aggressive mode is enabled on the switch:

```
Console> (enable) show udld port 4/1
UDLD      : enabled
Message Interval: 30 seconds
Port      Admin Status  Aggressive Mode Link State
--------  ------------  --------------- ---------
 4/1      enabled       Enabled         bidirectional
Console> (enable)
```

Displaying the UDLD Configuration

To display the UDLD enable state, perform this task in privileged mode:

Task	Command
Display the UDLD enable state.	**show udld**

This example shows how to display the UDLD enable state:

```
Console> (enable) show udld
UDLD              : enabled
Message Interval  : 15 seconds
Console> (enable)
```

To display UDLD configuration for a module or port, perform this task in privileged mode:

Task	Command
Display the UDLD configuration for a module or port.	**show udld port** [*mod*] [*mod/port*]

This example shows how to display the UDLD configuration for ports on module 4:

```
Console> (enable) show udld port 4
UDLD      : enabled
Message Interval: 15 seconds
Port      Admin Status  Aggressive Mode Link State
--------  ------------  --------------- ---------
 4/1      enabled       disabled        bidirectional
 4/2      enabled       disabled        bidirectional
 4/3      enabled       disabled        undetermined
 4/4      enabled       disabled        bidirectional

.
.

Console> (enable)
```

Table 32-2 describes the fields in the **show udld** command output.

Table 32-2 show udld Command Output Fields

Field	Description
UDLD	Status of whether UDLD is enabled or disabled.
Message Interval	Message interval in seconds.
Port	Module and port number(s).
Admin Status	Status of whether administration status is enabled or disabled.
Aggressive Mode	Status of whether aggressive mode is enabled or disabled.
Link State	Status of the link: undetermined (detection in progress or UDLD on the neighbors has been disabled), not applicable (UDLD and/or the local port has been manually disabled), shutdown (unidirectional link has been detected and the port errdisabled), or bidirectional (a bidirectional link has been detected since the port is functioning properly in both directions).

CHAPTER 33

Configuring DHCP Snooping and IP Source Guard

This chapter describes how to configure Dynamic Host Configuration Protocol (DHCP) snooping and IP source guard on the Catalyst 6500 series switches.

This chapter consists of these sections:

Note For complete syntax and usage information for the switch commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* and related publications at http://www.cisco.com/en/US/docs/switches/lan/catalyst6500/catos/8.x/command/reference/cmd_ref.html

Understanding How DHCP Snooping Works

DHCP snooping provides the security against the Denial-Of-Service (DoS) attacks that are launched using the DHCP messages by filtering the DHCP packets and building and maintaining a DHCP-snooping binding table. DHCP snooping uses both trusted and untrusted ports.

The DHCP packets that are received from a trusted port are forwarded without validation. Typically, the trusted ports are used to reach a DHCP server or relay agent. When the switch receives the DHCP packets from an untrusted port, DHCP snooping validates that only the DHCP packets from the clients are allowed and verifies that no spoofing of information is occurring.

The DHCP-snooping binding table contains the MAC address, IP address, lease time in seconds, and VLAN port information for the DHCP clients on the untrusted ports of a switch. The information that is contained in a DHCP-snooping binding table is removed from the binding table once its lease expires or DHCP snooping is disabled in the VLAN.

Note In software release 8.6(1) and later releases, you can enable DHCP snooping on a per-port basis.

These DHCP messages are used to build the DHCP binding table:

- DHCPACK—Adds a new dynamic DHCP binding entry if the binding entry does not already exist.
- DHCPNAK—Deletes an existing DHCP binding entry.
- DHCPRELEASE—Deletes a dynamic DHCP binding entry if the binding entry exists.
- DHCPDECLINE—Deletes a dynamic DHCP binding entry if the binding entry exists.

Each switch maintains a DHCP-snooping binding table for only the local untrusted ports. The table does not store information about the DHCP-snooping binding table for the hosts that are directly connected to other switches, and it does not contain information about the hosts that are connected through a trusted port. A trusted port has an entity, such as a relay agent or DHCP server, that is directly connected or is the forwarding path to such an entity. Any path to a relay agent or DHCP server should be trusted.

DHCP Snooping Configuration Guidelines

This section describes the guidelines for configuring DHCP snooping in your network:

- In software release 8.6(1) and later releases, you can enable DHCP snooping on a per-port basis.
- If you do a non-high availability switchover with DHCP snooping enabled, you will lose the contents of the DHCP-snooping binding table. We do not recommend using this configuration.
- DHCP snooping is supported on the Policy Feature Card (PFC) and later versions.
- The DHCP-snooping binding table is limited to 16,384 entries. Once the limit is reached, no new entries can be added until the lease time is reached on the older entries.
- 802.1X-DHCP and DHCP snooping are mutually exclusive. You should not configure a VLAN for both 802.1X-DHCP and DHCP snooping. If you configure both 802.1X and DHCP snooping in your ACL, the feature that is positioned higher up in the ACL overrides the other feature.
- We recommend that you enable high availability when using dynamic ARP inspection (DAI), DHCP snooping, and IP source guard. If high availability is not enabled, the clients have to renew their IP addresses for these features to work after a switchover. For configuration details on DAI, see the "Dynamic ARP Inspection" section on page 15-39.

Configuring DHCP Snooping on a VLAN

Typically, DHCP snooping is used at the access-level network, such as a wiring closet. When you enable DHCP snooping on a VLAN, it builds a table of IP addresses to MAC-address bindings for the DHCP clients on that VLAN.

Note In software release 8.6(1) and later releases, you can enable DHCP snooping on a per-port basis.

Note In software release 8.5(1) and later releases, you can enable DHCP snooping on the management VLANs sc0 and sc1.

These sections describe how to configure DHCP snooping:

Default Configuration for DHCP Snooping

DHCP snooping is disabled by default. Table 33-1 shows the default configuration values for each DHCP-snooping option. If you want to change the default configuration values, see the "Enabling DHCP Snooping" section on page 33-4.

Table 33-1 Default Configuration Values for DHCP Snooping

Option	Default Value/State
DHCP-snooping host tracking information option	Disabled.
DHCP-snooping limit rate	1000 pps shared with ARP inspection and 802.1X-DHCP. Rate limiting is supported on PFC2 and later versions.
DHCP-snooping trust on a port	Untrusted.
DHCP snooping on a VLAN	Disabled.
DHCP-snooping bindings-database auto-save option	Disabled.
DHCP-snooping bindings-database storage device and filename	bootflash:dhcp-snooping-bindings-database

Enabling DHCP Snooping

DHCP snooping is enabled on the VLANs through the security VLAN access control lists (VACLs). DHCP snooping is enabled on a VLAN by adding a DHCP-snooping access control entry (ACE) to a new or existing security ACL. You must determine where to position DHCP snooping in the ACL depending on your policy for the DHCP packets. For example, if you want to deny the DHCP packets that come from a certain host and perform DHCP snooping for the other DHCP packets, then you must place a deny ACE before the DHCP-snooping ACE.

To enable DHCP snooping on a VLAN, perform this task in privileged mode:

	Task	Command
Step 1	Add DHCP snooping to the VACL.	**set security acl ip** *acl_name* **permit dhcp-snooping**
Step 2	Configure the VACL to allow DHCP snooping from all hosts.	**set security acl ip** *acl_name* **permit ip any any**

	Task	Command
Step 3	Save the VACL.	**commit security acl** *acl_name*
Step 4	Add an ACL to a VLAN.	**set security acl map** *acl_name* **10**

This example shows how to configure DHCP snooping on a VLAN:

```
Console> (enable) set security acl ip dhcpsnoop permit dhcp-snooping
Successfully configured DHCP Snooping for ACL dhcpsnoop. Use 'commit' command to save
changes.
Console> (enable) set security acl ip dhcpsnoop permit ip any any
dhcpsnoop editbuffer modified. Use 'commit' command to apply changes.
Console> (enable) commit security acl dhcpsnoop
ACL commit in progress.

ACL 'dhcpsnoop' successfully committed.
Console> (enable) set security acl map dhcpsnoop 10
Mapping in progress.

ACL dhcpsnoop successfully mapped to VLAN 10.
Console> (enable)
```

Note If you create a VACL just for enabling DHCP snooping, the VACL has an implicit deny at the end and no other packets are allowed unless there is an explicit permit for those packets.

Note 802.1X-DHCP and DHCP snooping are mutually exclusive. Do not configure a VLAN with both features.

Enabling DHCP Snooping on a Private VLAN

You must enable DHCP snooping separately on the primary and secondary (isolated or community) private VLANs (PVLANs). The DHCP-snooping binding table contains binding information about the primary VLAN only and not the secondary VLANs. If you enable DHCP snooping on a PVLAN and not on the secondary VLAN, the DHCP-snooping binding table entries are not added, even though the packet is seen on the PVLAN.

Enabling the DHCP-Snooping Host-Tracking Information Option

If you enable the host-tracking information option, the DHCP relay agent information option (option 82) is added to the client packets that are being forwarded. The relay agent option contains the agent circuit ID and the agent remote ID information. The circuit ID suboption contains the port and the VLAN number of the client. The remote ID suboption contains the MAC address of the switch. Before inserting the host-tracking information, the switch verifies that the DHCP messages do not have an existing relay information option or a nonzero giaddr field. Before removing the host-tracking information, the switch verifies that the DHCP reply messages are from a trusted port and that the MAC address of the remote ID and the local switch match. If the packet comes from a trusted port and the addresses do not match, the packet is forwarded.

To configure the host-tracking information option for DHCP snooping, perform this task in privileged mode:

	Task	Command
Step 1	Enable the DHCP-snooping host-tracking information option.	**set dhcp-snooping information host-tracking enable**
Step 2	Display the MAC address for the host-tracking information option.	**show dhcp-snooping config**

This example shows how to configure the DHCP-snooping host-tracking information option:

```
Console> (enable) set dhcp-snooping information host-tracking enable
DHCP Snooping Information Option Enabled.
Console> (enable) show dhcp-snooping config
DHCP Snooping MAC address matching is enabled.
DHCP Snooping host-tracking information option is enabled.
Remote ID used in information option is 00-d0-00-4c-1b-ff.
Console> (enable)
```

Enabling the DHCP Snooping MAC-Address Matching Option

If you enable the MAC-address matching option, the source MAC address in the Ethernet header is matched with the chaddr field in the DHCP payload for the DHCP packets that are coming from the untrusted ports. If the match fails, the packets are dropped and the counter for the packets that are dropped on the untrusted ports is incremented. This feature is enabled by default.

To configure the MAC-address matching option for DHCP snooping, perform this task in privileged mode:

	Task	Command
Step 1	Enable the DHCP-snooping MAC-address matching option.	**set dhcp-snooping match-mac enable**
Step 2	Display the DHCP-snooping configuration.	**show dhcp-snooping config**

This example shows how to configure the DHCP-snooping MAC-address matching option:

```
Console> (enable) set dhcp-snooping match-mac enable
DHCP Snooping MAC address matching enabled.
Console> (enable) show dhcp-snooping config
DHCP Snooping MAC address matching is enabled.
DHCP Snooping host-tracking information option is enabled.
Remote ID used in information option is 00-d0-00-4c-1b-ff.
Console> (enable)
```

Configuration Examples for DHCP Snooping

These configuration examples show how to enable DHCP snooping.

Example 1: Enabling DHCP Snooping

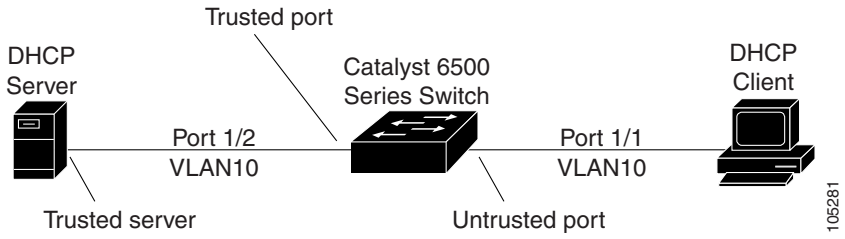
This example shows how to enable DHCP snooping for VLAN 10 with a DHCP server on port 1/2:

```
Console> (enable) set security acl ip dhcpsnoop permit dhcp-snooping
Successfully configured DHCP Snooping for ACL dhcpsnoop. Use 'commit' command to
 save changes.
Console> (enable) set security acl ip dhcpsnoop permit ip any any
dhcpsnoop editbuffer modified. Use 'commit' command to apply changes.
Console> (enable) commit security acl dhcpsnoop
ACL commit in progress.

ACL 'dhcpsnoop' successfully committed.
Console> (enable) set security acl map dhcpsnoop 10
Mapping in progress.

ACL dhcpsnoop successfully mapped to VLAN 10.
Console> (enable) set port dhcp-snooping 1/2 trust enable
Port(s)  1/2 state set to trusted for DHCP Snooping.
Console> show dhcp-snooping config
DHCP Snooping MAC address matching is enabled.
DHCP Snooping host-tracking information option is disabled.
Remote ID used in information option is 00-d0-00-4c-1b-ff.
Console> show port dhcp-snooping 1/1-2
Port     Trust
----     ------
1/1     untrusted
1/2     trusted
Console> (enable)
```

Note If you want to configure DHCP-snooping host tracking after enabling DHCP snooping, enter the **set dhcp-snooping information-option host-tracking** command.

Figure 33-1 shows the typical topology that is used when you configure DHCP snooping in a client/server network.

Figure 33-1 DHCP Snooping Configured for a Client and Server



Example 2: Enabling DHCP Snooping with an MSFC as a DHCP Relay Agent

This example shows how to configure the Multilayer Switch Feature Card (MSFC) as a relay agent with the DHCP host tracking enabled.

Note In this example, the client is untrusted and accesses the switch with the MSFC as a relay agent. The MSFC relay agent switch connects to the MSFC DHCP server switch through a trusted trunk port.

This example shows how to configure the MSFC as a DHCP relay agent:

```
service dhcp
on int vlan 810
    ip address 192.168.80.241 255.255.255.0
    ip helper-address 192.168.94.247
    ip dhcp relay information trusted
on int vlan 4094
    ip address 192.168.94.241 255.255.255.0
```

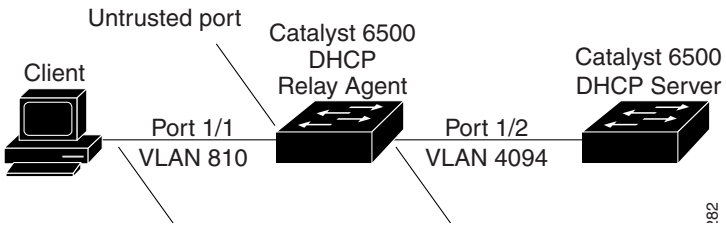
This example shows how to configure the MSFC as a DHCP server:

```
service dhcp
ip dhcp excluded-address 192.168.80.241
!
ip dhcp pool net810
   network 192.168.80.0 255.255.255.0
on int vlan 4094
   ip address 192.168.94.247 255.255.255.0
```

Note The MSFC port is configured by the system as a DHCP-snooping trusted port.

Figure 33-2 shows the typical topology that is used when you configure the MSFC as a relay agent.

Figure 33-2 MSFC as a Relay Agent



Example 3: Enabling DHCP Snooping in Port-Based Mode

The following example shows how to enable DHCP snooping in port-based mode with a router as the MSFC. The DHCP-snooping ACL is mapped to the host VLAN.

```
Console> (enable) set security acl map  dhcp 1/2
Mapping in progress.
ACL dhcp successfully mapped to port(s) 1/2
Console> (enable) set security acl map dhcp 16
Mapping in progress.

ACL dhcp successfully mapped to VLAN 16.
```

Enter the **show** command to display the security-acl mode:

```
Console> (enable) show port security-acl 1/2
Port  Interface Type Interface Type Interface Merge Status
config         runtime        runtime
----- -------------- -------------- ----------------------
1/2      port-based     port-based         not applicable

Config:
Port  ACL name                          Type
----- -------------------------------- ----
1/2  dhcp                              IP

Runtime:
Port  ACL name                          Type
----- -------------------------------- ----
1/2  dhcp                              IP

dhcp-snooping:
Port     Trust        Source-Guard    Source-Guarded IP Addresses
-----    -----------  ------------    ---------------------------
1/2     untrusted    disabled

Port     Binding Limit         No. of Existing Bindings
-----    ------------------    ------------------------
1/2     32                    0
```

Enter the **show** command to verify the mapping:

```
Console> (enable) show security acl map config all
ACL Name                          Type Ports/Vlans
--------------------------------- ---- -----------------
dhcp                              IP   16
dhcp                              IP   1/2
```

The following example shows how to enable DHCP snooping in port-based mode with an external router configuration. DHCP snooping ACL is mapped to the host and the DHCP server port.

Note Both the host and server ports are in port-based security ACL mode.

```
Console> (enable) set port security-acl 1/2 port-based
Warning: Vlan-based ACL features will be disabled on ports 1/2
ACL interface is set to port-based mode for port(s) 1/2.

Console> (enable) set port security-acl 5/2 port-based
Warning: Vlan-based ACL features will be disabled on ports 5/2
ACL interface is set to port-based mode for port(s) 5/2.

Console> (enable) set security acl map dhcp 1/2
Mapping in progress.
ACL dhcp successfully mapped to port(s) 1/2
Console> (enable) set security acl map dhcp 5/2
Mapping in progress.
ACL dhcp successfully mapped to port(s) 5/2
```

Enter the **show** command to display the security ACL mode:

```
Console> (enable) show port security-acl 1/2
Port  Interface Type Interface Type Interface Merge Status
config         runtime        runtime
----- -------------- -------------- ----------------------
1/2      port-based     port-based         not applicable
```

```
Config:
Port  ACL name                          Type
----- -------------------------------- ----
1/2  dhcp                              IP

Runtime:
Port  ACL name                          Type
----- -------------------------------- ----
1/2  dhcp                              IP

dhcp-snooping:
Port     Trust        Source-Guard    Source-Guarded IP Addresses
-----    -----------  ------------    ---------------------------
1/2     untrusted    disabled

Port     Binding Limit         No. of Existing Bindings
-----    ------------------    ------------------------
1/2     32                    0

Console> (enable) show port security-acl 5/2
Port  Interface Type Interface Type Interface Merge Status
config         runtime        runtime
----- -------------- -------------- ----------------------
5/2      port-based     port-based         not applicable

Config:
Port  ACL name                          Type
----- -------------------------------- ----
5/2  dhcp                              IP

Runtime:
Port  ACL name                          Type
----- -------------------------------- ----
5/2  dhcp                              IP

dhcp-snooping:
Port     Trust        Source-Guard    Source-Guarded IP Addresses
-----    -----------  ------------    ---------------------------
5/2     trusted      disabled

Port     Binding Limit         No. of Existing Bindings
-----    ------------------    ------------------------
5/2     32                    0
```

Enter the **show** command to verify the ACL mappings:

```
Console> (enable) show security acl map config all
ACL Name                         Type Ports/Vlans
-------------------------------- ---- ---------------
dhcp                             IP   1/2,5/2
No Mappings have been defined for any vlan yet.
```

Specifying the DHCP-Snooping Binding Limit on a Per-Port Basis

Use the **set port dhcp-snooping** *mod/port* **binding-limit** *count* command to specify the DHCP-snooping binding limit on a per-port basis. The minimum binding limit is 1, the maximum is 1024, and the default is 32. To specify the DHCP-snooping binding limit on a per-port basis, perform this task in privileged mode:

	Task	Command
Step 1	Specify the DHCP-snooping binding limit on a per-port basis.	**set port dhcp-snooping** *mod/port* **binding-limit** *count*
Step 2	Display the DHCP-snooping configuration.	**show port dhcp-snooping** [*mod*[/*ports*]]
Step 3	Display the static binding information.	**show dhcp-snooping bindings**
Step 4	Clear static bindings.	**clear dhcp-snooping binding** [port *mod/port*] [vlan *vlanid*] *IP Address MAC Address*

This example shows how to set the DHCP-snooping binding limit to 48 on port 5/9:

```
Console> (enable) set port dhcp-snooping 5/9 binding-limit 48
Port 5/9, DHCP Snooping binding limit set to 48
Console> (enable)
```

This example shows how to display the DHCP-snooping binding limit on port 5/9:

```
Console> (enable) show port dhcp-snooping 5/9
Port     Trust        Source-Guard    Source-Guarded IP Addresses
-----    -----------  ------------    ---------------------------
 5/9     untrusted    disabled

Port     Binding Limit
-----    ------------------
 5/9     48
Console> (enable)
```

This example shows how to display DHCP-snooping static bindings:

```
Console (enable) show dhcp-snooping bindings
MAC Address          IP Address       Lease(sec)    VLAN     Port
------------------   --------------   ----------    -------- -------------

00-01-7b-9b-05-3f   172.20.52.67     permanent       1      5/29
Console> (enable)
```

Specifying the DHCP-Snooping IP Address-to-MAC Address Binding on a Per-Port Basis

To specify the IP address-to-MAC address binding for the specified port, perform this task in privileged mode:

	Task	Command
Step 1	Specify the IP address-to-MAC address binding for the specified port.	**set port dhcp-snooping** *mod/port* **add-binding** *ip-addr mac-addr* [*vlan*]
Step 2	Display the DHCP-snooping configuration.	**show port dhcp-snooping** [*mod*[*/ports*]]

This example shows how to specify the IP address-to-MAC address binding for the specified port:

```
Console> (enable) set port dhcp-snooping 5/29 add-binding 172.20.52.67 00-01-7b-9b-05-3f 1
DHCP Snooping Binding addition successful for Port 5/29, Vlan 1
 IP addr 172.20.52.67, Mac Addr 00-01-7b-9b-05-3f.
Console> (enable)
```

Displaying DHCP-Snooping Information

You can display the DHCP-snooping binding table and configuration information using the commands in this section.

Displaying the Binding Table

The DHCP-snooping binding table for each switch contains the binding entries that correspond to the untrusted ports. The table does not contain information about the hosts that are interconnected with a trusted port, because each interconnected switch has its own binding table.

To display DHCP-snooping binding table information, perform this task in privileged mode:

Task	Command
Display the DHCP-snooping binding table information.	**show dhcp-snooping bindings**

This example shows how to display the DHCP-snooping binding information for a switch:

```
Console# show dhcp-snooping bindings
MacAddress          IpAddress        Lease(sec)     VLAN  Port
------------------  ---------------  ----------     ----  -----

00-01-7b-9b-05-3f   192.168.80.221   86377          810   1/8
```

Table 33-2 describes the fields in the **show dhcp-snooping binding** command output.

Table 33-2 show dhcp-snooping bindings Command Output

Field	Description
MAC Address	Client-hardware MAC address.
IP Address	Client IP address assigned from the DHCP server.
Lease (seconds)	IP address lease time.
VLAN	VLAN number of the client port.
Port	Port that connects to the DHCP-client host.

Displaying the DHCP-Snooping Configuration and Statistics

To display DHCP-snooping configuration information for a switch, perform this task in privileged mode:

Task	Command
Display the DHCP-snooping configuration for a switch.	**show dhcp-snooping config**

This example shows how to display the DHCP-snooping host tracking and match-MAC configuration:

```
Console# show dhcp-snooping config
DHCP Snooping MAC address matching is enabled.
DHCP Snooping host-tracking information option is disabled.
Remote ID used in information option is 00-01-64-41-60-ff.
DHCP Snooping auto save is disabled.
DHCP Snooping bindings storage file is bootflash:dhcp-snooping-bindings-databas.
DHCP Snooping global bindings limit 16384.
DHCP Snooping available global bindings limit 16383.
Console> (enable)
```

To display the DHCP-snooping statistics for a switch, perform this task in privileged mode:

Task	Command
Display the DHCP-snooping statistics for a switch.	**show dhcp-snooping statistics**

This example shows how to display the DHCP-snooping statistics for a switch:

```
Console# show dhcp-snooping statistics
Packets forwarded                       =       125
Packets dropped                         =       3
Packets dropped from untrusted ports    =       0
Number of bindings entries              =       5
Console#
```

To display the DHCP-snooping port configuration for a switch, perform this task in privileged mode:

Task	Command
Display the DHCP-snooping port configuration for a switch.	**show port dhcp-snooping**

This example shows how to display the DHCP-snooping port configuration for a switch:

```
Console> (enable) show port dhcp-snooping
Port     Trust        Source-Guard    Source-Guarded IP Addresses
-----    -----------  ------------    ---------------------------
 5/1     untrusted    disabled

 5/2     trusted      disabled

 5/3     untrusted    disabled

 5/4     untrusted    disabled

 5/5     untrusted    disabled

 5/6     untrusted    disabled

 5/7     untrusted    disabled

 5/8     untrusted    disabled

 5/9     untrusted    disabled

 5/10    untrusted    disabled

 5/11    untrusted    disabled

 5/12    untrusted    disabled

 5/13    untrusted    disabled

 5/14    untrusted    disabled

 5/15    untrusted    disabled

 5/16    untrusted    disabled

 5/17    untrusted    disabled

 5/18    untrusted    disabled

 5/19    untrusted    disabled

 5/20    untrusted    disabled

 5/21    untrusted    disabled

 5/22    untrusted    disabled

 5/23    untrusted    disabled

 5/24    untrusted    disabled

Port     Binding Limit         No. of Existing Bindings
-----    ------------------    ------------------------
 5/1     32                    0
 5/2     32                    0
 5/3     32                    0
 5/4     32                    0
 5/5     32                    0
 5/6     32                    0
 5/7     32                    0
 5/8     32                    0
 5/9     32                    0
 5/10    32                    0
 5/11    32                    0
```

```
 5/12    32                    0
 5/13    32                    0
 5/14    32                    0
 5/15    32                    0
 5/16    32                    0
 5/17    32                    0
 5/18    32                    0
 5/19    32                    0
 5/20    32                    0
 5/21    32                    0
 5/22    32                    0
 5/23    32                    0
 5/24    32                    0
Console> (enable)
```

Storing DHCP-Snooping Binding Entries to a Flash Device

The DHCP-snooping binding entries can be stored to a flash device so the bindings can be restored immediately after the switch is reset.

The **auto-save** *interval* option is for configuring the auto-save interval for DHCP-snooping bindings. Valid ranges for the interval are 1 through 35000 minutes. Specifying a 0 disables the periodic saving of bindings on the flash device and deletes the bindings file stored in flash. Specifying a 0 does not clear a user-specified filename. The user-specified filename is cleared and returned to the default filename after the **clear config all** command is entered.

The *device:filename* option is for specifying the flash device and filename for storing the bindings. By default, the flash device is bootflash and the default filename is dhcp-snooping-bindings-database. If you have not configured a filename, the bindings are automatically saved with the default filename on the flash device.

To enable the **auto-save** option for DHCP-snooping binding entries and specify the interval to periodically save the bindings, perform this task in privileged mode:

Task	Command
Enable the **auto-save** option for DHCP-snooping binding entries and specify the interval to periodically save the bindings.	**set dhcp-snooping bindings-database auto-save** *interval*

This example shows how to enable the **auto-save** option for DHCP-snooping binding entries and specify an interval of 600 minutes to periodically save the bindings:

```
Console> (enable) set dhcp-snooping bindings-database auto-save 600
DHCP Snooping auto-save interval set to 600 minutes.
Console> (enable)
```

To specify the flash device and filename for storing the bindings, perform this task in privileged mode:

Task	Command
Specify the flash device and filename for storing the bindings.	**set dhcp-snooping bindings-database** *device:*[*filename*]

This example shows how to specify the flash device and filename for storing the bindings:

```
Console> (enable) set dhcp-snooping bindings-database disk1:dhcp-bindings
```

```
DHCP Snooping bindings storage file set to disk1:dhcp-bindings.
Console> (enable)
```

This example shows how to display the DHCP-snooping bindings-database configuration:

```
Console> (enable) show dhcp-snooping config
DHCP Snooping MAC address matching is enabled.
DHCP Snooping host-tracking information option is disabled.
Remote ID used in information option is 00-01-64-41-60-ff.
DHCP Snooping auto save interval is 600.
DHCP Snooping bindings storage file is disk1:dhcp-bindings.
Console> (enable)
```

Understanding How IP Source Guard Works

IP source guard prevents IP spoofing by allowing only the IP addresses that are obtained through DHCP snooping on a particular port. Initially, all IP traffic on the port is blocked except for the DHCP packets that are captured by DHCP snooping. When a client receives a valid IP address from the DHCP server, a port access control list (PACL) is installed on the port that permits the traffic from the IP address. This process restricts the client IP traffic to those source IP addresses that are obtained from the DHCP server; any IP traffic with a source IP address other than that in the PACL's permit list is filtered out. This filtering limits the ability of a host to attack the network by claiming a neighbor host's IP address.

Note If you enable IP source guard on a trunk port with a large number of VLANs that have DHCP snooping enabled, you might run out of the ACL hardware resources, and some clients that are connected to the ports may not be able to send the traffic. We do not recommend using this configuration because you are limited to ten IP addresses per port.

Note In software releases prior to software release 8.6(1), you are limited to ten IP addresses per port. In software release 8.6(1) and later releases, you can have up to 48 IP addresses per port.

IP source guard uses source IP address filtering, which filters the IP traffic that is based on its source IP address. Only the IP traffic with a source IP address that matches the IP source binding entry is permitted.

A port's IP source address filter is changed when a new DHCP-snooping binding entry for a port is created or deleted. The port PACL is modified and reapplied in the hardware to reflect the IP source binding change. By default, if you enable IP source guard without any DHCP-snooping bindings on the port, a default PACL that denies all IP traffic is installed on the port. When you disable IP source guard, any IP source filter PACL is removed from the port.

IP Source Guard Configuration Guidelines

This section describes the guidelines for configuring IP source guard in your network:

- IP source guard is supported on PFC 3 and later versions.
- In software releases prior to software release 8.6(1), you are limited to ten IP addresses per port. In software release 8.6(1) and later releases, you can have up to 48 IP addresses per port.
- IP source guard is not recommended on trunk ports.

- IP source guard cannot coexist with PACLs.
- IP source guard is not supported on EtherChannel-enabled ports, and EtherChannel is not supported on IP source guard-enabled ports.
- VLAN-based ACL features, such as static ARP inspection, are disabled when you enable IP source guard.
- We recommend that you enable high availability when using dynamic ARP inspection (DAI), DHCP snooping, and IP source guard. If high availability is not enabled, clients have to renew their IP addresses for these features to work after a switchover. For configuration details on DAI, see the "Dynamic ARP Inspection" section on page 15-39.

Enabling IP Source Guard on a Port

To enable IP source guard, perform this task in privileged mode:

	Task	Command
Step 1	Configure the port as port based.	**set port security-acl 3/1 port-based**
Step 2	Enable IP source guard.	**set port dhcp-snooping 3/1 source-guard enable**
Step 3	Enable DHCP snooping.	**set security acl ip dhcpsnoop permit dhcp-snooping**
Step 4	Allow the port to forward other traffic.	**set security acl ip dhcpsnoop permit ip any any**
Step 5	Save the ACL configuration.	**commit security acl dhcpsnoop**
Step 6	Enable the ACL on the VLAN.	**set security acl map dhcpsnoop 10**
Step 7	Enable DHCP-snooping trust on a port.	**set port dhcp-snooping 1/2 trust enable**

Note Before you can enable IP source guard, you must enable DHCP snooping on the VLAN to which the port belongs. You must configure the port as either port based or in merge mode for security ACLs. You should only enable IP source guard on DHCP-snooping untrusted ports.

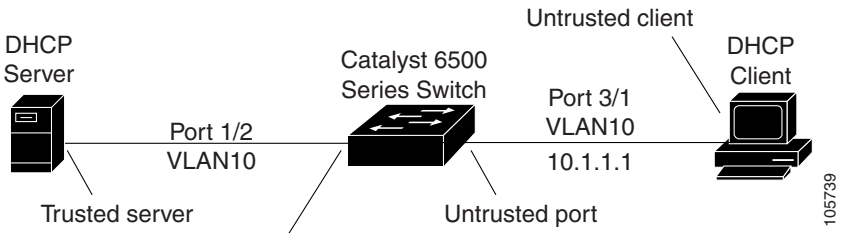
This example shows how to enable IP source guard:

```
Console> (enable) set port security-acl 3/1 port-based
Warning:Vlan-based ACL features will be disabled on ports 3/1.
ACL interface is set to port-based mode for port(s) 3/1.
Console> (enable) set port dhcp-snooping 3/1 source-guard enable
IP Source Guard enabled on port(s)  3/1.

Console> (enable) set port dhcp-snooping 1/2 trust enable
Port(s)  1/2 state set to trusted for DHCP Snooping.
Console> (enable) set security acl ip dhcpsnoop permit dhcp-snooping
Successfully configured DHCP Snooping for ACL dhcpsnoop. Use 'commit' command to
 save changes.

Console> (enable) set security acl ip dhcpsnoop permit ip any any
dhcpsnoop editbuffer modified. Use 'commit' command to apply changes.
Console> (enable) commit security acl dhcpsnoop
ACL commit in progress.

ACL 'dhcpsnoop' successfully committed.
Console> (enable) set security acl map dhcpsnoop 10
Mapping in progress.
```

```
ACL dhcp successfully mapped to port(s) 5/1.
Console>
```

Figure 33-3 shows the typical topology that is used when you configure IP source guard on an untrusted port.

Figure 33-3 IP Source Guard Enabled on an Untrusted Port



Displaying the IP Source Guard Information

You can display the information about IP source guard for all ports on a switch using the **show port dhcp-snooping** command. To display information about IP source guard on a module, perform this task in normal mode:

Task	Command
Display information about IP source guard on a port.	**show port dhcp-snooping 4**

This example shows how to display the configuration for IP source guard on a port:

```
Console> (enable) show port dhcp-snooping 3/25
Port     Trust     Source-Guard    Source-Guarded IP Addresses
-----  ----------  ------------    ---------------------------
 3/25  untrusted      enabled      192.168.80.6, 192.168.80.5,
                                   192.168.80.4, 192.168.80.3,
                                   192.168.80.2, 192.168.80.1

Console> (enable)
```

CHAPTER 34

Configuring NTP

This chapter describes how to configure the Network Time Protocol (NTP) on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Understanding How NTP Works

NTP synchronizes the timekeeping among a set of distributed time servers and clients. This synchronization allows the events to be correlated when the system logs are created and the other time-specific events occur.

An NTP server must be accessible by the client switch. NTP runs over User Datagram Protocol (UDP), which runs over IP. NTP is documented in RFC 1305. All NTP communication uses Coordinated Universal Time (UTC), which is the same as Greenwich Mean Time. An NTP network usually gets its time from an authoritative time source, such as a radio clock or an atomic clock that is attached to a time server. NTP distributes this time across the network. NTP is extremely efficient; no more than one packet per minute is necessary to synchronize two machines to within a millisecond of one another.

NTP uses a stratum to describe how many NTP hops away a machine is from an authoritative time source. A stratum 1 time server has a radio or atomic clock that is directly attached; a stratum 2 time server receives its time from a stratum 1 time server, and so on. A machine running NTP automatically chooses as its time source the machine with the lowest stratum number that it is configured to communicate with through NTP. This strategy effectively builds a self-organizing tree of NTP speakers.

NTP has two ways to avoid synchronizing to a machine whose time might be ambiguous:

- NTP never synchronizes to a machine that is not synchronized itself.
- NTP compares the time that is reported by several machines and does not synchronize to a machine that has its time significantly different from the others, even if its stratum is lower.

The communications between the machines running NTP, known as associations, are usually statically configured; each machine is given the IP address of all machines with which it should form the associations. Accurate timekeeping is possible by exchanging the NTP messages between each pair of machines with an association. However, in a LAN environment, you can configure NTP to use the IP broadcast messages. With this alternative, you can configure the machine to send or receive the broadcast messages, but the accuracy of timekeeping is marginally reduced because the information flow is one-way only.

Cisco's implementation of NTP does not support stratum 1 service; it is not possible to connect to a radio or atomic clock. We recommend that you obtain the time service for your network from the public NTP servers that are available in the IP Internet. If the network is isolated from the Internet, Cisco's NTP implementation allows a machine to be configured so that it acts as though it is synchronized using NTP, when in fact it has determined the time using other means. Other machines then synchronize to that machine using NTP.

A number of manufacturers include NTP software for their host systems, and a publicly available version for systems running UNIX and its various derivatives is also available. This software allows time-synchronized host systems.

NTP Default Configuration

Table 34-1 shows the default NTP configuration.

Table 34-1 NTP Default Configuration

Feature	Default Value
Broadcast client mode	Disabled
Client mode	Disabled
Broadcast delay	3000 microseconds
Time zone	Not specified
Offset from UTC	0 hours
Summertime adjustment	Disabled
NTP server	None specified
Authentication mode	Disabled

Configuring NTP on the Switch

These sections describe how to configure NTP:

Enabling NTP in Broadcast-Client Mode

Configure the switch in NTP broadcast-client mode if an NTP broadcast server, such as a router, regularly broadcasts time-of-day information on the network. To compensate for any server-to-client packet latency, you can specify an NTP broadcast delay (a time adjustment factor for the receiving of broadcast packets by the switch).

To enable NTP broadcast-client mode on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Enable NTP broadcast-client mode.	**set ntp broadcastclient enable**
Step 2	(Optional) Set the estimated NTP broadcast packet delay.	**set ntp broadcast delay** *microseconds*
Step 3	Verify the NTP configuration.	**show ntp** [**noalias**]

This example shows how to enable NTP broadcast-client mode on the switch, set a broadcast delay of 4000 microseconds, and verify the configuration:

```
Console> (enable) set ntp broadcastclient enable
NTP Broadcast Client mode enabled
Console> (enable) set ntp broadcastdelay 4000
NTP Broadcast delay set to 4000 microseconds
Console> (enable) show ntp

Current time: Tue Jun 23 1998, 20:25:43
Timezone: '', offset from UTC is 0 hours
Summertime: '', disabled
Last NTP update:
Broadcast client mode: enabled
Broadcast delay: 4000 microseconds
Client mode: disabled

NTP-Server
--------------------------------------
Console> (enable)
```

Configuring NTP in Client Mode

Configure the switch in NTP client mode if you want the client switch to regularly send time-of day requests to an NTP server. You can configure up to ten server addresses per client.

To configure the switch in NTP client mode, perform this task in privileged mode:

	Task	Command
Step 1	Configure the IP address of the NTP server.	**set ntp server** *ip_addr*
Step 2	Enable NTP client mode.	**set ntp client enable**
Step 3	Verify the NTP configuration.	**show ntp** [**noalias**]

This example shows how to configure the NTP server address, enable NTP client mode on the switch, and verify the configuration:

```
Console> (enable) set ntp server 172.20.52.65
NTP server 172.20.52.65 added.
Console> (enable) set ntp client enable
NTP Client mode enabled
Console> (enable) show ntp

Current time: Tue Jun 23 1998, 20:29:25
Timezone: '', offset from UTC is 0 hours
Summertime: '', disabled
Last NTP update: Tue Jun 23 1998, 20:29:07
Broadcast client mode: disabled
Broadcast delay: 3000 microseconds
Client mode: enabled

NTP-Server
---------------------------------------
172.16.52.65
Console> (enable)
```

Configuring Authentication in Client Mode

Authentication can enhance the security of a system running NTP. When you enable authentication, the client switch sends the time-of-day requests to the trusted NTP servers only. Authentication is documented in RFC 1305.

You can configure up to ten authentication keys per client. Each authentication key is actually a pair of two keys:

- A public key number—A 32-bit integer that can range from 1 to 4294967295
- A secret key string—An arbitrary string of 32 characters including all printable characters and spaces

To authenticate the message, the client authentication key must match that of the server. The authentication key must be securely distributed in advance (the client administrator must get the key pair from the server administrator and configure it on the client).

To configure authentication, perform this task in privileged mode:

	Task	Command
Step 1	Configure an authentication key pair for NTP and specify whether the key is trusted or untrusted.	**set ntp key** *public_key* [*trusted* | *untrusted*] **md5** *secret_key*
Step 2	Specify the IP address of the NTP server and the public key.	**set ntp server** *ip_addr* [key *public_key*]
Step 3	Enable NTP client mode.	**set ntp client enable**
Step 4	Enable NTP authentication.	**set ntp authentication enable**
Step 5	Verify the NTP configuration.	**show ntp** [**noalias**]

This example shows how to configure the NTP server address, enable NTP client and authentication modes on the switch, and verify the configuration:

```
Console> (enable) set ntp server 172.20.52.65 key 879
NTP server 172.20.52.65 with key 879 added.
Console> (enable) set ntp client enable
NTP Client mode enabled
Console> (enable) set ntp authentication enable
NTP authentication feature enabled
Console> (enable) show ntp

Current time: Tue Jun 23 1998, 20:29:25
Timezone: '', offset from UTC is 0 hours
Summertime: '', disabled
Last NTP update: Tue Jun 23 1998, 20:29:07
Broadcast client mode: disabled
Broadcast delay: 3000 microseconds
Client mode: enabled
Authentication: enabled

NTP-Server                                Server Key
----------------------------------------- ----------
172.16.52.65

Key Number  Mode       Key String
----------  ---------  --------------------------------

Console> (enable)
```

Setting the Time Zone

You can specify a time zone for the switch to display the time in that time zone. You must enable NTP before you set the time zone. If NTP is not enabled, this command has no effect. If you enable NTP and do not specify a time zone, UTC is shown by default.

To set the time zone, perform this task in privileged mode:

	Task	Command
Step 1	Set the time zone.	**set timezone** *zone hours* [*minutes*]
Step 2	Verify the time zone configuration.	**show timezone**

This example shows how to set the time zone on the switch:

```
Console> (enable) set timezone Pacific -8
Timezone set to 'Pacific', offset from UTC is -8 hours
Console> (enable)
```

Enabling the Daylight Saving Time Adjustment

Following the U.S. standards, you can have the switch advance the clock one hour at 2:00 a.m. on the first Sunday in April and move back the clock one hour at 2:00 a.m. on the last Sunday in October. You can also explicitly specify the start and end dates and times and whether or not the time adjustment recurs every year.

To enable the daylight saving time adjustment following the U.S. standards, perform this task in privileged mode:

	Task	Command
Step 1	Enable the daylight saving time adjustment.	**set summertime enable** [*zone_name*] **set summertime recurring**
Step 2	Verify the configuration.	**show summertime**

This example shows how to set the clock that is adjusted for Pacific Daylight Time following the U.S. standards:

```
Console> (enable) set summertime enable PDT
Console> (enable) set summertime recurring
Summertime is enabled and set to 'PDT'
Console> (enable)
```

To enable the daylight saving time adjustment that recurs every year on different days or with a different offset than the U.S. standards, perform this task in privileged mode:

	Task	Command
Step 1	Enable the daylight saving time adjustment.	**set summertime recurring** *week day month hh:mm week day month hh:mm offset*
Step 2	Verify the configuration.	**show summertime**

This example shows how to set the daylight saving time adjustment, repeating every year, starting on the third Monday of February at noon and ending on the second Saturday of August at 3:00 p.m. with a 30-minute offset forward in February and back in August.

```
Console> (enable) set summertime recurring 3 mon feb 3:00 2 saturday aug 15:00 30
Summer time is disabled and set to ''
    start: Sun Feb 13 2000, 03:00:00
    end: Sat Aug 26 2000, 14:00:00
    Offset: 30 minutes
    Recurring: yes, starting at 3:00am Sunday of the third week of February and ending
    14:00pm Saturday of the fourth week of August.
Console> (enable)
```

To enable the daylight saving time adjustment to a nonrecurring specific date, perform this task in privileged mode:

	Task	Command
Step 1	Enable the daylight saving time adjustment.	**set summertime date** *month date year hh:mm month date year hh:mm offset*
Step 2	Verify the configuration.	**show summertime**

This example shows how to set the nonrecurring daylight saving time adjustment on April 30, 1999 at 11:32, ending on February 1, 2003 at 12:02 a.m., with an offset of 50 minutes:

```
Console> (enable) set summertime date apr 13 2000 4:30 jan 21 2002 5:30 1440
Summertime is disabled and set to ''
Start : Thu Apr 13 2000, 04:30:00
End   : Mon Jan 21 2002, 05:30:00
Offset: 1440 minutes (1 day)
Recurring: no
Console> (enable)
```

Disabling the Daylight Saving Time Adjustment

To disable the daylight saving time adjustment, perform this task in privileged mode:

	Task	Command
Step 1	Disable the daylight saving time adjustment.	**set summertime disable** [*zone_name*]
Step 2	Verify the configuration.	**show summertime**

This example shows how to disable the daylight saving time adjustment:

```
Console> (enable) set summertime disable Arizona
Summertime is disabled and set to 'Arizona'
Console> (enable)
```

Clearing the Time Zone

To clear the time zone settings and return the time zone to Coordinated Universal Time (UTC), perform this task in privileged mode:

Task	Command
Clear the time zone settings.	**clear timezone**

This example shows how to clear the time zone settings adjustment:

```
Console> (enable) clear timezone
Timezone name and offset cleared
Console> (enable)
```

Clearing NTP Servers

To clear an NTP server address from the NTP servers table on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Specify the NTP server to clear.	**clear ntp server** [*ip_addr* | **all**]
Step 2	Verify the NTP configuration.	**show ntp** [**noalias**]

This example shows how to clear an NTP server address from the NTP server table:

```
Console> (enable) clear ntp server 172.16.64.10
NTP server 172.16.64.10 removed.
Console> (enable)
```

Disabling NTP

To disable NTP broadcast-client mode on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Disable NTP broadcast-client mode.	**set ntp broadcastclient disable**
Step 2	Verify the NTP configuration.	**show ntp** [**noalias**]

This example shows how to disable NTP broadcast-client mode on the switch:

```
Console> (enable) set ntp broadcastclient disable
NTP Broadcast Client mode disabled
Console> (enable)
```

To disable NTP client mode on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Disable NTP client mode.	**set ntp client disable**
Step 2	Verify the NTP configuration.	**show ntp** [**noalias**]

This example shows how to disable NTP client mode on the switch:

```
Console> (enable) set ntp client disable
NTP Client mode disabled
Console> (enable)
```

CHAPTER 35

Configuring Broadcast Suppression

This chapter describes how to configure broadcast suppression on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Understanding How Broadcast Suppression Works

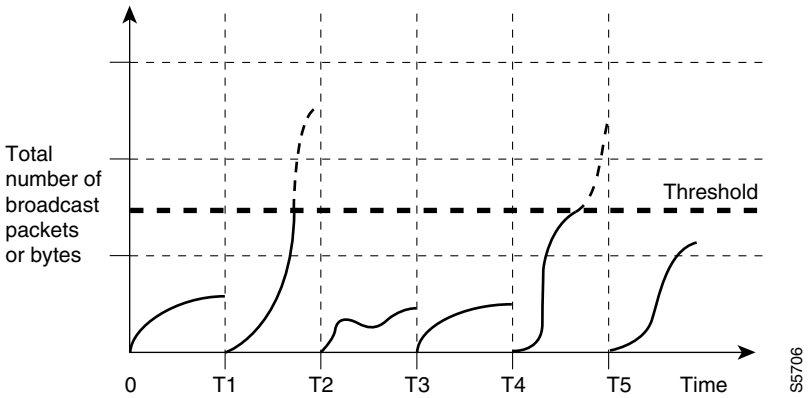
Note Broadcast and multicast suppression is not supported on the WS-X6148A-GE-TX, WS-X6148A-GE-45A, and WS-X6548-GE-TX modules.

Broadcast suppression prevents the switched ports on a LAN from being disrupted by a broadcast storm on one of the ports. A LAN broadcast storm occurs when the broadcast or multicast packets flood the LAN, creating excessive traffic and degrading the network performance. Errors in the protocol-stack implementation or in the network configuration can cause a broadcast storm.

Broadcast suppression uses filtering that measures the broadcast activity on a LAN over a time period (15264 nsec to ~1 sec) that varies based on the type of line card and speed setting on the port, and compares the measurement with a predefined threshold. If the threshold is reached, further broadcast activity is suppressed for the duration of a specified time period. Broadcast suppression is disabled by default.

Figure 35-1 shows the broadcast traffic patterns on a port over a given period of time. In this example, broadcast suppression occurs between the time intervals T1 and T2 and between T4 and T5. During those time periods, the amount of broadcast traffic exceeded the configured threshold.

Figure 35-1 Broadcast Suppression



The broadcast suppression threshold numbers and the time interval make the broadcast suppression algorithm work with different levels of granularity. A higher threshold allows more broadcast packets to pass through.

Broadcast suppression on the Catalyst 6500 series switches is implemented in the hardware. The suppression circuitry monitors the packets passing from a port to the switching bus. Using the Individual/Group bit in the packet destination address, the broadcast suppression circuitry determines if the packet is unicast or broadcast, keeps track of the current count of broadcasts within the time interval, and when a threshold is reached, filters out the subsequent broadcast packets.

Because hardware broadcast suppression uses a bandwidth-based method to measure the broadcast activity, the most significant implementation factor is setting the percentage of the total available bandwidth that can be used by the broadcast traffic. A threshold value of 100 percent means that no limit is placed on the broadcast traffic. By entering the **set port broadcast** command, you can set up the broadcast suppression threshold value.

Because the packets do not arrive at uniform intervals, the time interval during which the broadcast activity is measured can affect the behavior of broadcast suppression.

On the Gigabit Ethernet ports, you can use the broadcast suppression to filter the multicast and unicast traffic. You can suppress the multicast or unicast traffic separately on a port; both require that you configure broadcast suppression. When you specify a percentage of the total bandwidth to be used for the multicast or unicast traffic, the same limit applies to the broadcast traffic.

Note When broadcast, multicast, or unicast suppression occurs, you can configure the ports to go into the *errdisable* state. See the "Enabling the errdisable State" section on page 35-5 for details.

Note Multicast suppression does not drop the bridge protocol data unit (BPDU) packets.

Note The reception of BPDUs is not guaranteed when multicast suppression is enabled on the following modules: WS-X6724-SFP, WS-X6748-GE-TX, WS-X6748-SFP, WS-X6704-10GE, WS-SUP32-GE-3B, and WS-SUP32-10GE-3B. Enabling multicast suppression on these modules can cause BPDUs to be suppressed when the multicast suppression threshold is exceeded. We strongly advise that you do not use multicast suppression on ports that need to receive BPDUs because potential side effects can be root port loss or spanning tree loops when the suppression threshold is exceeded.

Configuring Broadcast Suppression on the Switch

These sections describe how to configure broadcast suppression on the Catalyst 6500 series switches:

Note The switch supports multicast and unicast traffic storm control on Gigabit and 10 Gigabit Ethernet LAN ports. Most FastEthernet switching modules do not support multicast and unicast traffic storm control, with the exception of WS-X6148A-RJ-45 and the WS-X6148-SFP.

Enabling Broadcast Suppression

To enable broadcast suppression for one or more ports, perform this task in privileged mode:

	Task	Command
Step 1	Enable the broadcast suppression threshold for one or more ports as a percentage of the total bandwidth.	**set port broadcast** *mod/port threshold%* [**violation** {**drop-packets** | **errdisable**}] [**multicast** {**enable** | **disable**}] [**unicast** {**enable** | **disable**}]
Step 2	Verify the broadcast suppression configuration.	**show port broadcast** [*mod*[/*port*]]

Note Although you can specify the broadcast suppression threshold to 0.01 percent, not all modules adjust to that level of precision. Most thresholds vary between 0.01 percent and 0.05 percent. If you specify a finer threshold, the threshold percent adjusts as closely as possible.

Note On these modules, a level value of 0.33 percent or less suppresses all traffic:
—WS-X6704-10GE
—WS-X6748-SFP
—WS-X6724-SFP
—WS-X6748-GE-TX

This example shows how to enable bandwidth-based broadcast suppression and verify the configuration:

```
Console> (enable) set port broadcast 3/1-6 75.25%
Ports 3/1-6 broadcast traffic limited to 75.25%.
On broadcast suppression ports 3/1-6 are configured to drop-packets.
Console> (enable) show port broadcast 3

Port     Broadcast-Limit Multicast Unicast Total-Drop           Action
-------- --------------- --------- ------- -------------------- ------------
 3/1             75.25 %         -       -                    0 drop-packets
 3/2             75.25 %         -       -                    0 drop-packets
 3/3             75.25 %         -       -                    2 drop-packets
 3/4             75.25 %         -       -                    0 drop-packets
 3/5             75.25 %         -       -                    0 drop-packets
```

```
 3/6             75.25 %         -       -                    0 drop-packets
 3/7                   -         -       -                    0 drop-packets
.
.<snip>
.
Console> (enable)
```

This example shows how to limit the multicast and broadcast traffic to 80 percent for port 1 on module 2 and verify the configuration:

```
Console> (enable) set port broadcast 2/1 80% multicast enable
Port 2/1 broadcast and multicast traffic limited to 80.00%.
On broadcast suppression port 2/1 is configured to drop-packets.
Console> (enable) show port broadcast 2/1

Port     Broadcast-Limit Multicast Unicast Total-Drop           Action
-------- --------------- --------- ------- -------------------- ------------
 2/1             80.00 %   80.00 %       -                    0 drop-packets
Console> (enable)
```

Disabling Broadcast Suppression

To disable broadcast suppression on one or more ports, perform this task in privileged mode:

Task	Command
Disable broadcast suppression on one or more ports.	**clear port broadcast** *mod/port*

This example shows how to disable broadcast suppression on one or more ports:

```
Console> (enable) clear port broadcast 2/1
Port 2/1 traffic unlimited.
Console> (enable)
```

Enabling the errdisable State

Note A port is in the errdisable state if it is enabled in NVRAM but is disabled at runtime by any process. For example, if UniDirectional Link Detection (UDLD) detects a unidirectional link, the port shuts down at runtime. However, because the NVRAM configuration for the port is enabled (you have not disabled the port), the port status is shown as errdisable.

When broadcast, multicast, or unicast suppression occurs, you can configure the ports to either drop the packets or go into the errdisable state. The errdisable state feature can be enabled or disabled on a per-port basis and is disabled by default (the **drop-packets** option is enabled by default).

Note When broadcast, multicast, or unicast suppression occurs and a port is configured for errdisable, there is a delay before the port stops dropping the packets and goes to the errdisable state. The delay period varies; the exact amount of delay can vary from switch to switch.

To enable the errdisable state on a port, perform this task in privileged mode:

	Task	Command
Step 1	Enable the errdisable state.	**set port broadcast** *mod/port threshold%* [**violation** {**drop-packets** | **errdisable**}] [**multicast** {**enable** | **disable**}] [**unicast** {**enable** | **disable**}]
Step 2	Verify that the errdisable state is enabled.	**show port broadcast** [*mod*[/*port*]]

This example shows how to limit the broadcast traffic to 90 percent and error disable the port when broadcast suppression occurs:

```
Console> (enable) set port broadcast 4/6 90% violation errdisable
Port 4/6 broadcast traffic limited to 90.00%.
On broadcast suppression port 4/6 is configured to move to errdisabled state.
Console> (enable)
```

Note Enter the **set errdisable-timeout enable bcast-suppression** command to enable the errdisable timeout feature for broadcast suppression.

Once a port is put into errdisable state, it can be reenabled after a specific timeout interval has expired. Enter the **set errdisable-timeout interval** command to specify the timeout interval.

Enter the **set port errdisable-timeout** command to control on a per-port basis whether a port should be enabled after a certain time or continue to be in the errdisabled state once it has been errdisabled.

For more information, see the "Configuring a Timeout Period for Ports in errdisable State" section on page 4-12.

CHAPTER 36

Configuring Layer 3 Protocol Filtering

This chapter describes how to configure Layer 3 protocol filtering on Ethernet, Fast Ethernet, and Gigabit Ethernet ports on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Understanding How Layer 3 Protocol Filtering Works

Layer 3 protocol filtering prevents certain protocol traffic from being forwarded out the switch ports. Layer 3 protocol filtering is implemented on the supervisor engine and does not require a Policy Feature Card (PFC) or Multilayer Switch Feature Card (MSFC). The broadcast and unicast flood traffic is filtered based on the membership of the ports in the different protocol groups. This filtering is in addition to the filtering that is provided by the port-VLAN membership. Layer 3 protocol filtering is supported only on the nontrunking Ethernet, Fast Ethernet, and Gigabit Ethernet ports.

The trunking ports are always members of all protocol groups. To avoid compatibility issues with the other networking devices, Layer 3 protocol filtering is not performed on the trunk ports. Layer 2 protocols, such as Spanning Tree Protocol (STP) and Cisco Discovery Protocol (CDP), are not affected by Layer 3 protocol filtering. The dynamic ports and ports that have port security enabled are members of all protocol groups.

You can configure a port with any one of these modes for each protocol group: **on**, **off**, or **auto**.

If the configuration is set to **on**, the port receives all the flood traffic for that protocol. If the configuration is set to **off**, the port does not receive any flood traffic for that protocol.

If the configuration is set to **auto**, the port is added to the group only after the packets of the specific protocol are received on that port. With autolearning, the ports become members of the protocol group only after receiving the packets of the corresponding protocol from the device that is attached to that port. The autoconfigured ports are removed from the protocol group if no packets are received for that protocol within 60 minutes. The ports are also removed from the protocol group when the supervisor engine detects that the link is down on the port.

For example, if a host that supports both IP and Internetwork Packet Exchange (IPX) is connected to a switch port that is configured as **auto** for IPX, but the host is transmitting only the IP traffic, the port to which the host is connected does not forward any IPX flood traffic to the host. However, if the host sends an IPX packet, the supervisor engine software detects the protocol traffic and the port is added to the IPX group, allowing the port to receive the IPX flood traffic. If the host stops sending the IPX traffic for more than 60 minutes, the port is removed from the IPX protocol group.

By default, the ports are configured to **on** for the IP protocol group. Typically, you should configure a port to **auto** for IP only if there is a directly connected end station out the port. The default port configuration for IPX and Group is **auto**.

With Layer 3 protocol filtering enabled, the ports are identified on a protocol basis. A port can be a member of one or more protocol groups. The flood traffic for each protocol group is forwarded out a port only if that port belongs to the appropriate protocol group.

The packets are classified into these protocol groups:

- IP
- IPX
- AppleTalk, DECnet, and Banyan VINES (**group** mode)
- Packets not belonging to any of these protocols

Default Layer 3 Protocol Filtering Configuration

Table 36-1 shows the default Layer 3 protocol filtering configuration.

Table 36-1 Layer 3 Protocol Filtering Default Configuration

Feature	Default Value
Layer 3 protocol filtering	Disabled
ip mode	**on**
ipx mode	**auto**
group mode	**auto**

Configuring Layer 3 Protocol Filtering on the Switch

These sections describe how to configure Layer 3 protocol filtering on the Ethernet-type VLANs and on any type of Ethernet port:

- Enabling Layer 3 Protocol Filtering, page 36-3
- Disabling Layer 3 Protocol Filtering, page 36-3

Enabling Layer 3 Protocol Filtering

Note Protocol filtering is supported only on the Ethernet VLANs and on the nontrunking EtherChannel ports. The **set protocolfilter** command is not supported on the Network Analysis Module (NAM), the Supervisor Engine 720, or the Supervisor Engine 32.

To enable Layer 3 protocol filtering on the Ethernet ports, perform this task in privileged mode:

	Task	Command
Step 1	Enable Layer 3 protocol filtering on the switch.	**set protocolfilter enable**
Step 2	Set the protocol membership of the desired ports.	**set port protocol** *mod/port* {**ip** | **ipx** | **group**} {**on** | **off** | **auto**}
Step 3	Verify the port filtering configuration.	**show port protocol** [*mod*[*/port*]]

This example shows how to enable Layer 3 protocol filtering, set the protocol membership of the ports, and verify the configuration:

```
Console> (enable) set protocolfilter enable
Protocol filtering enabled on this switch.
Console> (enable) set port protocol 7/1-4 ip on
IP protocol set to on mode on ports 7/1-4.
Console> (enable) set port protocol 7/1-4 ipx off
IPX protocol disabled on ports 7/1-4.
Console> (enable) set port protocol 7/1-4 group auto
Group protocol set to auto mode on ports 7/1-4.
Console> (enable) show port protocol 7/1-4
Port     Vlan       IP       IP Hosts IPX      IPX Hosts Group    Group Hosts
-------- ---------- -------- -------- -------- --------- -------- -----------
7/1      4          on       1        off      0         auto-off 0
7/2      5          on       1        off      0         auto-on  1
7/3      2          on       1        off      0         auto-off 0
7/4      4          on       1        off      0         auto-on  1
Console> (enable)
```

Disabling Layer 3 Protocol Filtering

To disable Layer 3 protocol filtering, perform this task in privileged mode:

Task	Command
Disable Layer 3 protocol filtering on the switch.	**set protocolfilter disable**

This example shows how to disable Layer 3 protocol filtering:

```
Console> (enable) set protocolfilter disable

Protocol filtering disabled on this switch.
Console> (enable)
```

CHAPTER 37

Configuring the IP Permit List

This chapter describes how to configure the IP permit list on the Catalyst 6500 series switches.

Note The functionality of the IP permit list can also be achieved with the VLAN access control lists (VACLs). Because the VACLs are handled by the hardware (Policy Feature Card [PFC]), the VACL processing is faster than the IP permit list processing.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Understanding How the IP Permit List Works

The IP permit list prevents the inbound Telnet and SNMP access to the switch from the unauthorized source IP addresses. All other TCP/IP services (such as IP traceroute and IP ping) continue to work normally when you enable the IP permit list. The outbound Telnet, TFTP, and other IP-based services are unaffected by the IP permit list.

The Telnet attempts from the unauthorized source IP addresses are denied a connection. When the SNMP requests from the unauthorized IP addresses receive no response; the request times out. If you want to log the unauthorized access attempts to the console or a syslog server, you must change the logging severity level for IP, as described in the "Enabling the IP Permit List" section on page 37-3. If you want to generate the SNMP traps when the unauthorized access attempts are made, you must enable the IP permit list (ippermit) SNMP traps, as described in the "Enabling the IP Permit List" section on page 37-3. Multiple access attempts from the same unauthorized host only trigger notifications every 10 minutes.

You can configure up to 100 entries in the permit list. Each entry consists of an IP address and subnet mask pair in dotted decimal format and information on whether the IP address is part of the SNMP permit list, Telnet permit list, or both lists. The bits that are set to one in the mask are checked for a match with the source IP address of the incoming packets, while the bits that are set to zero are not checked. This process allows you to specify a wildcard address.

If you do not specify the mask for an IP permit list entry, or if you enter a host name instead of an IP address, the mask has an implicit value of all bits that are set to one (255.255.255.255 or 0xffffffff), which matches only the IP address of that host.

If you do not specify SNMP or Telnet for the type of permit list for the IP address, the IP address is added to both the SNMP and Telnet permit lists.

You can specify the same IP address in more than one entry in the permit list if the masks are different. The mask is applied to the address before it is stored in NVRAM, so that the entries that have the same effect but different addresses are not stored. When you add such an address to the IP permit list, the system displays the address after the mask is applied.

IP Permit List Default Configuration

Table 37-1 shows the default IP permit list configuration.

Table 37-1 IP Permit List Default Configuration

Feature	Default Value
IP permit list enable state	Disabled
Permit list entries	None configured
IP syslog message severity level	2
SNMP IP permit trap (ippermit)	Disabled

Configuring the IP Permit List on the Switch

These sections describe how to configure the IP permit list:

- Adding IP Addresses to the IP Permit List, page 37-2
- Enabling the IP Permit List, page 37-3
- Disabling the IP Permit List, page 37-4
- Clearing an IP Permit List Entry, page 37-5

Adding IP Addresses to the IP Permit List

You can add an IP address to the SNMP permit list, the Telnet permit list, or both lists.

To add IP addresses to the IP permit list, perform this task in privileged mode:

	Task	Command
Step 1	Specify the IP addresses to add to the IP permit list.	**set ip permit** *ip_address* [*mask*] [**telnet** | **snmp** | **ssh**]
Step 2	Verify the IP permit list configuration.	**show ip permit**

This example shows how to add the IP addresses to the IP permit list and verify the configuration:

```
Console> (enable) set ip permit 172.16.0.0 255.255.0.0 telnet
172.16.0.0 with mask 255.255.0.0 added to telnet permit list.
Console> (enable) set ip permit 172.20.52.32 255.255.255.224 snmp
172.20.52.32 with mask 255.255.255.224 added to snmp permit list.
Console> (enable) set ip permit 172.20.52.3 all
172.20.52.3 added to IP permit list.
Console> (enable) show ip permit
Telnet permit list feature enabled.
Snmp permit list feature enabled.
Permit List        Mask                Access Type
----------------   -----------------   -------------
172.16.0.0         255.255.0.0         telnet
172.20.52.3                            snmp telnet
172.20.52.32       255.255.255.224     snmp
Denied IP Address   Last Accessed Time Type    Telnet Count   SNMP Count
-----------------   ------------------ ------  ------------   ----------
172.100.101.104     01/20/97,07:45:20  SNMP              14         1430
172.187.206.222     01/21/97,14:23:05  Telnet             7          236

Console> (enable)
```

Note An IP not included in the Permit list is denied inbound Telnet and SNMP access to the switch.

Enabling the IP Permit List

You can enable either the SNMP permit list, the Telnet permit list, or both lists. If you do not specify a permit list, both the SNMP and Telnet permit lists are enabled.

Caution Before enabling the IP permit list, make sure that you add the IP address of your workstation or network management system to the permit list, especially when configuring through SNMP. Failure to do so could result in your connection being dropped by the switch that you are configuring. We recommend that you disable the IP permit list before clearing the IP permit entries or host addresses.

Note Enabling the IP permit list for Telnet without having an IP address in the permit list disables the Telnet access to the switch; however, the Telnet process will continue to run on the switch becasue Telnet cannot be disabled on CatOS.

To enable the IP permit list on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Enable the IP permit list.	**set ip permit enable** [**telnet** | **snmp** | **ssh**]
Step 2	If desired, enable the IP permit trap to generate the traps for the unauthorized access attempts.	**set snmp trap enable ippermit**
Step 3	If desired, configure the logging level to see the syslog messages for the unauthorized access attempts.	**set logging level ip 4 default**
Step 4	Verify the IP permit list configuration.	**show ip permit** **show snmp**

This example shows how to enable the IP permit list and verify the configuration:

```
Console> (enable) set ip permit enable
IP permit list enabled.
Console> (enable) set snmp trap enable ippermit
SNMP IP Permit traps enabled.
Console> (enable) set logging level ip 4 default
System logging facility <ip> set to severity 4(warnings)
Console> (enable) show ip permit
Telnet permit list feature enabled.
Snmp permit list feature disabled.

Permit List          Mask              Access-Type
----------------     ---------------   ---------------
172.16.0.0         255.255.0.0         telnet
172.20.52.3                            snmp telnet
172.20.52.32       255.255.255.224     snmp

Denied IP Address   Last Accessed Time Type    Telnet Count   SNMP Count
-----------------   ------------------ ------  ------------   ----------
172.100.101.104     01/20/97,07:45:20  SNMP              14         1430
172.187.206.222     01/21/97,14:23:05  Telnet             7          236

Console> (enable) show snmp
RMON:                         Disabled
Extended Rmon:                Extended RMON module is not present
Traps Enabled:
ippermit
Port Traps Enabled: None

Community-Access     Community-String
----------------     --------------------
read-only            public
read-write           private
read-write-all       secret

Trap-Rec-Address                            Trap-Rec-Community
------------------------------------------  --------------------
Console> (enable)
```

Disabling the IP Permit List

To disable the IP permit list on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Disable the IP permit list on the switch.	**set ip permit disable** [**telnet** | **snmp** | **ssh**]
Step 2	Verify the IP permit list configuration.	**show ip permit**

This example shows how to disable the IP permit list:

```
Console> (enable) set ip permit disable
IP permit list disabled.
Console> (enable)
```

Clearing an IP Permit List Entry

You can clear an IP address from the SNMP permit list, the Telnet permit list, or both lists. If you do not specify which permit list to clear the IP address from, the IP address is deleted from both permit lists.

Caution Disable the IP permit list before you clear the IP permit entries or the host addresses to prevent your connection from being dropped by the switch that you are configuring (in case you clear your current IP address).

Note Enabling the IP permit list for Telnet without having any IP addresses in the permit list will disable the Telnet access to the switch, but the Telnet process will be still running on the switch.

To clear an IP permit list entry, perform this task in privileged mode:

	Task	Command
Step 1	Disable the IP permit list.	**set ip permit disable** [**telnet** | **snmp** | **ssh**]
Step 2	Specify the IP address to remove from the IP permit list.	**clear ip permit** {*ip_address* [*mask*] | **all**} [**telnet** | **snmp** | **ssh**]
Step 3	Verify the IP permit list configuration.	**show ip permit**

This example shows how to clear an IP permit list entry:

```
Console> (enable) set ip permit disable all
Console> (enable) clear ip permit 172.100.101.102
172.100.101.102 cleared from IP permit list.
Console> (enable) clear ip permit 172.160.161.0 255.255.192.0 snmp
172.160.128.0 with mask 255.255.192.0 cleared from snmp permit list.
Console> (enable) clear ip permit 172.100.101.102 telnet
172.100.101.102 cleared from telnet permit list.
Console> (enable) clear ip permit all
IP permit list cleared.
Console> (enable)
```

CHAPTER 38

Configuring Port Security

This chapter describes how to configure port security and how to limit the number of MAC addresses that are learned on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

Note For information on configuring MAC address authentication bypass, see Chapter 41, "Configuring MAC Authentication Bypass."

Note For information on configuring 802.1X authentication to restrict the unauthorized devices from connecting to a LAN through the publicly accessible ports, see Chapter 40, "Configuring 802.1X Authentication."

Note For information on configuring authentication, authorization, and accounting (AAA) to monitor and control access to the command-line interface (CLI) on the Catalyst 6500 series switches, see Chapter 39, "Configuring the Switch Access Using AAA."

Note For information on configuring network admission control, see Chapter 44, "Configuring Network Admission Control."

This chapter consists of these sections:

Understanding How Port Security Works

You can use port security to block input to an Ethernet, Fast Ethernet, or Gigabit Ethernet port when the MAC address of the station attempting to access the port is different from any of the MAC addresses that are specified for that port. Alternatively, you can use port security to filter the traffic that is destined to or received from a specific host that is based on the host MAC address.

These sections describe the traffic filtering methods:

Allowing the Traffic Based on the Host MAC Address

The total number of MAC addresses that you can specify per port is limited as follows:

- In software releases prior to 8.1(1), the total number of MAC addresses that you can specify per port is limited to the global resource of 1024 plus 1 default MAC address. The total number of MAC addresses on any port cannot exceed 1025.
- In software release 8.1(1) and later releases, the total number of MAC addresses that you can specify per port is limited to the global resource of 4096 plus 1 default MAC address. The total number of MAC addresses on any port cannot exceed 4097.

Whether you allocate the maximum number of MAC addresses for each port depends on your network configuration. These combinations are examples of the valid allocations for the software releases prior to 8.1(1); the logic is the same for software release 8.1(1) and later releases:

- 1025 (1 + 1024) addresses on 1 port and 1 address each on the rest of the ports.
- 513 (1 + 512) each on 2 ports in a system and 1 address each on the rest of the ports.
- 901 (1 + 900) on 1 port, 101 (1 + 100) on another port, 25 (1 + 24) on the third port, and 1 address each on the rest of the ports.

After you allocate the maximum number of MAC addresses on a port, you can either specify the secure MAC address for the port manually or you can have the port dynamically configure the MAC address of the connected devices. Out of an allocated number of maximum MAC addresses on a port, you can manually configure all, allow all to be learned dynamically, or configure some manually and allow the rest to be learned dynamically. Once you manually configure or autoconfigure the addresses, the addresses are stored in nonvolatile RAM (NVRAM) and maintained after a reset. The addresses that have been learned dynamically are not saved, so after a reset of the switch, all dynamically learned addresses are cleared.

After you allocate a maximum number of MAC addresses on a port, you can specify how long the addresses on the port will remain secure. After the age time expires, the MAC addresses on the port become insecure. By default, all addresses on a port are secured permanently.

If a security violation occurs, you can configure the port to go into shutdown mode or restrictive mode. The shutdown mode allows you to specify whether the port is to be permanently disabled or disabled for only a specified time. The default is for the port to shut down permanently. The restrictive mode allows you to configure the port to remain enabled during a security violation and drop only the packets that are coming in from the insecure hosts.

Note If you configure a secure port in restrictive mode, and a station is connected to the port whose MAC address is already configured as a secure MAC address on another port on the switch, the port in restrictive mode shuts down instead of restricting the traffic from that station. For example, if you configure MAC-1 as the secure MAC address on port 2/1 and MAC-2 as the secure MAC address on port 2/2 and then connect the station with MAC-1 to port 2/2 when port 2/2 is configured for restrictive mode, port 2/2 shuts down instead of restricting the traffic from MAC-1.

When a secure port receives a packet, the source MAC address of the packet is compared to the list of secure source addresses that were manually configured or learned dynamically on the port. If a MAC address of a device that is attached to the port differs from the list of secure addresses, the port either shuts down permanently (default mode), shuts down for the time that you have specified, or drops the incoming packets from the insecure host. The port's behavior depends on how you configure it to respond to a security violation.

If a security violation occurs, the LED labeled "Link" for that port turns orange, and a link-down trap is sent to the Simple Network Management Protocol (SNMP) manager. An SNMP trap is not sent if you configure the port for restrictive violation mode. A trap is sent only if you configure the port to shut down during a security violation.

Restricting the Traffic Based on the Host MAC Address

You can filter the traffic that is based on a host MAC address so that the packets that are tagged with a specific source MAC address are discarded. When you specify a MAC address filter with the **set cam filter** command, the incoming traffic from that host MAC address is dropped and the packets that are addressed to that host are not forwarded.

Note The **set cam filter** command allows filtering for the unicast addresses only. You cannot filter the traffic for the multicast addresses with this command.

Blocking the Unicast Flood Packets on the Secure Ports

You can block the unicast flood packets on a secure Ethernet port by disabling the unicast flood feature. If you disable the unicast flood on a port, the port drops the unicast flood packets when it reaches the allowed maximum number of MAC addresses.

The port automatically restarts the unicast flood packet learning when the number of MAC addresses drops below the maximum number that is allowed. The learned MAC address count decreases when a configured MAC address is removed or a time to live counter (TTL) is reached.

Understanding How MAC-Address Monitoring Works

Because the Catalyst 6500 series switches learn the source MAC addresses automatically, the system is vulnerable to flooding of spoofed traffic and potential Denial of Service (DoS) attacks. To prevent the traffic flooding and the DoS attacks, you can monitor the number of MAC addresses that are learned by the system on a per-port, per-VLAN, or per-port-per-VLAN basis.

MAC-address monitoring is supported in the software.

For information on configuring MAC-address monitoring, see the "Configuring MAC-Address Monitoring" section on page 38-14.

Port Security Configuration Guidelines

This section describes the guidelines for configuring port security:

- Do not enable port security on a SPAN destination port and vice versa.
- Do not configure dynamic, static, or permanent CAM entries on a secure port.

Configuring Port Security on the Switch

These sections describe how to configure port security:

Enabling Port Security

When you enable port security on a port, any static or dynamic CAM entries that are associated with the port are cleared; any currently configured permanent CAM entries are treated as secure.

To enable port security, perform this task in privileged mode:

	Task	Command
Step 1	Enable port security on the desired ports. You can also specify the secure MAC address. To enable port security on a trunk port, specify the VLANs on which a secure MAC address is allowed.	**set port security** *mod/port* **enable** [*mac_addr*] [*vlan_list*]
Step 2	Add the MAC addresses to the list of secure addresses.	**set port security** *mod/port mac_addr* [*vlan_list*]
Step 3	Verify the configuration.	**show port** [*mod*[*/port*]] [*mac_addr*][*vlan_list*]

This example shows how to enable port security using the learned MAC address on a port and verify the configuration:

```
Console> (enable) set port security 2/1 enable
Port 2/1 security enabled.
Console> (enable) show port 2/1
Port  Name               Status     Vlan       Level  Duplex Speed Type
----- ------------------ ---------- ---------- ------ ------ ----- ------------
 2/1                     connected  522        normal   half   100 100BaseTX

Port  Security Secure-Src-Addr   Last-Src-Addr     Shutdown Trap     IfIndex
----- -------- ----------------- ----------------- -------- -------- -------
 2/1  enabled  00-90-2b-03-34-08 00-90-2b-03-34-08 No       disabled 1081

Port     Broadcast-Limit Broadcast-Drop
-------- --------------- --------------
 2/1                   -              0

Port  Align-Err  FCS-Err    Xmit-Err   Rcv-Err    UnderSize
----- ---------- ---------- ---------- ---------- ---------
 2/1           0          0          0          0         0

Port  Single-Col Multi-Coll Late-Coll  Excess-Col Carri-Sen Runts     Giants
----- ---------- ---------- ---------- ---------- --------- --------- ---------
 2/1           0          0          0          0         0         0         0

Last-Time-Cleared
--------------------------
Fri Jul 10 1998, 17:53:38
```

This example shows how to enable port security on a port and manually specify the secure MAC address:

```
Console> (enable) set port security 2/1 enable 00-90-2b-03-34-08
Port 2/1 port security enabled with 00-90-2b-03-34-08 as the secure mac address
Trunking disabled for Port 2/1 due to Security Mode
Console> (enable)
```

This example shows how to set port security on a trunk port:

```
Console> (enable) set port security 2/2 00-90-2b-03-34-09 1,20,30
Mac address 00-90-2b-03-34-09 set for port 2/2 on vlan 1,20,20
Console> (enable)
```

Setting the Maximum Number of Secure MAC Addresses

You can set the number of MAC addresses to secure on a port. By default, at least one MAC address per port can be secured. In addition to this default, a global resource is available to be shared by the ports as follows:

- In software releases prior to 8.1(1), you can configure up to 1024 MAC addresses on a port. The total number of MAC addresses on any port cannot exceed 1025.
- In software release 8.1(1) and later releases, you can configure up to 4096 MAC addresses on a port. The total number of MAC addresses on any port cannot exceed 4097.

If the entire global resource of MAC addresses is used on some ports, you can still enable port security on the rest of the ports with a maximum of one MAC per port.

If you reduce the maximum number of MAC addresses, the system clears the specified number of MAC addresses and displays the list of removed addresses.

In software releases 8. 1 and 8.2, you can configure a single MAC address on the access ports that are located on different VLANs but you cannot configure port security on them. In software release 8.3(1) and later releases, which support port security on the trunk ports, a single MAC address can be configured and secured on multiple ports that are in different VLANs. For example, a MAC address "00-00-aa-00-00-aa" can be configured or secured on port 2/1 in VLAN 10 and 2/2 in VLAN 20. If both these ports were in VLAN 10, this MAC address could be configured or secured on only one of these ports. A MAC address can be configured or secured on only one of the ports belonging to a VLAN.

To set the number of MAC addresses to be secured for a particular port, perform this task in privileged mode:

Task	Command
Set the number of MAC addresses to be secured on a port.	**set port security** *mod/port* **maximum** *num_of_mac*

This example shows how to set the number of MAC addresses to be secured:

```
Console> (enable) set port security 7/7 maximum 20
Maximum number of secure addresses set to 20 for port 7/7.
Console> (enable)
```

This example shows how to reduce the number of MAC addresses and the list that displays the cleared MAC addresses:

```
Console> (enable) set port security 7/7 maximum 18
Maximum number of secure addresses set to 18 for port 7/7
00-11-22-33-44-55 cleared from secure address list for port 7/7
00-11-22-33-44-66 cleared from secure address list for port 7/7
Console> (enable)
```

Automatically Configuring Dynamically Learned MAC Addresses

The automatic configuration of dynamically learned MAC addresses enables dynamically learned MAC addresses to be associated with particular ports. This feature applies globally to all secure ports on the system.

The dynamically learned addresses are treated like manually configured addresses and the configuration is stored in NVRAM. The addresses are retained in the event that a secure port is shut down due to a security violation, port security is disabled, or a secure port is administratively disabled.

Note The dynamically learned addresses that have been configured using the automatic configuration option are not cleared under any circumstances. These addresses must be cleared manually by entering the **clear port security** command.

To enable the automatic configuration of dynamically learned MAC addresses, perform this task in privileged mode:

Task	Command
Enable automatic configuration of dynamically learned MAC addresses.	**set port security auto-configure enable | disable**

This example shows how to enable the automatic configuration of dynamically learned MAC addresses globally on the switch:

```
Console> (enable) set port security auto-configure enable
Automatic configuration of secure learnt addresses enabled.
Console> (enable)
```

To view the automatic configuration, enter the **show port security statistics system** command.

```
Console> (enable) show port security statistics system

Auto-Configure Option: Enabled
Module 2:
  Total ports: 24
  Total secure ports: 0
  Total MAC addresses: 24
  Total global address space used (out of 4096): 0
  Status: installed
Module 3:
  Total ports: 48
  Total secure ports: 0
  Total MAC addresses: 48
  Total global address space used (out of 4096): 0
  Status: installed
Module 5:
  Total ports: 2
  Total secure ports: 0
  Total MAC addresses: 2
  Total global address space used (out of 4096): 0
  Status: installed
Total secure ports in the system: 0
Total secure MAC addresses in the system: 74
Total global MAC address resource used in the system (out of 4096): 0
Console> (enable)
```

Setting the Port Security Age Time

The age time on a port specifies how long all addresses on that port will be secured. This age time is activated when a MAC address initiates the traffic on the port. After the age time expires for a MAC address, the entry for that MAC address on the port is removed from the secure address list. The valid range is from 1–1440 minutes. Setting the age time to zero disables the aging of the secure addresses.

To set the age time on a port, perform this task in privileged mode:

Task	Command
Set the age time for which addresses on a port will be secured.	**set port security** *mod/port* **age** *time*

This example shows how to set the age time on port 7/7:

```
Console> (enable) set port security 7/7 age 600
Secure address age time set to 600 minutes for port 7/7.
Console> (enable)
```

Setting the Port Security Aging Type

Note The **set port security** *mod/port* **timer-type** {**absolute** | **inactivity**} command is supported on the Supervisor Engine 720 and Supervisor Engine 32 only.

In software release 8.2(1) and later releases, you can set the type of aging to be applied to the addresses that were learned dynamically on a per-port basis. The two types of aging are as follows:

- Absolute aging—Times out the MAC address after the *age_time* has been exceeded, regardless of the traffic pattern. This is the default for any secured port, and the *age_time* is set to 0.
- Inactivity aging—Times out the MAC address only after the *age_time* of inactivity from the corresponding host has been exceeded.

To set the port-security aging type for the dynamically learned addresses on a per-port basis, perform this task in privileged mode:

Task	Command
Set the port-security aging type for the addresses learned dynamically on a per-port basis.	**set port security** *mod/port* **timer-type** {**absolute** | **inactivity**}

This example shows how to set the different port-security aging types on port 5/1:

```
Console> (enable) set port security 5/1 timer-type absolute
Port 5/1 security timer type absolute.
Console> (enable) set port security 5/1 timer-type inactivity
Port 5/1 security timer type inactive.
Console> (enable)
```

Clearing the MAC Addresses

Enter the **clear port security** command to clear the MAC addresses from a list of secure addresses on a port.

Note If you enter the **clear** command on a MAC address that is in use, that MAC address may be learned and made secure again. We recommend that you disable port security before you clear the MAC addresses.

To clear all or a particular MAC address from the list of secure MAC addresses, perform this task in privileged mode:

Task	Command
Clear all or a particular MAC address from the list of secure MAC addresses. **Note** On the trunk ports, you can clear a MAC address from the list for one or more specific VLANs by using the VLAN list parameter. If you specify the **all** keyword, the MAC address is cleared from the list of secure MAC addresses for all the VLANs on the trunk port.	**clear port security** *mod/port* **all** | *mac_addr* [**all** | *vlan_list*]

This example shows how to clear one MAC address from the secure address list on port 3/37:

```
Console> (enable) clear port security 3/37 00-00-aa-00-00-aa 20,30
Secure MAC address 00-00-aa-00-00-aa cleared for port 3/37 and Vlan 20.
Secure MAC address 00-00-aa-00-00-aa cleared for port 3/37 and Vlan 30.
Console> (enable)
```

This example shows how to clear all the MAC addresses from ports 3/37:

```
Console> (enable) clear port security 3/37 00-00-aa-00-00-aa all
Secure MAC address 00-00-aa-00-00-aa cleared for port 3/37 and Vlan 1.
Secure MAC address 00-00-aa-00-00-aa cleared for port 3/37 and Vlan 20.
Secure MAC address 00-00-aa-00-00-aa cleared for port 3/37 and Vlan 30.
Console> (enable)
```

This example shows how to clear a MAC address from VLAN 1 on trunk port 2/2:

```
Console> (enable) clear port security 2/2  00-90-2b-03-34-09 1
Secure MAC address 00-90-2b-03-34-09 cleared for port 2/2 and Vlan 1.
Console> (enable)
```

Configuring Unicast Flood Blocking on the Secure Ports

To configure unicast flood blocking on a secure port, you must disable the unicast flood feature.

Note The port disables the unicast flooding once the MAC-address limit is reached.

To configure unicast flood blocking on a secure port, perform this procedure in privileged mode:

	Task	Command
Step 1	Disable unicast flood blocking on the desired secure ports.	**set port security** *mod/port* **unicast-flood disable**
Step 2	Verify the configuration of the unicast flood.	**show port security** *mod/port*
Step 3	Verify the status of unicast flood blocking.	**show port unicast-flood** *mod/port*

This example shows how to configure the switch to disable the unicast flood packets on a port and how to verify its configuration:

```
Console> (enable) set port security 4/1 unicast-flood disable
Port 4/1 security flood mode set to disable.
Console> (enable) show port security 4/1
Port  Security Violation Shutdown-Time Age-Time Max-Addr Trap     IfIndex
----- -------- --------- ------------- -------- -------- -------- -------
 4/1  disabled  shutdown             0        0        1 disabled      50

Port  Num-Addr Secure-Src-Addr   Age-Left Last-Src-Addr     Shutdown/Time-Left
----- -------- ----------------- -------- ----------------- ------------------
 4/1        0                 -        -                 -        -         -

Port Flooding on Address Limit
---- -------------------------
 4/1                  Disabled
Console> (enable) show port unicast-flood 4/1
Port      Unicast Flooding
----      ----------------
4/1       Disabled
Console> (enable)
```

Note The **show port unicast-flood** command displays the run-time status of the unicast flood blocking. The output can show the unicast flooding as either enabled or disabled depending if the port has exceeded its address limitation.

Specifying the Security Violation Action

You can set the port for the following two modes to handle a security violation:

- Shutdown—Shuts down the port permanently or for a specified time. Permanent shutdown is the default mode.
- Restrictive—Drops all packets from the insecure hosts but remains enabled.

To specify the security violation action to be taken, perform this task in privileged mode:

Task	Command
Specify the violation action on a port.	**set port security** *mod/port* **violation** {**shutdown** | **restrict**}

This example shows how to specify that port 7/7 drop all packets from the insecure hosts:

```
Console> (enable) set port security 7/7 violation restrict
Port security violation on port 7/7 will cause insecure packets to be dropped.
Console> (enable)
```

Note If you restrict the number of secure MAC addresses on a port to one and additional hosts attempt to connect to that port, port security prevents these additional hosts from connecting to that port and to any other port in the same VLAN for the duration of the VLAN aging time. By default, the VLAN aging time is 5 minutes. If a host is blocked from joining a port in the same VLAN as the secured port, allow the VLAN aging time to expire before you attempt to connect the host to the port again.

Setting the Shutdown Timeout

You can set the time that a port remains disabled in case of a security violation. By default, the port is shut down permanently. The valid range is from 1–1440 minutes.

If the time is set to zero, the shutdown is disabled for this port.

Note When the shutdown timeout expires, the port is reenabled and all port security-related configuration is maintained.

To set the shutdown timeout, perform this task in privileged mode:

Task	Command
Set the shutdown timeout on a port.	**set port security** *mod/port* **shutdown** *time*

This example shows how to set the shutdown timeout to 600 minutes on port 7/7:

```
Console> (enable) set port security 7/7 shutdown 600
Secure address shutdown time set to 600 minutes for port 7/7.
Console> (enable)
```

Disabling Port Security

To disable port security, perform this task in privileged mode:

	Task	Command
Step 1	Disable port security on the desired ports.	**set port security** *mod/port* **disable**
Step 2	Verify the configuration.	**show port security** [*mod/port*]

This example shows how to disable port security:

```
Console> (enable) set port security 2/1 disable
Port 2/1 port security disabled.
Console> (enable)
Console> (enable) show port security 2/1
Port  Security Violation Shutdown-Time Age-Time Max-Addr Trap     IfIndex
----- -------- --------- ------------- -------- -------- -------- -------
 3/24 disabled  restrict            20      300       10 disabled     921

Port  Num-Addr Secure-Src-Addr   Age-Left Last-Src-Addr     Shutdown/Time-Left
----- -------- ----------------- -------- ----------------- ------------------
 3/24        1 00-e0-4f-ac-b4-00        -                 -        -         -
Console> (enable)
```

Restricting the Traffic Based on a Host MAC Address

To restrict the traffic for a specific MAC address, perform this task in privileged mode:

	Task	Command
Step 1	Restrict the traffic destined to or originating from a specific MAC address.	**set cam** {**static** | **permanent**} **filter** *unicast_mac vlan*
Step 2	Remove the filter.	**clear cam** *mac_address vlan*
Step 3	Verify the configuration.	**show cam** {**static** | **permanent**}

This example shows how to create a filter that restricts the traffic for a specific MAC address:

```
Console> (enable) set cam static filter 00-02-03-04-05-06 1
Filter entry added to CAM table.
Console> (enable)
```

This example shows how to clear the filter:

```
Console> (enable) clear cam 00-02-03-04-05-06 1
CAM entry cleared.
Console> (enable)
```

This example shows how to display the static CAM entries:

```
Console> show cam static

VLAN  Dest MAC/Route Des    [CoS]  Destination Ports or VCs / [Protocol Type]
----  ------------------    -----  -------------------------------------------
3     04-04-05-06-07-08   *          FILTER
```

Displaying Port Security

The **show port security** command displays the following information:

- List of secure MAC addresses for a port
- Maximum number of secure addresses that are allowed on a port
- Total number of secure MAC addresses
- Age
- Age left and shutdown timeout left
- Shutdown/security mode
- Statistics that are related to port security

To display the port security configuration information and statistics, perform this task in privileged mode:

	Task	Command
Step 1	Display the configuration.	**show port security** [**statistics**] *mod/port*
Step 2	Display the port security statistics.	**show port security statistics** [**system**] [*mod/port*]

This example shows how to display the port security configuration information and statistics:

```
Console> (enable) show port security 4/1
* = Configured MAC Address

Port  Security Violation Shutdown-Time Age-Time Maximum-Addrs Trap     IfIndex
----- -------- --------- ------------- -------- ------------- -------- -------
 4/1  enabled  shutdown  120           1440     25            disabled 3

Port Secure-Src-Addrs  Age-Left Last-Src-Addr     Shutdown Shutdown-Time-Left
---- ----------------- -------- ----------------- -------- ------------------
 4/1 00-11-22-33-44-55 4        00-11-22-33-44-55 No       -
     00-10-14-da-77-f1 100
Port  Flooding on Address Limit
----- -------------------------
 4/1                    Enabled
Console> (enable) show port security statistics 4/1
Port  Total-Addrs Maximum-Addrs
----- ----------- -------------
 4/1            4            10
Console> (enable)
```

This example shows how to display the port security statistics on a module:

```
Console> (enable) show port security statistics 7
Port  Total-Addrs Maximum-Addrs
----- ----------- -------------
 7/1            0             1
 7/2            0             1
 7/3            0             1
 7/4            0             1
 7/5            0             1
 7/6            0             1
 7/7            0             1
 7/8            0             1
 7/9            0             1
 7/10           0           200
 7/11           0             1
 7/12           0             1
 7/13           0             1
 7/14           0             1
 7/15           0             1
 7/16           0             1
 7/17           0             1
 7/18           0             1
 7/19           0             1
 7/20           0             1
 7/21           0             1
 7/22           0             1
 7/23           0             1
 7/24           0             1
Module 7:
  Total ports: 24
  Total secure ports: 0
  Total MAC address(es): 223
  Total global address space used (out of 4096): 199
  Status: installed
Console> (enable)
```

This example shows how to display the port security statistics on the system:

```
Console> (enable) show port security statistics system

Auto-Configure Option: Enabled
Module 2:
  Total ports: 24
  Total secure ports: 0
  Total MAC addresses: 24
  Total global address space used (out of 4096): 0
  Status: installed
Module 3:
  Total ports: 48
  Total secure ports: 0
  Total MAC addresses: 48
  Total global address space used (out of 4096): 0
  Status: installed
Module 5:
  Total ports: 2
  Total secure ports: 0
  Total MAC addresses: 2
  Total global address space used (out of 4096): 0
  Status: installed
Total secure ports in the system: 0
Total secure MAC addresses in the system: 74
Total global MAC address resource used in the system (out of 4096): 0
Console> (enable)
```

Configuring MAC-Address Monitoring

These sections describe how to configure MAC-address monitoring:

Configuring Global MAC-Address Monitoring

You can enable or disable MAC-address monitoring globally. Globally disabling MAC-address monitoring does not clear any configuration.

To enable or disable MAC-address monitoring globally, perform this task in privileged mode:

Task	Command
Enable or disable MAC-address monitoring globally. **Note** Monitoring is enabled globally by default.	**set cam monitor** {**disable** | **enable**}

This example shows how to disable and enable the global MAC-address monitoring configuration:

```
Console> (enable) set cam monitor disable
Cam monitor disabled
Console> (enable) set cam monitor enable
Cam monitor enabled
Console> (enable)
```

Monitoring the MAC Addresses in the CAM Table

To monitor the MAC addresses that are learned and stored in the CAM table, perform this task in privileged mode:

Task	Command
Monitor the MAC addresses that are learned and stored in the CAM table on a per-port basis, per-VLAN basis, or on a per-port- per-VLAN basis. **Note** MAC-address monitoring is disabled by default on an interface (port, VLAN, or port/VLAN basis).	**set cam monitor** {**disable** | **enable**} [*mod/port* | {*mod/port vlan*} | *vlan*]

This example shows how to monitor the MAC addresses that are learned on a specific port and stored in the CAM table:

```
Console> (enable) set cam monitor enable 3/1
Successfully enabled cam monitor on 3/1
Console> (enable)
```

This example shows how to disable monitoring of the MAC addresses that are learned on a specific port:

```
Console> (enable) set cam monitor disable 3/1
Successfully disabled cam monitor on 3/1
Console> (enable)
```

Specifying the Polling Interval for Monitoring

MAC-address monitoring is supported in the software. If there are a large number of MAC addresses in the CAM table and a large number of configured interfaces (ports, VLANs, or port-VLANs), the CPU usage might go up. You can reduce the load on the CPU by entering the **set cam monitor interval** command to adjust the software polling interval.

To specify the polling interval for the CAM table, perform this task in privileged mode:

Task	Command
Specify the polling interval in seconds for monitoring the CAM table. The valid range is from 5–30 seconds. **Note** The default polling interval is 5 seconds.	**set cam monitor interval** *time_s*

This example shows how to specify the polling interval for the CAM table:

```
Console> (enable) set cam monitor interval 20
Cam monitor interval set to 20 sec
Console> (enable)
```

Specifying the Lower Threshold for MAC-Address Monitoring

To specify the lower threshold for MAC-address monitoring, perform this task in privileged mode:

Task	Command
Specify the lower threshold for MAC-address monitoring and the action to be taken when the system exceeds this threshold. The valid range for the lower threshold is 5–32000. **Note** If you specify the **no-learn** keyword, and the configuration is a port/VLAN configuration, the violation action stops learning the MAC addresses on the port from all the VLANs. If you specify the **warning** keyword, the system displays a system message when the low threshold is exceeded.	**set cam monitor low-threshold** *value* [**action** {**no-learn** | **warning**}] {*mod/port* | {*mod/port vlan*} | *vlan*}

This example shows how to specify the low threshold for a port and the action to be taken when this threshold is exceeded:

```
Console> (enable) set cam monitor low-threshold 500 action warning 3/1
Successfully configured cam monitor on 3/1
Console> (enable)
```

Specifying the Upper Threshold for MAC-Address Monitoring

To specify the upper threshold for MAC-address monitoring, perform this task in privileged mode:

Task	Command
Specify the upper threshold or MAC-address monitoring and the action to be taken when the system exceeds this threshold. The valid range for the high threshold is 5–32000. **Note** If you specify the **no-learn** keyword, and the configuration is a port/VLAN combination, the violation action stops learning the MAC addresses on the port from all the VLANs. If you specify the **shutdown** keyword, and the configuration is a port/VLAN combination, the violation action error disables the port. If you specify the **warning** keyword, the system displays a system message when the high threshold is exceeded.	**set cam monitor high-threshold** *value* [**action** {**no-learn** | **shutdown** | **warning**}] {*mod/port* | {*mod/port vlan*} | *vlan*}

This example shows how to specify the high threshold for a port and the action to be taken when this threshold is exceeded:

```
Console> (enable) set cam monitor high-threshold 28000 action shutdown 3/1
Successfully configured cam monitor on 3/1
Console> (enable)
```

Clearing the Configuration for MAC-Address Monitoring

To clear the configuration for the MAC-address monitoring and actions, perform this task in privileged mode:

Task	Command
Clear the configuration for MAC-address monitoring.	**clear cam monitor** *mod/port* | *mod/port vlan* | *vlan* **clear cam monitor all** **clear cam monitor high-threshold** *mod/port* | *mod/port vlan* | *vlan* **clear cam monitorlow-threshold** *mod/port* | *mod/port vlan* | *vlan*

This example shows how to clear the high threshold on port 3/1:

```
Console> (enable) clear cam monitor high-threshold 3/1
Successfully cleared high-threshold on 3/1
```

This example shows how to clear all CAM table monitoring and MAC-address monitoring configurations from all ports:

```
Console> (enable) clear cam monitor all
Cleared all cam monitor configuration
Console> (enable)
```

Displaying the Configuration for the CAM Monitor

To display the configuration for the CAM monitor, perform this task in privileged mode:

Task	Command
Display the configuration for the CAM monitor.	**show cam monitor** [*mod/port* | *mod/port vlan* | *vlan* | **all**]

This example shows how to display the configuration for the CAM monitor:

```
Console> (enable) show cam monitor all
Cam monitor global configuration:
status : enabled
interval : 5 seconds
* = violation occured

Port Status Low Low High High No. of
Threshold Action Threshold Action mac addrs
------ -------- --------- -------- --------- -------- ---------
3/1 enabled 500 warning 32000 warning 0
4/2 enabled 500 warning* 32000 warning 0

Total port entries = 2

Console> (enable)
```

Displaying the Global Configuration for the CAM Monitor

To display the global CAM monitor configuration, perform this task in privileged mode:

Task	Command
Display the global configuration for the CAM monitor.	**show cam monitor**

```
Console> (enable) show cam monitor
Cam monitor global configuration:
status : enabled
interval : 5 seconds
Console> (enable)
```

CHAPTER 39

Configuring the Switch Access Using AAA

This chapter describes how to configure authentication, authorization, and accounting (AAA) to monitor and control access to the command-line interface (CLI) on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

Note For information on configuring 802.1X authentication to restrict unauthorized devices from connecting to a LAN through publicly accessible ports, see Chapter 40, "Configuring 802.1X Authentication."

Note For information on configuring MAC address authentication bypass, see Chapter 41, "Configuring MAC Authentication Bypass."

Note For information on configuring ports to allow or restrict traffic based on host MAC addresses, see Chapter 38, "Configuring Port Security."

Note For information on configuring network admission control, see Chapter 44, "Configuring Network Admission Control."

This chapter consists of these sections:

Understanding How Authentication Works

These sections describe how the different authentication methods work:

Authentication Overview

You can configure any combination of these authentication methods to control access to the switch:

- Login authentication
- Local authentication
- RADIUS authentication
- TACACS+ authentication
- Kerberos authentication

Note Kerberos authentication does not work if TACACS+ is used as the authentication method.

When you enable local authentication with one or more other authentication methods, local authentication is always attempted last. However, you can specify different authentication methods for the console and Telnet connections. For example, you might use local authentication for the console connections and RADIUS authentication for the Telnet connections.

Understanding How Login Authentication Works

Login authentication increases the security of the system by keeping the unauthorized users from guessing the password. The user is limited to a specific number of attempts to successfully log in to the switch. If the user fails to authorize the password, the system delays the accesses and captures the user ID and the IP address of the station in the syslog and in the SNMP trap.

The maximum number of login attempts is configurable from the CLI and SNMP through the **set authentication login attempt** *count* command. Enter the **set authentication enable attempt** *count* command to set the login limits for accessing enable mode. The configurable range is three (default) to ten tries. Setting the login authentication limit to zero (0) disables this function.

All authentication methods are supported (RADIUS, TACACS+, Kerberos, or local).

You can configure the lockout (delay) time from the CLI and SNMP through the **set authentication login lockout** *time* command. Use the **set authentication enable lockout** *time* command to set a delay time for accessing enable mode. The configurable range is 30–43200 seconds. Setting the lockout time to zero (0) disables this function.

If you are locked out at the console, the console does not allow you to log in during that lockout time. If you are locked out with a Telnet session, the connection closes when the time limit is reached. The switch closes any subsequent access from that station during the lockout time and provides an appropriate notice.

Understanding How Local Authentication Works

Local authentication uses locally configured login and enable passwords to authenticate login attempts. The login and enable passwords are local to each switch and are not mapped to the individual usernames.

By default, local authentication is enabled. You can disable local authentication *only* after enabling one or more of the other authentication methods. However, when local authentication is disabled, if you disable all other authentication methods, local authentication is reenabled automatically.

You can enable local authentication and one or more of the other authentication methods at the same time. The switch attempts local authentication only if the other authentication methods fail.

Understanding How Local User Authentication Works

Local user authentication uses local user accounts and passwords that you create to validate the login attempts of local users. Each switch can have a maximum of 25 local user accounts. Before you can enable local user authentication, you must define at least one local user account.

You set up local user accounts by creating a unique username and password combination for each local user. Each username must be fewer than 65 characters and can be any alphanumeric character (at least one character must be alphabetic).

You configure each local user account with a privilege level; the valid privilege levels are 0 or 15. The privilege level assigned to a username and password combination designates whether a user will be logged in to normal or privileged mode after successful authentication. A user with a privilege level of 0 is automatically logged in to normal mode, and a user with a privilege level of 15 is logged in to privileged mode. A user with a privilege level of 0 can still access privileged mode by entering the **enable** command and password combination. Once a local user is logged in, only the commands that are available for that privilege level can be displayed.

Note If you are running a CiscoView image or are logging in using an HTTP login, the system completes its initial authentication using the username and password combination. You can enter privileged mode by either providing the privilege password or using the username and password combination if the local user has a privilege level of 15.

Understanding How TACACS+ Authentication Works

TACACS+ controls access to the network devices by exchanging Network Access Server (NAS) information between a network device and a centralized database to determine the identity of a user or an entity. TACACS+ is an enhanced version of TACACS, a User Datagram Protocol (UDP)-based access-control protocol that is specified by RFC 1492. TACACS+ uses TCP to ensure reliable delivery and encrypt all traffic between the TACACS+ server and the TACACS+ daemon on a network device.

TACACS+ works with many authentication types, including fixed password, one-time password, and challenge-response authentication. TACACS+ authentication usually occurs in these instances:

- When you first log on to a machine
- When you send a service request that requires privileged access

When you request privileged or restricted services, TACACS+ encrypts your user password information using the MD5 encryption algorithm and adds a TACACS+ packet header. This header information identifies the packet type that is being sent (for example, an authentication packet), the packet sequence number, the encryption type that is used, and the total packet length. The TACACS+ protocol then forwards the packet to the TACACS+ server.

A TACACS+ server can provide authentication, authorization, and accounting functions. These services, while all part of TACACS+, are independent of one another, so a given TACACS+ configuration can use any or all of the three services.

When the TACACS+ server receives the packet, it does the following:

- Authenticates the user information and notifies the client that authentication has either passed or failed.
- Notifies the client that authentication will continue and that the client must provide additional information. This challenge-response process can continue through multiple iterations until authentication either passes or fails.

You can configure a TACACS+ key on the client and server. If you configure a key on the switch, it must be the same as the one that is configured on the TACACS+ servers. The TACACS+ clients and servers use the key to encrypt all the transmitted TACACS+ packets. If you do not configure a TACACS+ key, the packets are not encrypted.

You can configure the following TACACS+ parameters on the switch:

- Enable or disable TACACS+ authentication to determine if a user has permission to access the switch
- Enable or disable TACACS+ authentication to determine if a user has permission to enter privileged mode
- Specify a key that is used to encrypt the protocol packets
- Specify the server on which the TACACS+ server daemon resides
- Set the number of login attempts that are allowed
- Set the timeout interval for a server daemon response
- Enable or disable the directed-request option

TACACS+ authentication is disabled by default. You can enable TACACS+ authentication and local authentication at the same time.

When local authentication is disabled, if you disable all other authentication methods, local authentication is reenabled automatically.

Understanding How RADIUS Authentication Works

RADIUS is a client-server authentication and authorization access protocol that is used by the NAS to authenticate the users attempting to connect to a network device. The NAS functions as a client, passing user information to one or more RADIUS servers. The NAS permits or denies network access to a user based on the response it receives from one or more RADIUS servers. RADIUS uses UDP for transport between the RADIUS client and server.

You can configure a RADIUS key on the client and server. If you configure a key on the client, it must be the same as the one that is configured on the RADIUS servers. The RADIUS clients and servers use the key to encrypt all the transmitted RADIUS packets. If you do not configure a RADIUS key, the packets are not encrypted. The key itself is never transmitted over the network.

Note For more information about how the RADIUS protocol operates, refer to RFC 2138, "Remote Authentication Dial In User Service (RADIUS)."

You can configure the following RADIUS parameters on the switch:

- Enable or disable RADIUS authentication to control login access
- Enable or disable RADIUS authentication to control enable access
- Specify the IP addresses and UDP ports of the RADIUS servers
- Specify the RADIUS key that is used to encrypt the RADIUS packets
- Specify the RADIUS server timeout interval
- Specify the RADIUS retransmit count
- Specify the RADIUS server dead time interval

RADIUS authentication is disabled by default. You can enable RADIUS authentication and other authentication methods at the same time. You can specify which method to use first using the **primary** keyword.

When local authentication is disabled, if you disable all other authentication methods, local authentication is reenabled automatically.

Understanding How Kerberos Authentication Works

Kerberos is a client-server based secret-key network authentication method that uses a trusted Kerberos server to verify secure access to both services and users. In Kerberos, this trusted server is called the key distribution center (KDC). The KDC issues a ticket to validate users and services. A ticket is a temporary set of electronic credentials that verifies the identity of a client for a particular service.

These tickets have a limited life span and can be used in place of the standard user password pair authentication mechanism if a service trusts the Kerberos server that issued the ticket. If the standard user password method is used, Kerberos encrypts the user passwords into the tickets, ensuring that the passwords are not sent on the network in clear text. When you use Kerberos, the passwords are not stored on any machine, other than the Kerberos server, for more than a few seconds. Kerberos also guards against intruders who might pick up the encrypted tickets from the network.

Table 39-1 defines the Kerberos terms.

Table 39-1 Kerberos Terminology

Term	Definition
Kerberized	Applications and services that have been modified to support the Kerberos credential infrastructure.
Kerberos credential	Authentication tickets, such as ticket granting tickets (TGTs), and service credentials. Kerberos credentials verify the ticket of a user or service. If a network service decides to trust the Kerberos server that issued the ticket, the Kerberos credential can be used in place of retyping in a username and password. Credentials have a default life span of eight hours.
Kerberos identity	(See Kerberos principal.)
Kerberos principal	The Kerberos principal is who you are or what a service is according to the Kerberos server. (Also known as a Kerberos identity.)
Kerberos realm	A domain consisting of users, hosts, and network services that are registered to a Kerberos server. The Kerberos server is trusted to verify the identity of a user or network service to another user or network service. Kerberos realms must always be in uppercase characters.
Kerberos server	A daemon running on a network host. Users and network services register their identity with the Kerberos server. Network services query the Kerberos server to authenticate to other network services.
Key distribution center (KDC)	A Kerberos server and database program running on a network host that allocates the Kerberos credentials to different users or network services.
Service credential	A credential for a network service. When issued from the KDC, this credential is encrypted with the password that is shared by the network service and the KDC and with the user's TGT.
SRVTAB	A password that a network service shares with the KDC. The network service authenticates an encrypted service credential by using the SRVTAB (also known as a KEYTAB) to decrypt it.
Ticket granting ticket (TGT)	A credential that the KDC issues to authenticated users. When users receive a TGT, they can authenticate to network services within the Kerberos realm that is represented by the KDC.

In the Catalyst 6500 series switches, the Telnet clients and servers through both the console and in-band management port can be Kerberized.

Note Kerberos authentication does not work if TACACS+ is used as the authentication mechanism.

Note If you are logged in to the console through a modem or a terminal server, you cannot use a Kerberized login procedure.

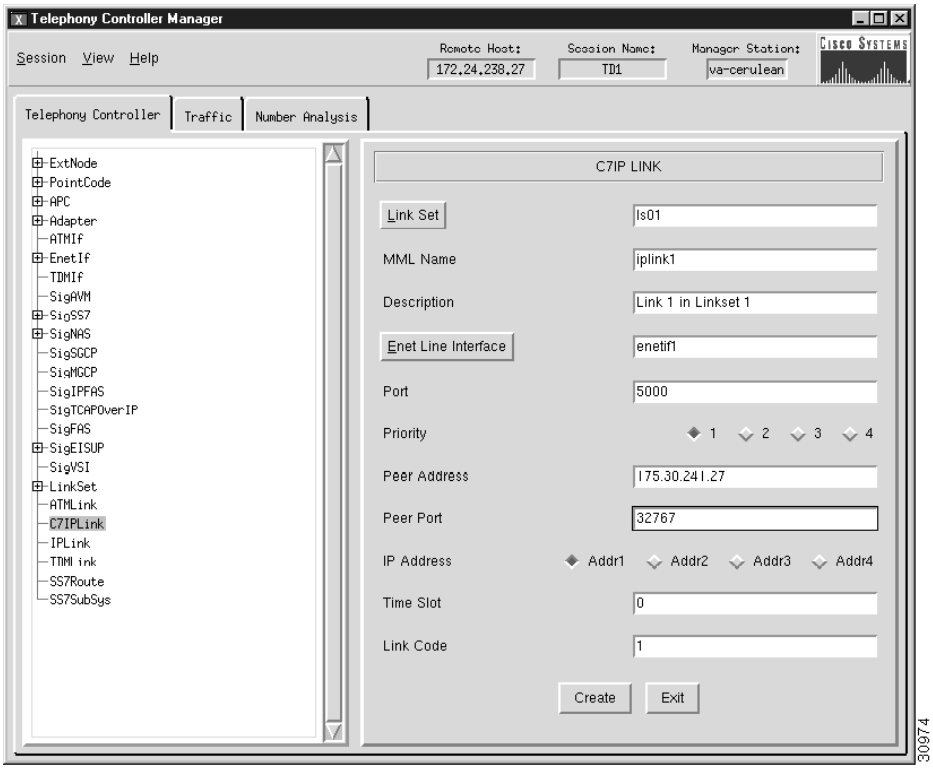
Using a Kerberized Login Procedure

You can use a Kerberized Telnet session if you are logging in through the in-band management port. When the Telnet client and services have been Kerberized, you follow this process when attempting to access the switch through Telnet:

1. The Telnet client asks you for the username and issues a request for a TGT to the KDC on the Kerberos server.
2. The KDC creates the TGT, which contains the user's identity, the KDC's identity, and the TGT's expiration time. The KDC then encrypts the TGT with your password and sends the TGT to the client.
3. When the Telnet client receives the encrypted TGT, it prompts you for the password. If the Telnet client can decrypt the TGT with the entered password, you are successfully authenticated to the KDC. The client then builds a service credential request and sends it to the KDC. This request contains your user identity and a message saying that it wants to access the switch through Telnet. This request is encrypted using the TGT.
4. When the KDC successfully decrypts the service credential request with the TGT that it issued to the client, it builds a service to the switch. The service credential has the client's identity and the identity of the desired Telnet server. The KDC then encrypts the credential with the password that it shares with the switch's Telnet server, encrypts the resulting packet with the Telnet client's TGT, and sends this packet to the client.
5. The Telnet client decrypts the packet first with its TGT. If the encryption is successful, the client then sends the resulting packet to the switch's Telnet server. At this point, the packet is still encrypted with the password that the switch's Telnet server and the KDC share.
6. If the Telnet client has been instructed to do so, it forwards the TGT to the switch. This step ensures that you do not need to get another TGT in order to use another network service from the switch.

Figure 39-1 shows the Kerberos Telnet connection process.

Figure 39-1 Kerberized Telnet Connection



Using a Non-Kerberized Login Procedure

If you use a non-Kerberized login procedure to log in to the switch, the switch takes care of the authentication to the KDC on behalf of the login client. However, the user password is now transferred in clear text from the login client to the switch.

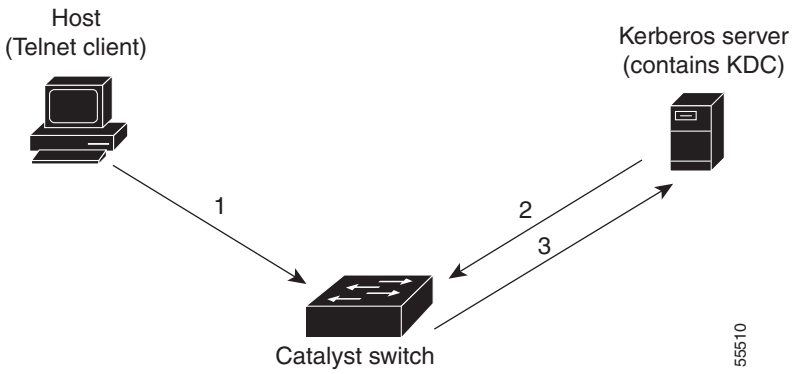
Note A non-Kerberized login can be performed through a modem or terminal server through the in-band management port. Telnet does not support non-Kerberized login.

If you launch a non-Kerberized login, the following process takes place:

1. The switch prompts you for a username and password.
2. The switch requests a TGT from the KDC so that you can be authenticated to the switch.
3. The KDC sends an encrypted TGT to the switch, which contains your identity, KDC's identity, and TGT's expiration time.
4. The switch tries to decrypt the TGT with the password that you entered. If the decryption is successful, you are authenticated to the switch.
5. If you want to access the other network services, the KDC must be contacted directly for authentication. To obtain the TGT, you can run the program "kinit," which is the client software that is provided with the Kerberos package.

Figure 39-2 shows the non-Kerberized login process.

Figure 39-2 Non-Kerberized Telnet Connection



Configuring Authentication on the Switch

These sections describe how to configure the different authentication methods:

- Authentication Default Configuration, page 39-10
- Authentication Configuration Guidelines, page 39-11
- Configuring Login Authentication, page 39-11
- Configuring Local Authentication, page 39-13
- Configuring Local User Authentication, page 39-17

Authentication Default Configuration

Table 39-2 shows the default authentication configuration.

Table 39-2 Authentication Default Configuration

Feature	Default Value
Login authentication (console and Telnet)	Enabled
Local authentication (console and Telnet)	Enabled
Local user authentication	Disabled
TACACS+ login authentication (console and Telnet)	Disabled
TACACS+ enable authentication (console and Telnet)	Disabled
TACACS+ key	None specified
TACACS+ login attempts	3
TACACS+ server timeout	5 seconds
TACACS+ directed request	Disabled
RADIUS login authentication (console and Telnet)	Disabled
RADIUS enable authentication (console and Telnet)	Disabled
RADIUS server IP address	None specified
RADIUS server UDP auth-port	Port 1812
RADIUS key	None specified
RADIUS server timeout	5 seconds
RADIUS server dead time	0 (servers not marked dead)
RADIUS retransmit attempts	2 times
Kerberos login authentication (console and Telnet)	Disabled
Kerberos enable authentication (console and Telnet)	Disabled
Kerberos server IP address	None specified
Kerberos DES key	None specified
Kerberos server auth-port	Port 750
Kerberos local-realm name	NULL string
Kerberos credentials forwarding	Disabled
Kerberos clients mandatory	Not mandatory
Kerberos preauthentication	Disabled

Authentication Configuration Guidelines

This section describes the guidelines for configuring authentication on the switch:

- Authentication configuration applies to both console and Telnet connection attempts unless you use the **console** and **telnet** keywords to specify the authentication methods to use for each connection type individually.
- If you configure a RADIUS or TACACS+ key on the switch, make sure that you configure an identical key on the RADIUS or TACACS+ server.
- You must specify a RADIUS or TACACS+ server before enabling RADIUS or TACACS+ on the switch.
- If you configure multiple RADIUS or TACACS+ servers, the first server that is configured is the primary server and authentication requests are sent to this server first. You can specify a server as primary by using the **primary** keyword.
- RADIUS and TACACS+ support one privileged mode only (level 1).
- Kerberos authentication does not work if TACACS+ is also used as an authentication mechanism.
- Before you can enable local user authentication, you must define at least one username.
- Local user accounts and passwords must be fewer than 65 characters and can consist of any alphanumeric characters. Local user accounts must contain at least one alphabetic character.

Configuring Login Authentication

These sections describe how to configure login authentication on the switch:

- Setting Authentication Login Attempts on the Switch, page 39-11
- Setting Authentication Login Attempts for the Privileged Mode, page 39-12

Setting Authentication Login Attempts on the Switch

To set up login authentication on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Enable login attempt limits on the switch. Enter the **console** or **telnet** keyword if you want to enable local authentication only for the console port or for Telnet connection attempts.	**set authentication login attempt** {*count*} [**console** | **telnet**]
Step 2	Enable the login lockout time on the switch. Enter the **console** or **telnet** keyword if you want to enable local authentication only for the console port or for Telnet connection attempts.	**set authentication login lockout** {*time*} [**console** | **telnet**]
Step 3	Verify the local authentication configuration.	**show authentication**

This example shows how to limit login attempts to 5, set the lockout time for both console and Telnet connections to 50 seconds, and verify the configuration:

```
Console> (enable) set authentication login attempt 5
Login authentication attempts for console and telnet logins set to 5.
Console> (enable) set authentication login lockout 50
Login lockout time for console and telnet logins set to 50.
Console> (enable) show authentication

Login Authentication:  Console Session   Telnet Session    Http Session
---------------------  ----------------  ----------------  ----------------
tacacs                 disabled          disabled          disabled
radius                 disabled          disabled          disabled
kerberos               disabled          disabled          disabled
local                  enabled(primary)  enabled(primary)  enabled(primary)
attempt limit          5                 5                 -
lockout timeout (sec)  50                50                -

Enable Authentication: Console Session   Telnet Session    Http Session
---------------------- ----------------- ----------------  ----------------
tacacs                 disabled          disabled          disabled
radius                 disabled          disabled          disabled
kerberos               disabled          disabled          disabled
local                  enabled(primary)  enabled(primary)  enabled(primary)
attempt limit          3                 3                 -
lockout timeout (sec)  disabled          disabled          -
Console> (enable)
```

Setting Authentication Login Attempts for the Privileged Mode

To set up login authentication for privileged mode, perform this task in privileged mode:

	Task	Command
Step 1	Enable the login attempt limits for privileged mode. Enter the **console** or **telnet** keyword if you want to enable local authentication only for the console port or for Telnet connection attempts.	**set authentication enable attempt** {*count*} [**console** | **telnet**]
Step 2	Enable the login lockout time for privileged mode. Enter the **console** or **telnet** keyword if you want to enable local authentication only for the console port or for Telnet connection attempts.	**set authentication enable lockout** {*time*} [**console** | **telnet**]
Step 3	Verify the local authentication configuration.	**show authentication**

This example shows how to limit enable mode login attempts to 5, set the enable mode lockout time for both console and Telnet connections to 50 seconds, and verify the configuration:

```
Console> (enable) set authentication enable attempt 5
Enable mode authentication attempts for console and telnet logins set to 5.
Console> (enable) set authentication enable lockout 50
Enable mode lockout time for console and telnet logins set to 50.
```

```
Console> (enable) show authentication

Login Authentication:  Console Session   Telnet Session    Http Session
---------------------  ---------------  ----------------  ----------------
tacacs                 disabled         disabled          disabled
radius                 disabled         disabled          disabled
kerberos               disabled         disabled          disabled
local                  enabled(primary) enabled(primary)  enabled(primary)
attempt limit          5                5                 -
lockout timeout (sec)  50               50                -

Enable Authentication: Console Session   Telnet Session    Http Session
---------------------- ----------------- ----------------  ----------------
tacacs                 disabled          disabled          disabled
radius                 disabled          disabled          disabled
kerberos               disabled          disabled          disabled
local                  enabled(primary)  enabled(primary)  enabled(primary)
attempt limit          5                 5                 -
lockout timeout (sec)  50                50                -
Console> (enable)
```

Configuring Local Authentication

These sections describe how to configure local authentication on the switch:

Enabling Local Authentication

Note Local login and enable authentication are enabled for both console and Telnet connections by default. You do not need to perform this task unless you want to modify the default configuration or you have disabled local authentication.

To enable local authentication on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Enable local login authentication on the switch. Enter the **console** or **telnet** keyword if you want to enable local authentication only for the console port or Telnet connection attempts.	**set authentication login local enable** [**all** | **console** | **http** | **telnet**]
Step 2	Enable local enable authentication on the switch. Enter the **console** or **telnet** keyword if you want to enable local authentication only for the console port or Telnet connection attempts.	**set authentication enable local enable** [**all** | **console** | **http** | **telnet**]
Step 3	Verify the local authentication configuration.	**show authentication**

This example shows how to enable local login, enable authentication for both console and Telnet connections, and verify the configuration:

```
Console> (enable) set authentication login local enable
local login authentication set to enable for console and telnet session.
Console> (enable) set authentication enable local enable
local enable authentication set to enable for console and telnet session.
Console> (enable) show authentication

Login Authentication:  Console Session   Telnet Session
---------------------  ----------------  ----------------
tacacs                 disabled          disabled
radius                 disabled          disabled
kerberos               disabled          disabled
local                  enabled(primary)  enabled(primary)

Enable Authentication: Console Session   Telnet Session
---------------------- ----------------- ----------------
tacacs                 disabled          disabled
radius                 disabled          disabled
kerberos               disabled          disabled
local                  enabled(primary)  enabled(primary)
Console> (enable)
```

Setting the Login Password

The login password controls access to the user mode CLI. The passwords are case sensitive, contain up to 19 characters, and use any printable character including a space.

Note The passwords that were set in releases prior to software release 5.4 remain non-case sensitive. You must reset the password after installing software release 5.4 to activate case sensitivity.

To set the login password for local authentication, perform this task in privileged mode:

Task	Command
Set the login password for access. Enter your old password (press **Return** on a switch with no password configured), enter your new password, and reenter your new password.	**set password**

This example shows how to set the login password on the switch:

```
Console> (enable) set password
Enter old password: <old_password>
Enter new password: <new_password>
Retype new password: <new_password>
Password changed.
Console> (enable)
```

Setting the Enable Password

The login password controls access to the user mode CLI. The passwords are case sensitive, contain up to 19 characters, and use any printable character including a space.

Note The passwords that were set in releases prior to software release 5.4 remain non-case sensitive. You must reset the password after installing software release 5.4 to activate case sensitivity.

To set the enable password for local authentication, perform this task in privileged mode:

Task	Command
Set the password for privileged mode. Enter your old password (press **Return** on a switch with no password configured), enter your new password, and reenter your new password.	**set enablepass**

This example shows how to set the enable password on the switch:

```
Console> (enable) set enablepass
Enter old password: <old_password>
Enter new password: <new_password>
Retype new password: <new_password>
Password changed.
Console> (enable)
```

Disabling Local Authentication

Caution Make sure that RADIUS or TACACS+ authentication is configured and operating correctly before disabling local login or enable authentication. If you disable local authentication and RADIUS or TACACS+ is not configured correctly, or if the RADIUS or TACACS+ server is not online, you may be unable to log in to the switch.

To disable local authentication on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Disable local login authentication on the switch. Enter the **console** or **telnet** keyword if you want to disable local authentication only for the console port or Telnet connection attempts.	**set authentication login local disable** [**all** | **console** | **http** | **telnet**]
Step 2	Disable local enable authentication on the switch. Enter the **console** or **telnet** keyword if you want to disable local authentication only for the console port or Telnet connection attempts.	**set authentication enable local disable** [**all** | **console** | **http** | **telnet**]
Step 3	Verify the local authentication configuration.	**show authentication**

Note You must have either RADIUS or TACACS+ authentication enabled before you disable local authentication.

This example shows how to disable local login authentication, enable authentication for both console and Telnet connections, and verify the configuration:

```
Console> (enable) set authentication login local disable
local login authentication set to disable for console and telnet session.
Console> (enable) set authentication enable local disable
local enable authentication set to disable for console and telnet session.
Console> (enable) show authentication

Login Authentication:  Console Session   Telnet Session
---------------------  ----------------  ----------------
tacacs                 disabled          disabled
radius                 enabled(primary)  enabled(primary)
kerberos               disabled          disabled
local                  disabled          disabled

Enable Authentication: Console Session   Telnet Session
---------------------- ----------------- ----------------
tacacs                 disabled          disabled
radius                 enabled(primary)  enabled(primary)
kerberos               disabled          disabled
local                  disabled          disabled
Console> (enable)
```

Recovering a Lost Password

Use the following procedure to recover a lost local authentication password. You must complete Steps 3 through 7 within 30 seconds of a power cycle or the recovery will fail. If you lost both the login and enable passwords, repeat the process for each password.

To recover a lost password, perform these steps in privileged mode:

Step 1 Connect to the switch through the supervisor engine console port. You cannot recover the password if you are connected through a Telnet connection.

Step 2 Enter the **reset system** command to reboot the switch.

Step 3 At the "Enter Password" prompt, press **Return**. The login password is null for 30 seconds when you are connected to the console port.

Step 4 Enter privileged mode using the **enable** command.

Step 5 At the "Enter Password" prompt, press **Return**. (The enable password is null for 30 seconds when you are connected to the console port.)

Step 6 Enter the **set password** or **set enablepass** command, as appropriate.

Step 7 When prompted for your old password, press **Return**.

Step 8 Enter and confirm your new password.

Configuring Local User Authentication

These sections describe how to configure local user authentication on the switch:

Creating a Local User Account

A local user account and password must be fewer than 65 characters and can consist of any alphanumeric characters. A local user account must also contain at least one alphabetic character.

To create a local user account on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Create a new local user account.	**set localuser user** *username* **password** *pwd* **privilege** *privilege_level*
Step 2	Verify the local user account.	**show localusers**

This example shows how to create a local user account and password, set the privilege level, and verify the configuration:

```
Console> (enable) set localuser user picard password captain privilege 15
Added local user picard.
Console> (enable) show localusers
Local User Authentication: disabled
Username                        Privilege Level
---------                        -------------
picard                             15
Console> (enable)
```

Enabling Local User Authentication

To enable local user authentication on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Enable local user authentication.	**set localuser authentication enable**
Step 2	Verify the local user authentication configuration.	**show authentication**

This example shows how to create a local user account, enable local user authentication, and verify the configuration:

```
Console> (enable) set localuser authentication enable
Local User Authentication enabled.
Console> (enable) show authentication
Login Authentication:  Console Session   Telnet Session    Http Session
---------------------  ---------------  ----------------  ----------------
tacacs                 disabled         disabled          disabled
radius                 disabled         disabled          disabled
```

```
kerberos                disabled          disabled          disabled
local  *                enabled(primary)  enabled(primary)  enabled(primary)
attempt limit           3                 3                 -
lockout timeout (sec)   disabled          disabled          -

Enable Authentication: Console Session   Telnet Session    Http Session
---------------------- ----------------- ----------------  ----------------
tacacs                  disabled          disabled          disabled
radius                  disabled          disabled          disabled
kerberos                disabled          disabled          disabled
local  *                enabled(primary)  enabled(primary)  enabled(primary)
attempt limit           3                 3                 -
lockout timeout (sec)   disabled          disabled          -
* Local User Authentication enabled.
Console> (enable)
```

Disabling Local User Authentication

To disable local user authentication on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Disable local user authentication.	**set localuser authentication disable**
Step 2	Verify the local authentication configuration.	**show authentication**

This example shows how to disable local user authentication for the switch and how to verify the configuration:

```
Console> (enable) set localuser authentication disable
local user authentication set to disable.
Console> (enable) show authentication
Login Authentication:  Console Session   Telnet Session    Http Session
---------------------  ----------------  ----------------  ----------------
tacacs                  disabled          disabled          disabled
radius                  disabled          disabled          disabled
kerberos                disabled          disabled          disabled
local  *                enabled(primary)  enabled(primary)  enabled(primary)
attempt limit           3                 3                 -
lockout timeout (sec)   disabled          disabled          -

Enable Authentication: Console Session   Telnet Session    Http Session
---------------------- ----------------- ----------------  ----------------
tacacs                  disabled          disabled          disabled
radius                  disabled          disabled          disabled
kerberos                disabled          disabled          disabled
local  *                enabled(primary)  enabled(primary)  enabled(primary)
attempt limit           3                 3                 -
lockout timeout (sec)   disabled          disabled          -
* Local User Authentication disabled.
Console> (enable)
```

Deleting a Local User Account

To delete a local user account on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Delete a local user account.	**clear localuser picard**
Step 2	Verify that the local user account has been deleted.	**show localusers**

This example shows how to delete local user authentication for the switch and verify the configuration:

```
Console> (enable) clear localuser number1
Local user cleared.
Console> (enable) show localusers
Local User Authentication: enabled
Username                         Privilege Level
---------                         -------------
picard                             15
number1                            0
worf                               15
troy                               0
Console> (enable)
```

Configuring TACACS+ Authentication

These sections describe how to configure TACACS+ authentication on the switch:

Specifying TACACS+ Servers

Specify one or more TACACS+ servers before you enable TACACS+ authentication on the switch. The first server that you specify is the primary server, unless you explicitly make one server the primary using the **primary** keyword.

To specify one or more TACACS+ servers, perform this task in privileged mode:

	Task	Command
Step 1	Specify the IP address of one or more TACACS+ servers.	**set tacacs server** *ip_addr* [**primary**]
Step 2	Verify the TACACS+ configuration.	**show tacacs**

This example shows how to specify TACACS+ servers and verify the configuration:

```
Console> (enable) set tacacs server 172.20.52.3
172.20.52.3 added to TACACS server table as primary server.
Console> (enable) set tacacs server 172.20.52.2 primary
172.20.52.2 added to TACACS server table as primary server.
Console> (enable) set tacacs server 172.20.52.10
172.20.52.10 added to TACACS server table as backup server.
Console> (enable)
Console> (enable) show tacacs

Login Authentication:  Console Session   Telnet Session
---------------------  ----------------  ----------------
tacacs                 disabled          disabled
radius                 disabled          disabled
local                  enabled(primary)  enabled(primary)

Enable Authentication: Console Session   Telnet Session
---------------------- ----------------- ----------------
tacacs                 disabled          disabled
radius                 disabled          disabled
local                  enabled(primary)  enabled(primary)
Tacacs key:
Tacacs login attempts: 3
Tacacs timeout: 5 seconds
Tacacs direct request: disabled
Tacacs-Server                              Status
-----------------------------------------  -------
172.20.52.3
172.20.52.2                                primary
172.20.52.10
Console> (enable)
```

Enabling TACACS+ Authentication

Note Specify at least one TACACS+ server before enabling TACACS+ authentication on the switch. For information on specifying a TACACS+ server, see the "Specifying TACACS+ Servers" section on page 39-19.

You can enable TACACS+ authentication for login and enable access to the switch. If desired, you can use the **console** and **telnet** keywords to specify that TACACS+ authentication is used only on the console or Telnet connections. If you are using both RADIUS and TACACS+, you can use the **primary** keyword to force the switch to try TACACS+ authentication first.

To enable TACACS+ authentication, perform this task in privileged mode:

	Task	Command
Step 1	Enable TACACS+ authentication for normal login mode. Enter the **console** or **telnet** keyword if you want to enable TACACS+ only for the console port or Telnet connection attempts.	**set authentication login tacacs enable** [**all** | **console** | **http** | **telnet**] [**primary**]
Step 2	Enable TACACS+ authentication for enable mode. Enter the **console** or **telnet** keyword if you want to enable TACACS+ only for the console port or Telnet connection attempts.	**set authentication enable tacacs enable** [**all** | **console** | **http** | **telnet**] [**primary**]
Step 3	Verify the TACACS+ configuration.	**show authentication**

This example shows how to enable TACACS+ authentication for the console and Telnet connections and verify the configuration:

```
Console> (enable) set authentication login tacacs enable
tacacs login authentication set to enable for console and telnet session.
Console> (enable) set authentication enable tacacs enable
tacacs enable authentication set to enable for console and telnet session.
Console> (enable) show authentication

Login Authentication:  Console Session   Telnet Session
---------------------  ----------------  ----------------
tacacs                 enabled(primary)  enabled(primary)
radius                 disabled          disabled
local                  enabled           enabled

Enable Authentication: Console Session   Telnet Session
---------------------- ----------------- ----------------
tacacs                 enabled(primary)  enabled(primary)
radius                 disabled          disabled
local                  enabled           enabled
Console> (enable)
```

Specifying the TACACS+ Key

Note If you configure a TACACS+ key on the client, make sure that you configure an identical key on the TACACS+ server.

To specify a TACACS+ key, perform this task in privileged mode:

	Task	Command
Step 1	Specify the key that is used to encrypt packets.	**set tacacs key** *key*
Step 2	Verify the TACACS+ configuration.	**show tacacs**

This example shows how to specify a TACACS+ key and verify the configuration:

```
Console> (enable) set tacacs key Secret_TACACS_key
The tacacs key has been set to Secret_TACACS_key.
Console> (enable) show tacacs
Tacacs key: Secret_TACACS_key
Tacacs login attempts: 3
Tacacs timeout: 5 seconds
Tacacs direct request: disabled

Tacacs-Server                              Status
-----------------------------------------  -------
172.20.52.3
172.20.52.2                                primary
172.20.52.10
Console> (enable)
```

Specifying the TACACS+ Timeout Interval

You can specify the timeout interval between retransmissions to the TACACS+ server. The default timeout is 5 seconds.

To specify a TACACS+ timeout interval, perform this task in privileged mode:

	Task	Command
Step 1	Specify a TACACS+ timeout interval.	**set tacacs timeout** *seconds*
Step 2	Verify the TACACS+ configuration.	**show tacacs**

This example shows how to specify the server timeout interval and verify the configuration:

```
Console> (enable) set tacacs timeout 30
Tacacs timeout set to 30 seconds.
Console> (enable) show tacacs
Tacacs key: Secret_TACACS_key
Tacacs login attempts: 3
Tacacs timeout: 30 seconds
Tacacs direct request: disabled

Tacacs-Server                              Status
-----------------------------------------  -------
172.20.52.3
172.20.52.2                                primary
172.20.52.10
Console> (enable)
```

Specifying the TACACS+ Login Attempts

You can specify the number of failed login attempts that are allowed.

To specify the number of login attempts that are allowed, perform this task in privileged mode:

	Task	Command
Step 1	Specify the number of allowed login attempts.	**set tacacs attempts** *number*
Step 2	Verify the TACACS+ configuration.	**show tacacs**

This example shows how to specify the number of login attempts and verify the configuration:

```
Console> (enable) set tacacs attempts 5
Tacacs number of attempts set to 5.
Console> (enable) show tacacs
Tacacs key: Secret_TACACS_key
Tacacs login attempts: 5
Tacacs timeout: 30 seconds
Tacacs direct request: disabled
Tacacs-Server                           Status
---------------------------------------   -------
172.20.52.3
172.20.52.2                             primary
172.20.52.10
Console> (enable)
```

Enabling TACACS+ Directed Request

When TACACS+ directed request is enabled, you can optionally specify the host name of a configured TACACS+ server to direct the TACACS+ authentication request to that particular TACACS+ server. Authentication will fail if the server that the switch contacts does not have an account for the user that is attempting to log in.

To enable TACACS+ directed request, perform this task in privileged mode:

	Task	Command
Step 1	Enable TACACS+ directed request on the switch.	**set tacacs directedrequest enable**
Step 2	Verify the TACACS+ configuration.	**show tacacs**

This example shows how to enable TACACS+ directed request and verify the configuration:

```
Console> (enable) set tacacs directedrequest enable
Tacacs direct request has been enabled.
Console> (enable) show tacacs
Tacacs key: Secret_TACACS_key
Tacacs login attempts: 5
Tacacs timeout: 30 seconds
Tacacs direct request: enabled

Tacacs-Server                           Status
---------------------------------------   -------
172.20.52.3
172.20.52.2                             primary
172.20.52.10
Console> (enable)
```

Disabling TACACS+ Directed Request

To disable TACACS+ directed request, perform this task in privileged mode:

	Task	Command
Step 1	Disable TACACS+ directed request on the switch.	**set tacacs directedrequest disable**
Step 2	Verify the TACACS+ configuration.	**show tacacs**

This example shows how to disable TACACS+ directed request:

```
Console> (enable) set tacacs directedrequest disable
Tacacs direct request has been disabled.
Console> (enable)
```

Clearing TACACS+ Servers

To clear one or more TACACS+ servers, perform this task in privileged mode:

	Task	Command
Step 1	Specify the IP address of the TACACS+ server to clear from the configuration. Enter the **all** keyword to clear all of the servers from the configuration.	**clear tacacs server** [*ip_addr* | **all**]
Step 2	Verify the TACACS+ server configuration.	**show tacacs**

This example shows how to clear a specific TACACS+ server from the configuration:

```
Console> (enable) clear tacacs server 172.20.52.3
172.20.52.3 cleared from TACACS table
Console> (enable)
```

This example shows how to clear all TACACS+ servers from the configuration:

```
Console> (enable) clear tacacs server all
All TACACS servers cleared
Console> (enable)
```

Clearing the TACACS+ Key

To clear the TACACS+ key, perform this task in privileged mode:

	Task	Command
Step 1	Clear the TACACS+ key.	**clear tacacs key**
Step 2	Verify the TACACS+ configuration.	**show tacacs**

This example shows how to clear the TACACS+ key:

```
Console> (enable) clear tacacs key
TACACS server key cleared.
Console> (enable)
```

Disabling TACACS+ Authentication

When local authentication is disabled and *only* TACACS+ authentication is enabled, if you disable TACACS+ authentication, local authentication is reenabled automatically.

To disable TACACS+ authentication, perform this task in privileged mode:

	Task	Command
Step 1	Disable TACACS+ authentication for normal login mode. Enter the **console** or **telnet** keyword if you want to disable TACACS+ only for the console port or Telnet connection attempts.	**set authentication login tacacs disable** [**all** | **console** | **http** | **telnet**]
Step 2	Disable TACACS+ authentication for enable mode. Enter the **console** or **telnet** keyword if you want to disable TACACS+ only for the console port or Telnet connection attempts.	**set authentication enable tacacs disable** [**all** | **console** | **http** | **telnet**]
Step 3	Verify the TACACS+ configuration.	**show authentication**

This example shows how to disable TACACS+ authentication for the console and Telnet connections and verify the configuration:

```
Console> (enable) set authentication login tacacs disable
tacacs login authentication set to disable for console and telnet session.
Console> (enable) set authentication enable tacacs disable
tacacs enable authentication set to disable for console and telnet session.
Console> (enable) show authentication

Login Authentication:  Console Session   Telnet Session
---------------------  ----------------  ----------------
tacacs                 disabled          disabled
radius                 disabled          disabled
local                  enabled(primary)  enabled(primary)

Enable Authentication: Console Session   Telnet Session
---------------------- ----------------- ----------------
tacacs                 disabled          disabled
radius                 disabled          disabled
local                  enabled(primary)  enabled(primary)
Console> (enable)
```

Configuring RADIUS Authentication

These sections describe how to configure RADIUS authentication on the switch:

Specifying RADIUS Servers

To specify one or more RADIUS servers, perform this task in privileged mode:

	Task	Command
Step 1	Specify the IP address of up to three RADIUS servers. Specify the primary server using the **primary** keyword. Optionally, specify the destination UDP port to use on the server.	**set radius server** *ip_addr* [**auth-port** *port*] [**primary**]
Step 2	Verify the RADIUS server configuration.	**show radius**

This example shows how to specify a RADIUS server and verify the configuration:

```
Console> (enable) set radius server 172.20.52.3
172.20.52.3 with auth-port 1812 added to radius server table as primary server.
Console> (enable) show radius

Login Authentication:  Console Session   Telnet Session
---------------------  ----------------  ----------------
tacacs                 disabled          disabled
radius                 disabled          disabled
local                  enabled(primary)  enabled(primary)
Enable Authentication: Console Session   Telnet Session
---------------------- ----------------- ----------------
tacacs                 disabled          disabled
radius                 disabled          disabled
local                  enabled(primary)  enabled(primary)

Radius Deadtime:              0 minutes
Radius Key:
Radius Retransmit:            2
Radius Timeout:               5 seconds

Radius-Server                 Status   Auth-port
----------------------------- -------  ------------
172.20.52.3                   primary  1812
Console> (enable)
```

Specifying the RADIUS Key

Note If you specify a RADIUS key on the client, make sure that you specify an identical key on the RADIUS server.

The RADIUS key is used to encrypt and authenticate all communication between the RADIUS client and server. You must configure the same key on the client and the RADIUS server.

The key is limited to 65 characters. It can include any printable ASCII characters except tabs.

To specify a RADIUS key, perform this task in privileged mode:

	Task	Command
Step 1	Specify the RADIUS key that is used to encrypt packets that are sent to the RADIUS server.	**set radius key** *key*
Step 2	Verify the RADIUS configuration.	**show radius**

This example shows how to specify a RADIUS key and verify the configuration (in normal mode, the RADIUS key value is hidden):

```
Console> (enable) set radius key Secret_RADIUS_key
Radius key set to Secret_RADIUS_key
Console> (enable) show radius
Login Authentication:  Console Session   Telnet Session
---------------------  ----------------  ----------------
tacacs                 disabled          disabled
radius                 enabled(primary)  enabled(primary)
local                  enabled           enabled

Enable Authentication: Console Session   Telnet Session
---------------------- ----------------- ----------------
tacacs                 disabled          disabled
radius                 enabled(primary)  enabled(primary)
local                  enabled           enabled

Radius Deadtime:              0 minutes
Radius Key:                   Secret_RADIUS_key
Radius Retransmit:            2
Radius Timeout:               5 seconds

Radius-Server                 Status   Auth-port
----------------------------- -------  ------------
172.20.52.3                   primary  1812
Console> (enable)
```

Enabling RADIUS Authentication

Note Specify at least one RADIUS server before enabling RADIUS authentication on the switch. For information on specifying a RADIUS server, see the "Specifying RADIUS Servers" section on page 39-26.

You can enable RADIUS authentication for login and enable access to the switch. If desired, you can enter the **console** or **telnet** keyword to specify that RADIUS authentication is used only on console or Telnet connections. If you are using both RADIUS and TACACS+, you can use the **primary** keyword to force the switch to try RADIUS authentication first.

To set up the RADIUS username and enable RADIUS authentication, perform this task in privileged mode:

	Task	Command
Step 1	Enable RADIUS authentication for normal login mode. Enter the **console** or **telnet** keyword if you want to enable RADIUS only for the console port or Telnet connection attempts.	**set authentication login radius enable** [**all** | **console** | **http** | **telnet**] [**primary**]
Step 2	Enable RADIUS authentication for enable mode. Enter the **console** or **telnet** keyword if you want to enable RADIUS only for the console port or Telnet connection attempts.	**set authentication enable radius enable** [**all** | **console** | **http** | **telnet**] [**primary**]
Step 3	Create a user $enab15$ on the RADIUS server and assign a password to that user.	See the Note below for additional information.
Step 4	Verify the RADIUS configuration.	**show authentication**

Note To use RADIUS authentication for enable mode, you must create a user $enab15$ on the RADIUS server and assign a password to that user. This user needs to be created in addition to your assigned username and password on the RADIUS server (for example, the username is john, and the password is hello). After you log in to the Catalyst 6500 series switch with your assigned username and password (john/hello), you can enter enable mode using the password that is assigned to the $enab15$ user.

If your RADIUS server does not support the $enab15$ username, you can set the service-type attribute (attribute 6) to Administrative (value 6) for a RADUIS user to directly launch the user into enable mode without asking for a separate enable password.

This example shows how to enable RADIUS authentication and verify the configuration:

```
Console> (enable) set authentication login radius enable
radius login authentication set to enable for console and telnet session.
Console> (enable) set authentication enable radius enable
radius enable authentication set to enable for console and telnet session.
Console> (enable) show authentication

Login Authentication:  Console Session   Telnet Session
---------------------  ----------------  ----------------
tacacs                 disabled          disabled
radius                 enabled(primary)  enabled(primary)
local                  enabled           enabled

Enable Authentication: Console Session   Telnet Session
---------------------- ----------------- ----------------
tacacs                 disabled          disabled
radius                 enabled(primary)  enabled(primary)
local                  enabled           enabled
Console> (enable)
```

Specifying the RADIUS Timeout Interval

You can specify the timeout interval between the retransmissions to the RADIUS server. The default timeout is 5 seconds.

To specify the RADIUS timeout interval, perform this task in privileged mode:

	Task	Command
Step 1	Specify the RADIUS timeout interval.	**set radius timeout** *seconds*
Step 2	Verify the RADIUS configuration.	**show radius**

This example shows how to specify the RADIUS timeout interval and verify the configuration:

```
Console> (enable) set radius timeout 10
Radius timeout set to 10 seconds.
Console> (enable) show radius

Login Authentication:  Console Session   Telnet Session
---------------------  ----------------  ----------------
tacacs                 disabled          disabled
radius                 enabled(primary)  enabled(primary)
local                  enabled           enabled

Enable Authentication: Console Session   Telnet Session
---------------------- ----------------- ----------------
tacacs                 disabled          disabled
radius                 enabled(primary)  enabled(primary)
local                  enabled           enabled

Radius Deadtime:              0 minutes
Radius Key:                   Secret_RADIUS_key
Radius Retransmit:            2
Radius Timeout:               10 seconds

Radius-Server                 Status   Auth-port
----------------------------- -------  ------------
172.20.52.3                   primary  1812
Console> (enable)
```

Specifying the RADIUS Retransmit Count

You can specify the number of times that the switch will attempt to contact a RADIUS server before the next configured server is tried. By default, each RADIUS server is tried two times.

To specify the RADIUS retransmit count, perform this task in privileged mode:

	Task	Command
Step 1	Specify the RADIUS server retransmit count.	**set radius retransmit** *count*
Step 2	Verify the RADIUS configuration.	**show radius**

This example shows how to specify the RADIUS retransmit count and verify the configuration:

```
Console> (enable) set radius retransmit 4
Radius retransmit count set to 4.
Console> (enable) show radius

Login Authentication:  Console Session   Telnet Session
---------------------  ----------------  ----------------
tacacs                 disabled          disabled
radius                 enabled(primary)  enabled(primary)
local                  enabled           enabled
Enable Authentication: Console Session   Telnet Session
---------------------- ----------------- ----------------
tacacs                 disabled          disabled
radius                 enabled(primary)  enabled(primary)
local                  enabled           enabled

Radius Deadtime:              0 minutes
Radius Key:                   Secret_RADIUS_key
Radius Retransmit:            4
Radius Timeout:               10 seconds

Radius-Server                 Status   Auth-port
----------------------------- -------  ------------
172.20.52.3                   primary  1812
Console> (enable)
```

Specifying the RADIUS Dead Time

You can configure the switch so that, when a RADIUS server does not respond to an authentication request, the switch marks that server as dead for the length of time that is specified by the dead time. Any authentication requests that are received during the dead time interval (such as other users attempting to log in to the switch) are not sent to a RADIUS server that is marked dead. Configuring a dead time speeds up the authentication process by eliminating the timeouts and the retransmissions to the dead RADIUS server.

If you configure only one RADIUS server, or if all of the configured servers are marked dead, the dead time is ignored because no alternate servers are available.

To set the RADIUS dead time, perform this task in privileged mode:

	Task	Command
Step 1	Specify the RADIUS server dead time.	**set radius deadtime** *minutes*
Step 2	Verify the RADIUS configuration.	**show radius**

This example shows how to specify the RADIUS dead time and verify the configuration:

```
Console> (enable) set radius deadtime 5
Radius deadtime set to 5 minute(s)
Console> (enable) show radius

Login Authentication:  Console Session   Telnet Session
---------------------  ----------------  ----------------
tacacs                 disabled          disabled
radius                 enabled(primary)  enabled(primary)
local                  enabled           enabled
```

```
Enable Authentication: Console Session   Telnet Session
---------------------- ----------------- ----------------
tacacs                 disabled          disabled
radius                 enabled(primary)  enabled(primary)
local                  enabled           enabled

Radius Deadtime:              5 minutes
Radius Key:                   Secret_RADIUS_key
Radius Retransmit:            4
Radius Timeout:               10 seconds

Radius-Server                 Status   Auth-port
----------------------------- -------  ------------
172.20.52.3                   primary  1812
172.20.52.2                            1812
Console> (enable)
```

Specifying Optional Attributes for RADIUS Servers

You can specify optional attributes in the RADIUS ACCESS_REQUEST packet. The **set radius attribute** command allows you to specify the transmission of certain optional attributes such as Framed-IP address, NAS-Port, Called-Station-Id, Calling-Station-Id, and so on. You can set attribute transmission by either the attribute number or the attribute name. Transmission of the attributes is disabled by default.

Note Software release 7.5(1) supports only the Framed-IP address (Attribute 8).

To specify the optional attributes for the RADIUS server, perform this task in privileged mode:

	Task	Command
Step 1	Specify the optional attributes for the RADIUS server.	**set radius attribute** [*number* | *name*] **include-in-access-req** [**enable** | **disable**]
Step 2	Verify the RADIUS configuration.	**show radius**

This example shows how to specify and enable the Framed-IP address attribute by number and verify the configuration:

```
Console> (enable) set radius attribute 8 include-in-access-req enable
Transmission of Framed-ip address in access-request packet is enabled.
Console> (enable) show radius
RADIUS Deadtime:              0 minutes
RADIUS Key:                   123456
RADIUS Retransmit:            2
RADIUS Timeout:               5 seconds
Framed-Ip Address Transmit:   Enabled

RADIUS-Server                 Status   Auth-port     Acct-port
----------------------------- -------  ------------  ------------
10.6.140.230                  primary  1812          1813
Console> (enable)
```

This example shows how to specify and disable the Framed-IP address attribute by name:

```
Console> (enable) set radius attribute framed-ip-address include-in-access-req disable
Transmission of Framed-ip address in access-request packet is disabled.
Console> (enable)
```

Clearing RADIUS Servers

To clear one or more RADIUS servers, perform this task in privileged mode:

	Task	Command
Step 1	Specify the IP address of the RADIUS server to clear from the configuration. Enter the **all** keyword to clear all of the servers from the configuration.	**clear radius server** [*ip_addr* | **all**]
Step 2	Verify the RADIUS server configuration.	**show radius**

This example shows how to clear a single RADIUS server from the configuration:

```
Console> (enable) clear radius server 172.20.52.3
172.20.52.3 cleared from radius server table.
Console> (enable)
```

This example shows how to clear all RADIUS servers from the configuration:

```
Console> (enable) clear radius server all
All radius servers cleared from radius server table.
Console> (enable)
```

Clearing the RADIUS Key

To clear the RADIUS key, perform this task in privileged mode:

	Task	Command
Step 1	Clear the RADIUS key.	**clear radius key**
Step 2	Verify the RADIUS configuration.	**show radius**

This example shows how to clear the RADIUS key and verify the configuration:

```
Console> (enable) clear radius key
Radius key cleared.
Console> (enable) show radius

Login Authentication:  Console Session   Telnet Session
---------------------  ----------------  ----------------
tacacs                 disabled          disabled
radius                 disabled          disabled
local                  enabled(primary)  enabled(primary)

Enable Authentication: Console Session   Telnet Session
---------------------- ----------------- ----------------
tacacs                 disabled          disabled
radius                 disabled          disabled
local                  enabled(primary)  enabled(primary)
```

```
Radius Deadtime:              0 minutes
Radius Key:
Radius Retransmit:            2
Radius Timeout:               5 seconds

Radius-Server                 Status   Auth-port
----------------------------- -------  ------------
172.20.52.3                   primary  1812
Console> (enable)
```

Disabling RADIUS Authentication

When local authentication is disabled and *only* RADIUS authentication is enabled, if you disable RADIUS authentication, local authentication is reenabled automatically.

To disable RADIUS authentication, perform this task in privileged mode:

	Task	Command
Step 1	Disable RADIUS authentication for login mode.	**set authentication login radius disable** [**all** | **console** | **http** | **telnet**]
Step 2	Disable RADIUS authentication for enable mode.	**set authentication enable radius disable** [**all** | **console** | **http** | **telnet**]
Step 3	Verify the RADIUS configuration.	**show authentication**

This example shows how to disable RADIUS authentication and verify the configuration:

```
Console> (enable) set authentication login radius disable
radius login authentication set to disable for console and telnet session.
Console> (enable) set authentication enable radius disable
radius enable authentication set to disable for console and telnet session.
Console> (enable) show authentication

Login Authentication:  Console Session   Telnet Session
---------------------  ----------------  ----------------
tacacs                 disabled          disabled
radius                 disabled          disabled
local                  enabled(primary)  enabled(primary)

Enable Authentication: Console Session   Telnet Session
---------------------- ----------------- ----------------
tacacs                 disabled          disabled
radius                 disabled          disabled
local                  enabled(primary)  enabled(primary)
Console> (enable)
```

Configuring Kerberos Authentication

These sections describe how to configure Kerberos authentication on the switch:

Configuring a Kerberos Server

Before you can use Kerberos as an authentication method on the switch, you need to configure the Kerberos server. You will need to create a database for the KDC and add the switch to the database.

Note Kerberos authentication requires that NTP is enabled. Additionally, we recommend that you enable DNS.

To configure the Kerberos server, perform these steps:

Step 1 Before you can enter the switch in the Kerberos server's key table, you must create the database that the KDC will use. In the following example, a database called CISCO.EDU is created:

```
/usr/local/sbin/kdb5_util create -r CISCO.EDU -s
```

Step 2 Add the switch to the database. The following example adds a switch called Cat6509 to the CISCO.EDU database:

```
ank host/Cat6509.cisco.edu@CISCO.EDU
```

Step 3 Add the username as follows:

```
ank user1@CISCO.EDU
```

Step 4 Add the administrative principals as follows:

```
ank user1/admin@CISCO.EDU
```

Step 5 Using the **admin.local ktadd** command, create the database entry for the switch as follows:

```
ktadd host/Cat6509.cisco.edu@CISCO.EDU
```

Step 6 Move the keytab file to a place where the switch can reach it.

Step 7 Start the KDC server as follows:

```
/usr/local/sbin/krb5kdc
    /usr/local/sbin/kadmind
```

Enabling Kerberos

To enable Kerberos authentication, perform this task in privileged mode:

	Task	Command
Step 1	Specify Kerberos as the authentication method.	**set authentication login kerberos enable** [**all** | **console** | **http** | **telnet**] [**primary**]
Step 2	Verify the configuration.	**show authentication**

This example shows how to enable Kerberos as the login authentication method for Telnet and verify the configuration:

```
kerberos> (enable) set authentication login kerberos enable telnet
kerberos login authentication set to enable for telnet session.
kerberos> (enable) show authentication

Login Authentication: Console Session   Telnet Session
---------------------  ----------------  ----------------
tacacs                 disabled          disabled
radius                 disabled          disabled
kerberos               disabled          enabled(primary)
local                  enabled(primary)  enabled

Enable Authentication:Console Session   Telnet Session
---------------------- ----------------- ----------------
tacacs                 disabled          disabled
radius                 disabled          disabled
kerberos               disabled          enabled(primary)
local                  enabled(primary)  enabled
kerberos> (enable)
```

This example shows how to enable Kerberos as the login authentication method for the console and verify the configuration:

```
kerberos> (enable) set authentication login kerberos enable console
kerberos login authentication set to enable for console session.
kerberos> (enable) show authentication

Login Authentication: Console Session   Telnet Session
---------------------  ----------------  ----------------
tacacs                 disabled          disabled
radius                 disabled          disabled
kerberos               enabled(primary)  enabled(primary)
local                  enabled           enabled

Enable Authentication:Console Session   Telnet Session
---------------------- ----------------- ----------------
tacacs                 disabled          disabled
radius                 disabled          disabled
kerberos               enabled(primary)  enabled(primary)
local                  enabled           enabled
kerberos> (enable)
```

Defining the Kerberos Local Realm

The Kerberos realm is a domain consisting of users, hosts, and network services that are registered to a Kerberos server. To authenticate a user that is defined in the Kerberos database, the switch must know the host name or IP address of the host running the KDC and the name of the Kerberos realm.

To configure the switch to authenticate to the KDC in a specified Kerberos realm, perform this task in privileged mode:

Task	Command
Define the default realm for the switch.	**set kerberos local-realm** *kerberos_realm*

Note Make sure that the realm is entered in uppercase letters. Kerberos will not authenticate users if the realm is entered in lowercase letters.

This example shows how to define a local realm and verify the configuration:

```
kerberos> (enable) set kerberos local-realm CISCO.COM
Kerberos local realm for this switch set to CISCO.COM.
kerberos> (enable) show kerberos
Kerberos Local Realm:CISCO.COM
Kerberos server entries:
Realm:CISCO.COM,  Server:187.0.2.1,  Port:750

Kerberos Domain<->Realm entries:
Domain:cisco.com,  Realm:CISCO.COM

Kerberos Clients NOT Mandatory
Kerberos Credentials Forwarding Enabled
Kerberos Pre Authentication Method set to None
Kerberos config key:
Kerberos SRVTAB Entries
Srvtab Entry 1:host/niners.cisco.com@CISCO.COM 0 932423923 1 1 8 01;;8>00>50;0=0=0
kerberos> (enable)
```

Specifying a Kerberos Server

You can specify to the switch which KDC to use in a specific Kerberos realm. Optionally, you can also specify the port number that the KDC is monitoring. The Kerberos server information that you enter is maintained in a table with one entry for each Kerberos realm. The maximum number of entries in the table is 100.

To specify the Kerberos server, perform this task in privileged mode:

	Task	Command
Step 1	Specify which KDC to use in a given Kerberos realm. Optionally, enter the port number that the KDC is monitoring. (The default port number is 750.)	**set kerberos server** *kerberos_realm* {*hostname* | *ip_address*} [*port*]
Step 2	Clear the Kerberos server entry.	**clear kerberos server** *kerberos_realm* {*hostname* | *ip_address*} [*port*]

This example shows how to specify which Kerberos server will serve as the KDC for the specified Kerberos realm and clear the entry:

```
kerberos> (enable) set kerberos server CISCO.COM 187.0.2.1 750
Kerberos Realm-Server-Port entry set to:CISCO.COM - 187.0.2.1 - 750
kerberos> (enable)

Console> (enable) clear kerberos server CISCO.COM 187.0.2.1 750
Kerberos Realm-Server-Port entry CISCO.COM-187.0.2.1-750  deleted
Console> (enable)
```

Mapping a Kerberos Realm to a Host Name or DNS Domain

Optionally, you can map a host name or domain name system (DNS) domain to a Kerberos realm.

To map a Kerberos realm to either a host name or DNS domain, perform this task in privileged mode:

	Task	Command
Step 1	(Optional) Map a host name or DNS domain to a Kerberos realm.	**set kerberos realm** {*dns_domain* | *host*} *kerberos_realm*
Step 2	Clear the Kerberos realm domain or host mapping entry.	**clear kerberos realm** {*dns_domain* | *host*} *kerberos_realm*

This example shows how to map a Kerberos realm to a DNS domain and clear the entry:

```
Console> (enable) set kerberos realm CISCO CISCO.COM
Kerberos DnsDomain-Realm entry set to CISCO - CISCO.COM
Console> (enable)

Console> (enable) clear kerberos realm CISCO CISCO.COM
Kerberos DnsDomain-Realm entry CISCO - CISCO.COM deleted
Console> (enable)
```

Copying SRVTAB Files

To allow the remote users to authenticate to the switch using the Kerberos credentials, the switch must share a key with the KDC. You must give the switch a copy of the key, which is on a file that is stored in the KDC. These files are called SRVTAB files on the switch and KEYTAB files on the servers.

The most secure method to copy the SRVTAB files to the hosts in your Kerberos realm is to copy them onto physical media and go to each host in turn and manually copy the files onto the system. To copy the SRVTAB files to a switch that does not have a physical media drive, you must transfer the files through the network by using the Trivial File Transfer Protocol (TFTP).

When you copy the SRVTAB file from the switch to the KDC, the switch parses the information in this file and stores it in the running configuration in the Kerberos SRVTAB entry format. If you enter the SRVTAB directly into the switch, create an entry for each Kerberos principal (service) on the switch. The entries are maintained in the SRVTAB table. The maximum size of the table is 20 entries.

To retrieve the SRVTAB files to the switch from the KDC, perform this task in privileged mode:

	Task	Command
Step 1	Retrieve a specified SRVTAB file from the KDC.	**set kerberos srvtab remote** {*hostname* | *ip_address*} *filename*
Step 2	(Optional) Enter the SRVTAB directly into the switch.	**set kerberos srvtab entry** *kerberos_principal principal_type timestamp key_version number key_type key_length encrypted_keytab*

This example shows how to retrieve an SRVTAB file from the KDC, enter an SRVTAB directly into the switch, and verify the configuration:

```
kerberos> (enable) set kerberos srvtab remote 187.20.32.10 /users/jdoe/krb5/ninerskeytab
kerberos> (enable)

kerberos> (enable) set kerberos srvtab entry host/niners.cisco.com@CISCO.COM 0 932423923 1
1 8 03;;5>00>50;0=0=0
Kerberos SRVTAB entry set to
Principal:host/niners.cisco.com@CISCO.COM
Principal Type:0
Timestamp:932423923
Key version number:1
Key type:1
Key length:8
Encrypted key tab:03;;5>00>50;0=0=0

kerberos> (enable) show kerberos
Kerberos Local Realm:CISCO.COM
Kerberos server entries:
Realm:CISCO.COM,  Server:187.0.2.1,  Port:750
Realm:CISCO.COM,  Server:187.20.2.1,  Port:750

Kerberos Domain<->Realm entries:
Domain:cisco.com,  Realm:CISCO.COM

Kerberos Clients NOT Mandatory
Kerberos Credentials Forwarding Enabled
Kerberos Pre Authentication Method set to None
Kerberos config key:
Kerberos SRVTAB Entries
Srvtab Entry 1:host/niners.cisco.com@CISCO.COM 0 932423923 1 1 8 03;;5>00>50;0=0=0
Srvtab Entry 2:host/niners.cisco.edu@CISCO.EDU 0 933974942 1 1 8 00?58:127:223=:;9
Console> (enable)
```

Deleting an SRVTAB Entry

To delete an SRVTAB entry, perform this task in privileged mode:

Task	Command
Delete the SRVTAB entry for a particular Kerberos principal.	**clear kerberos srvtab entry** *kerberos_principal principal_type*

This example shows how to delete an SRVTAB entry:

```
kerberos> (enable) clear kerberos srvtab entry host/niners.cisco.com@CISCO.COM 0
kerberos> (enable)
```

Enabling Credentials Forwarding

A user that is authenticated to a Kerberized switch has a TGT and can use it to authenticate to a host on the network. However, if forwarding is not enabled and a user tries to list the credentials after authenticating to a host, the output will show that no Kerberos credentials are present.

To enable credentials forwarding, configure the switch to forward user TGTs when they authenticate from the switch to the Kerberized remote hosts on the network using Kerberized Telnet.

As an additional layer of security, you can configure the switch so that after the users authenticate to it, these users can authenticate only to the other services on the network with the Kerberized clients. If you do not make Kerberos authentication mandatory and Kerberos authentication fails, the application attempts to authenticate the users using the default method of authentication for that network service. For example, Telnet prompts for a password.

To configure the clients to forward the user credentials as they connect to the other hosts in the Kerberos realm, perform this task in privileged mode:

	Task	Command
Step 1	Enable all clients to forward the user credentials upon successful Kerberos authentication.	**set kerberos credentials forward**
Step 2	(Optional) Configure Telnet to fail if the clients cannot authenticate to the remote server.	**set kerberos clients mandatory**

This example shows how to configure the clients to forward the user credentials and verify the configuration:

```
kerberos> (enable) set kerberos credentials forward
Kerberos credentials forwarding enabled
kerberos> (enable) show kerberos
Kerberos Local Realm:CISCO.COM
Kerberos server entries:
Realm:CISCO.COM,  Server:187.0.2.1,  Port:750
Realm:CISCO.COM,  Server:187.20.2.1, Port:750

Kerberos Domain<->Realm entries:
Domain:cisco.com,  Realm:CISCO.COM

Kerberos Clients NOT Mandatory
Kerberos Credentials Forwarding Enabled
Kerberos Pre Authentication Method set to None
Kerberos config key:
Kerberos SRVTAB Entries
Srvtab Entry 1:host/aspen-niners.cisco.edu@CISCO.EDU 0 933974942 1 1 8 00?91:107:423=:;9
kerberos> (enable)
```

This example shows how to configure the switch so that Kerberos clients are mandatory for users to authenticate to other network services:

```
Console> (enable) set kerberos clients mandatory
Kerberos clients set to mandatory
Console> (enable)
```

Disabling Credentials Forwarding

To disable the credentials forwarding, perform this task in privileged mode:

Task	Command
Disable the credentials forwarding configuration.	**clear kerberos credentials forward**

This example shows how to disable the credentials forwarding and verify the change:

```
Console> (enable) clear kerberos credentials forward
Kerberos credentials forwarding disabled
Console> (enable) show kerberos
Kerberos Local Realm not configured
Kerberos server entries:

Kerberos Domain<->Realm entries:

Kerberos Clients NOT Mandatory
Kerberos Credentials Forwarding Disabled
Kerberos Pre Authentication Method set to None
Kerberos config key:
Kerberos SRVTAB Entries
Console> (enable)
```

To clear the Kerberos clients mandatory configuration, perform this task in privileged mode:

Task	Command
Clear the Kerberos clients mandatory configuration.	**clear kerberos clients mandatory**

This example shows how to clear the clients mandatory configuration and verify the change:

```
Console> (enable) clear kerberos clients mandatory
Kerberos clients mandatory cleared
Console> (enable) show kerberos
Kerberos Local Realm not configured
Kerberos server entries:

Kerberos Domain<->Realm entries:

Kerberos Clients NOT Mandatory
Kerberos Credentials Forwarding Disabled
Kerberos Pre Authentication Method set to None
Kerberos config key:
Kerberos SRVTAB Entries
Console> (enable)
Kerberos server entries:

Kerberos Domain<->Realm entries:

Kerberos Clients Mandatory
Kerberos Credentials Forwarding Disabled
Kerberos Pre Authentication Method set to Encrypted Unix Time Stamp
Kerberos config key:
Kerberos SRVTAB Entries
Console> (enable)
```

Defining and Clearing a Private DES Key

You can define a private DES key for the switch. You can use the private DES key to encrypt the secret key that the switch shares with the KDC so that when the **show kerberos** command is executed, the secret key is not displayed in clear text. The key length should be eight characters or less.

To define a DES key, perform this task in privileged mode:

Task	Command
Define a DES key for the switch.	**set key config-key** *string*

This example shows how to define a DES key and verify the configuration:

```
kerberos> (enable) set key config-key abcd
Kerberos config key set to abcd
kerberos> (enable) show kerberos
Kerberos Local Realm:CISCO.COM
Kerberos server entries:
Realm:CISCO.COM,  Server:170.20.2.1,  Port:750
Realm:CISCO.COM,  Server:172.20.2.1,  Port:750

Kerberos Domain<->Realm entries:
Domain:cisco.com,  Realm:CISCO.COM

Kerberos Clients Mandatory
Kerberos Credentials Forwarding Disabled
Kerberos Pre Authentication Method set to Encrypted Unix Time Stamp
Kerberos config key:abcd
Kerberos SRVTAB Entries
Srvtab Entry 1:host/aspen-niners.cisco.edu@CISCO.EDU 0 933974942 1 1 8 12151><88?=>>3>11
kerberos> (enable)
```

To clear the DES key, perform this task in privileged mode:

Task	Command
Clear a DES key from the switch.	**clear key config-key** *string*

This example shows how to clear the DES key:

```
Console> (enable) clear key config-key
Kerberos config key cleared
Console> (enable)
```

Encrypting a Telnet Session

After a user authenticates to the switch using Kerberos and wants to access another switch or host through Telnet, whether or not this will be a Kerberized Telnet depends on the authentication method that the Telnet server uses. If the Telnet server uses Kerberos for authentication, you can choose to have all the application data packets that are encrypted for the duration of the Telnet session. To encrypt the Telnet session, select the **encrypt kerberos** option in the **telnet** command.

To encrypt a Telnet session, perform this task:

Task	Command
Encrypt a Telnet session.	**telnet encrypt kerberos** *host*

This example shows how to configure a Telnet session for Kerberos authentication and encryption:

```
Console> (enable) telnet encrypt kerberos
```

Displaying and Clearing Kerberos Configurations

Use these commands to display and clear the Kerberos configurations on the switch:

- **show kerberos**
- **show kerberos creds**
- **clear kerberos creds**

To display the Kerberos configuration, perform this task in privileged mode:

Task	Command
Display the Kerberos configuration.	**show kerberos**

This example shows how to display the Kerberos configuration:

```
kerberos> (enable) show kerberos
Kerberos Local Realm:CISCO.COM
Kerberos server entries:
Realm:CISCO.COM,  Server:187.0.2.1,  Port:750
Realm:CISCO.COM,  Server:187.20.2.1,  Port:750

Kerberos Domain<->Realm entries:
Domain:cisco.com,  Realm:CISCO.COM
Kerberos Clients NOT Mandatory
Kerberos Credentials Forwarding Enabled
Kerberos Pre Authentication Method set to None
Kerberos config key:
Kerberos SRVTAB Entries
Srvtab Entry 1:host/niners.cisco.com@CISCO.COM 0 932423923 1 1 8 03;;5>00>50;0=0=0
Srvtab Entry 2:host/niners.cisco.edu@CISCO.EDU 0 933974942 1 1 8 00?58:127:223=:;9
kerberos> (enable)
```

To display the Kerberos credentials, perform this task in privileged mode:

Task	Command
Display the Kerberos credentials.	**show kerberos creds**

This example shows how to display the Kerberos credentials:

```
Console> (enable) show kerberos creds
No Kerberos credentials.
Console> (enable)
```

To clear all the Kerberos credentials, perform this task in privileged mode:

Task	Command
Clear all the credentials.	**clear kerberos creds**

This example shows how to clear all the Kerberos credentials from the switch:

```
Console> (enable) clear kerberos creds
Console> (enable)
```

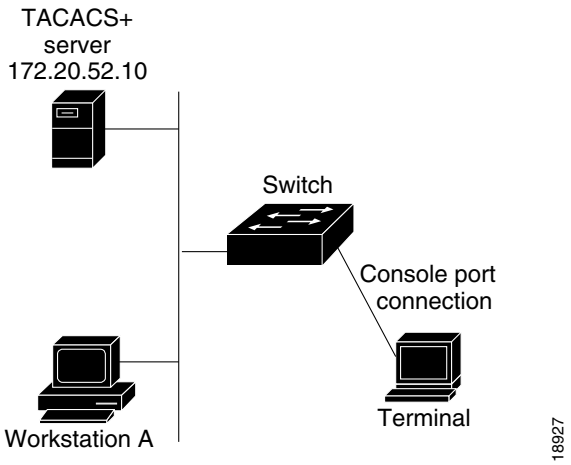
Authentication Example

Figure 39-3 shows a simple network topology using TACACS+.

In this example, TACACS+ authentication is enabled and local authentication is disabled for both the login and enable access to the switch for all Telnet connections. When Workstation A attempts to connect to the switch, the user is challenged for a TACACS+ username and password.

However, only local authentication is enabled for both the login and enable access on the console port. Any user with access to the directly connected terminal can access the switch using the login and enable passwords.

Figure 39-3 TACACS+ Example Network Topology



This example shows how to configure the switch so that TACACS+ authentication is enabled for Telnet connections, local authentication is enabled for the console connections, and a TACACS+ encryption key is specified:

```
Console> (enable) show tacacs
Tacacs key:
Tacacs login attempts: 3
Tacacs timeout: 5 seconds
Tacacs direct request: disabled

Tacacs-Server                              Status
-----------------------------------------  -------
```

```
Console> (enable) set tacacs server 172.20.52.10
172.20.52.10 added to TACACS server table as primary server.
Console> (enable) set tacacs key tintin_et_milou
The tacacs key has been set to tintin_et_milou.
Console> (enable) set authentication login tacacs enable telnet
tacacs login authentication set to enable for telnet session.
Console> (enable) set authentication enable tacacs enable telnet
tacacs enable authentication set to enable for telnet session.
Console> (enable) set authentication login local disable telnet
local login authentication set to disable for telnet session.
Console> (enable) set authentication enable local disable telnet
local enable authentication set to disable for telnet session.
Console> (enable) show tacacs
Tacacs key: tintin_et_milou
Tacacs login attempts: 3
Tacacs timeout: 5 seconds
Tacacs direct request: disabled

Tacacs-Server                               Status
------------------------------------------  -------
172.20.52.10                                primary
Console> (enable)
```

Understanding How Authorization Works

These sections describe how authorization works:

Authorization Overview

Catalyst 6500 series switches support TACACS+ and RADIUS authorization. Authorization limits access to specified users using a dynamically applied access list (or user profile) that is based on the username and password pair. The access list resides on the host running the TACACS+ or RADIUS server. The server responds to the user password information with an access list number that causes the specific list to be applied.

Authorization Events

You can enable authorization for the following:

- Commands—When you enable authorization for commands, the user must supply a valid username and password pair to execute certain commands. You can require authorization for all commands or for configuration (enable mode) commands only. When a user issues a command, the authorization server receives the command and user information and compares it against an access list. If the user is authorized to issue that command, the command is executed; otherwise, the command is not executed.
- EXEC mode (normal login)—When authorization is enabled for EXEC mode, the user must supply a valid username and password pair to gain access to EXEC mode. Authorization is required only if you have enabled the authorization feature.
- Enable mode (privileged login)—When authorization is enabled for enable mode, the user must supply a valid username and password pair to gain access to enable mode. Authorization is required only if you have enabled authorization for enable mode.

TACACS+ Primary Options and Fallback Options

You can specify the primary options and the fallback options that are used in the authorization process. The available options and fallback options include the following:

- **tacacs+**—If you have been authenticated, and there is no response from the TACACS+ server, then authorization will succeed immediately.
- **deny**—Deny is strictly a fallback option. Authorization will fail if the TACACS+ server fails to respond. This is the default behavior.
- **if-authenticated**—If you have been authenticated, and there is no response from the TACACS+ server, then authorization will succeed immediately.
- **none**—Authorization will succeed if the TACACS+ server does not respond.

TACACS+ Command Authorization

You can require authorization for all commands or for configuration (enable mode) commands only. The configuration commands include the following:

- **copy**
- **clear**
- **commit**
- **configure**
- **delete**
- **download**
- **format**
- **reload**
- **rollback**
- **session**
- **set**

- **squeeze**
- **switch**
- **undelete**

The following TACACS+ authorization process occurs for every command that you enter:

- If you have disabled the command authorization feature, the TACACS+ server will allow you to execute any command on the switch.
- If you have enabled authorization for configuration commands only, the switch will verify that the argument string matches one of the commands listed in this section. If there is no match, the switch completes the command. If there is a match, the switch forwards the command to the NAS for authorization.
- If you have enabled authorization for all commands, the switch forwards the command to the NAS for authorization.

RADIUS Authorization

RADUIS has limited authorization. There is one attribute, Service-Type, in the authentication protocol that provides authorization information. This attribute is part of the user-profile.

When you log in using RADIUS authentication and you do not have Administrative/Shell (6) Service-Type access, the network access server (NAS) authenticates you and logs you in to the EXEC mode. If you have Administrative/Shell (6) Service-Type access, the NAS authenticates you and logs you in to the privileged mode.

Configuring Authorization on the Switch

These sections describe how to configure authorization:

- TACACS+ Authorization Default Configuration, page 39-46
- TACACS+ Authorization Configuration Guidelines, page 39-47
- Configuring TACACS+ Authorization, page 39-47
- Configuring RADIUS Authorization, page 39-50

TACACS+ Authorization Default Configuration

Table 39-3 shows the TACACS+ default authorization configuration.

Table 39-3 Default Authorization Configuration

Feature	Default Value
TACACS+ login authorization (console and Telnet)	Disabled
TACACS+ EXEC authorization (console and Telnet)	Disabled
TACACS+ enable authorization (console and Telnet)	Disabled
TACACS+ commands authorization (console and Telnet)	Disabled

TACACS+ Authorization Configuration Guidelines

This section describes the guidelines for configuring TACACS+ authorization on the switch:

- TACACS+ authorization is disabled by default.
- Authorization configuration applies to console connections, Telnet connections, or both types of connections.
- You must specify the mode, option, fallback option, and connection type when enabling authorization.
- Configure the RADIUS and TACACS+ servers before enabling authorization. See the "Specifying TACACS+ Servers" section on page 39-19 or the "Specifying RADIUS Servers" section on page 39-26 for more information on the server setup.
- Configure the RADIUS and TACACS+ keys to encrypt the protocol packets before enabling authorization. See the "Specifying the TACACS+ Key" section on page 39-21 or the "Specifying the RADIUS Key" section on page 39-26 for more information on the key setup.

Configuring TACACS+ Authorization

These sections describe how to configure TACACS+ authorization on the switch:

Enabling TACACS+ Authorization

To enable TACACS+ authorization on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Enable authorization for normal mode. Enter the **console** or **telnet** keyword if you want to enable authorization only for the console port or Telnet connection attempts. Enter the **both** keyword to enable authorization for both console port and Telnet connection attempts.	**set authorization exec enable** {*option*}{*fallbackoption*} [**console** | **telnet** | **both**]
Step 2	Enable authorization for enable mode. Enter the **console** or **telnet** keyword if you want to enable authorization only for the console port or Telnet connection attempts. Enter the **both** keyword to enable authorization for both console port and Telnet connection attempts.	**set authorization enable enable** {*option*} {*fallbackoption*} [**console** | **telnet** | **both**]
Step 3	Enable authorization of the configuration commands. Enter the **console** or **telnet** keyword if you want to enable authorization only for the console port or Telnet connection attempts. Enter the **both** keyword to enable authorization for both console port and Telnet connection attempts.	**set authorization commands enable** {**config** | **all**} {*option*}{*fallbackoption*} [**console** | **telnet** | **both**]
Step 4	Verify the TACACS+ authorization configuration.	**show authorization**

This example shows how to enable TACACS+ EXEC mode authorization for both console and Telnet connections. Authorization is configured with the **tacacs+** option. The fallback option is **deny**.

```
Console> (enable) set authorization exec enable tacacs+ deny both
Successfully enabled enable authorization.
Console>
```

This example shows how to enable TACACS+ enable mode authorization for both console and Telnet connections. Authorization is configured with the **tacacs+** option. The fallback option is **deny**.

```
Console> (enable) set authorization enable enable tacacs+ deny both
Successfully enabled enable authorization.
Console>
```

This example shows how to enable TACACS+ command authorization for both console and Telnet connections. Authorization is configured with the **tacacs+** option. The fallback option is **deny**.

```
Console> (enable) set authorization commands enable config tacacs+ deny both
Successfully enabled commands authorization.
Console> (enable)
```

This example shows how to verify the configuration:

```
Console> (enable) show authorization
Telnet:
-------
            Primary   Fallback
            -------   --------
exec:       tacacs+    deny
enable:     tacacs+    deny
commands:
 config:    tacacs+    deny
 all:       -         -

Console:
--------
            Primary   Fallback
            -------   --------
exec:       tacacs+    deny
enable:     tacacs+    deny
commands:
 config:    tacacs+    deny
 all:       -         -
Console> (enable)
```

Disabling TACACS+ Authorization

To disable TACACS+ authorization on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Disable authorization for normal mode. Enter the **console** or **telnet** keyword if you want to disable authorization only for the console port or Telnet connection attempts. Enter the **both** keyword to disable authorization for both console port and Telnet connection attempts.	**set authorization exec disable** [**console** | **telnet** | **both**]
Step 2	Disable authorization for enable mode. Enter the **console** or **telnet** keyword if you want to disable authorization only for the console port or Telnet connection attempts. Enter the **both** keyword to disable authorization for both console port and Telnet connection attempts.	**set authorization enable disable** [**console** | **telnet** | **both**]
Step 3	Disable authorization of configuration commands. Enter the **console** or **telnet** keyword if you want to disable authorization only for the console port or Telnet connection attempts. Enter the **both** keyword to disable authorization for both console port and Telnet connection attempts.	**set authorization commands disable** [**console** | **telnet** | **both**]
Step 4	Verify the TACACS+ authorization configuration.	**show authorization**

This example shows how to disable TACACS+ EXEC mode authorization for both console and Telnet connections and verify the configuration:

```
Console> (enable) set authorization exec disable both
Successfully disabled enable authorization.
Console> (enable)
```

This example shows how to disable TACACS+ enable mode authorization for both console and Telnet connections and verify the configuration:

```
Console> (enable) set authorization enable disable both
Successfully disabled enable authorization.
Console> (enable)
```

This example shows how to disable TACACS+ command authorization for both console and Telnet connections and verify the configuration:

```
Console> (enable) set authorization commands disable both
Successfully disabled commands authorization.
Console> (enable)
```

This example shows how to verify the configuration:

```
Console> (enable) show authorization

Telnet:
-------
            Primary   Fallback
            -------   --------
exec:       tacacs+    deny
enable:     tacacs+    deny
commands:
 config:    tacacs+    deny
 all:       -         -

Console:
--------
            Primary   Fallback
            -------   --------
exec:       tacacs+    deny
enable:     tacacs+    deny
commands:
 config:    tacacs+    deny
 all:       -         -
Console> (enable)
```

Configuring RADIUS Authorization

These sections describe how to configure RADIUS authorization on the switch:

- Enabling RADIUS Authorization, page 39-50
- Disabling RADIUS Authorization, page 39-50

Enabling RADIUS Authorization

To enable RADIUS authorization and authentication on the switch, perform these steps in privileged mode:

Step 1 Enter the **set authentication login radius enable** command in privileged mode. This command enables both RADIUS authentication and authorization.

Step 2 Set the Service-Type (RADIUS attribute 6) for the user to Admistrative (that is, a value of 6) in the RADIUS server to launch the user into enable mode in the RADIUS server. If the service-type is set for anything other than 6-administrative (for example, 1-login, 7-shell, or 2-framed), you will be at the switch EXEC prompt, not the enable prompt.

Disabling RADIUS Authorization

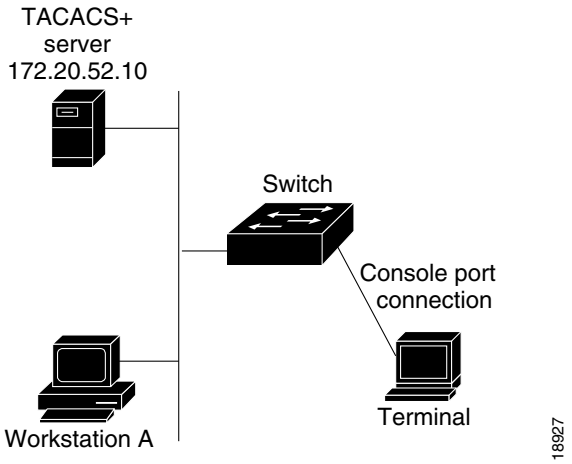
Enter the **set authentication login radius disable** command in privileged mode to disable RADIUS authorization.

Authorization Example

Figure 39-4 shows a simple network topology using TACACS+.

When Workstation A initiates a command on the switch, the switch registers a request with the TACACS+ daemon. The TACACS+ daemon determines if the user is authorized to use the feature and sends a response either executing the command or denying access.

Figure 39-4 TACACS+ Example Network Topology



In this example, TACACS+ authorization is enabled for enable mode access and for the configuration commands to be entered on the switch over the Telnet and console connections:

```
Console> (enable) set authorization enable enable tacacs+ deny both
Successfully enabled enable authorization.
Console> (enable) set authorization commands enable config tacacs+ deny both
Successfully enabled commands authorization.
Console> (enable) show authorization
Telnet:
-------
            Primary   Fallback
            -------   --------
exec:       tacacs+    deny
enable:     tacacs+    deny
commands:
 config:    tacacs+    deny
 all:       -          -

Console:
--------
            Primary   Fallback
            -------   --------
exec:       tacacs+    deny
enable:     tacacs+    deny
commands:
 config:    tacacs+    deny
 all:       -          -
Console> (enable)
```

Understanding How Accounting Works

These sections describe how the different accounting methods work:

Accounting Overview

You can configure these accounting methods to monitor access to the switch:

- TACACS+ accounting
- RADIUS accounting

Accounting allows you to track user activity to a specified host, suspicious connection attempts in the network, and unauthorized changes to the NAS configuration itself. The accounting information is sent to the accounting server where it is saved as a record. Accounting information typically consists of the user's action and the duration for which the action lasted. You can use accounting for security, billing, and resource allocation purposes.

The accounting protocol operates in a client-server model using TCP for transport. The NAS acts as the client and the accounting server acts as the daemon. The NAS sends accounting information to the server. The server, after successfully processing the information, sends a response to the NAS, acknowledging the request. All transactions between the NAS and server are authenticated using a key.

Once accounting has been enabled and an accountable event occurs on the system, the accounting information is gathered dynamically in memory. When the event ends, an accounting record is created and sent to the NAS, and then the system deletes the record from memory. The amount of memory that is used by the NAS for accounting varies depending on the number of concurrent accountable events.

Accounting Events

You can configure accounting for these event types:

- EXEC mode accounting—Provides information about user EXEC sessions (normal login sessions) on the NAS (includes the duration of the EXEC session but does not include the traffic statistics).
- Connect accounting—Provides information about all the outbound connections from the NAS (such as Telnet, rlogin).

Note If you get a connection immediately upon login and then your connection terminates, the EXEC and connect events overlap and have almost identical start and stop times.

- System accounting—Provides information on the system events that are not related to users (includes system reset, system boot, and user configuration of accounting).
- Command accounting—Sends a record for each command that is issued by the user. This feature permits the audit trail information to be gathered.

Specifying When to Create Accounting Records

You configure the switch to gather accounting information to create records. When you configure accounting (using the **set accounting commands**), the switch can generate two types of records:

- Start records—Include partial information of the event (when the event started, type of service, and traffic statistics).
- Stop records—Include complete information of the event (when the event started, its duration, type of service, and traffic statistics).

The accounting records are created and sent to the server at two events:

- Start-stop—Records are sent at both the start and stop of an action if the action has duration. If the NAS fails to send the accounting record at the start of the action, it still allows you to proceed with the action.
- Stop-only—Records are sent only at the termination of the event. Commands are assumed to have zero duration, so only stop records are generated for command accounting. No users are associated with system events; therefore, the **start-stop** option in the **set accounting system** command is ignored for system events.

Note The stop records include complete information of the event (when the event started, its duration, and traffic statistics). However, you might want redundancy and may monitor both the start and stop records of the events occurring on the NAS.

Specifying RADIUS Servers

To specify one or more RADIUS servers, perform this task in privileged mode:

	Task	Command
Step 1	Specify the IP address of up to three RADIUS servers. Specify the primary server using the **primary** keyword. Optionally, specify the destination UDP port to use on the server.	**set radius server** *ip_addr* [**acct-port** *port*] [**primary**]
Step 2	Verify the RADIUS server configuration.	**show radius**

This example shows how to specify a RADIUS server and verify the configuration:

```
Console> (enable) set radius server 172.20.52.3
172.20.52.3 with auth-port 1812 added to radius server table as primary server.
Console> (enable) show radius

Login Authentication:  Console Session   Telnet Session
---------------------  ----------------  ----------------
tacacs                 disabled          disabled
radius                 disabled          disabled
local                  enabled(primary)  enabled(primary)

Enable Authentication: Console Session   Telnet Session
---------------------- ----------------- ----------------
tacacs                 disabled          disabled
radius                 disabled          disabled
local                  enabled(primary)  enabled(primary)

Radius Deadtime:              0 minutes
Radius Key:
Radius Retransmit:            2
Radius Timeout:               5 seconds

Radius-Server                 Status   Auth-port
----------------------------- -------  ------------
172.20.52.3                   primary  1812
Console> (enable)
```

Updating the Server

You can configure the switch to send accounting information to the TACACS+ server. There are two options:

- Newinfo—Sends the accounting information to the server only when new accounting information becomes available.
- Periodic—Sends the accounting update records at regular intervals. This option could be used to keep up-to-date connection and session information even if the NAS restarts and loses the initial start time. You must set a time lapse between periodic updates. Valid intervals are from 1 to 71,582 minutes.

Suppressing Accounting

You can configure the system to suppress accounting when an unknown user with no username accesses the switch by using the **set accounting suppress null-username enable** command.

Note RADIUS and TACACS+ accounting are the same, except that RADIUS does not do command accounting, periodic updates, or allow null-username suppression.

Configuring Accounting on the Switch

These sections describe how to configure accounting for both TACACS+ and RADIUS:

Accounting Default Configuration

Table 39-4 shows the accounting default configuration.

Table 39-4 Accounting Default Configuration

Feature	Default Value
Accounting	Disabled
Accounting events (EXEC, system, commands, and connect)	Disabled
Accounting records	Stop-only

Accounting Configuration Guidelines

This section describes the guidelines for configuring accounting on the switch:

- Configure the RADIUS and TACACS+ servers before enabling accounting. See the "Specifying TACACS+ Servers" section on page 39-19 or the "Specifying RADIUS Servers" section on page 39-26 for more information on the server setup.
- Configure the RADIUS and TACACS+ keys to encrypt the protocol packets before enabling accounting. See the "Specifying the TACACS+ Key" section on page 39-21 or the "Specifying the RADIUS Key" section on page 39-26 for more information on the key setup.

Note The amount of DRAM that is allocated for one accounting event is approximately 500 bytes. The total amount of DRAM that is used by accounting depends on the number of *concurrent* accountable events in the system.

Configuring Accounting

These sections describe how to configure RADIUS and TACACS+ accounting on the switch:

Enabling Accounting

To enable accounting on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Enable accounting for connection events.	**set accounting connect enable** {**start-stop** | **stop-only**} {**tacacs+** | **radius**}
Step 2	Enable accounting for EXEC mode.	**set accounting exec enable** {**start-stop** | **stop-only**} {**tacacs+** | **radius**}
Step 3	Enable accounting for system events.	**set accounting system enable** {**start-stop** | **stop-only**} {**tacacs+** | **radius**}
Step 4	Enable accounting of configuration commands.	**set accounting commands enable** {**config** | **all**} {**stop-only**} **tacacs+**
Step 5	Enable suppression of information for unknown users.	**set accounting suppress null-username enable**
Step 6	Configure accounting to be updated as new information is available.	**set accounting update** {**new-info** | {**periodic** [*interval*]}}
Step 7	Verify the accounting configuration.	**show accounting**

This example shows how to enable the stop-only TACACS+ accounting events:

```
Console> (enable) set accounting connect enable stop-only tacacs+
Accounting set to enable for connect events in stop-only mode.
Console> (enable)

Console> (enable) set accounting exec enable stop-only tacacs+
Accounting set to enable for exec events in stop-only mode.
Console> (enable)

Console> (enable) set accounting system enable stop-only tacacs+
Accounting set to enable for system events in stop-only mode.
Console> (enable)

Console> (enable) set accounting commands enable all stop-only tacacs+
Accounting set to enable for commands-all events in stop-only mode.
Console> (enable)
```

This example shows how to suppress accounting of unknown users:

```
Console> (enable) set accounting suppress null-username enable
Accounting will be suppressed for user with no username.
Console> (enable)
```

This example shows how to update the server periodically:

```
Console> (enable) set accounting update periodic 120
Accounting updates will be periodic at 120 minute intervals.
Console> (enable)
```

This example shows how to verify the configuration:

```
Console> (enable) show accounting
Event     Method  Mode
-----     ------- ----
exec:     tacacs+ stop-only
connect:  tacacs+ stop-only
system:   tacacs+ stop-only
commands:
config:   -       -
all:      tacacs+ stop-only
TACACS+ Suppress for no username: enabled
Update Frequency: periodic, Interval = 120

Accounting information:
-----------------------
Active Accounted actions on tty0, User (null) Priv 0
Active Accounted actions on tty288091924, User (null) Priv 0
Overall Accounting Traffic:
          Starts   Stops  Active
          -----    -----  ------
Exec           0       0       0
Connect        0       0       0
Command        0       0       0
System         1       0       0
Console> (enable)
```

Disabling Accounting

To disable RADIUS accounting on the switch, perform this task in privileged mode:

	Task	Command
Step 1	Disable accounting for connection events.	**set accounting connect disable**
Step 2	Disable accounting for EXEC mode.	**set accounting exec disable**
Step 3	Disable accounting for system events.	**set accounting system disable**
Step 4	Disable accounting of configuration commands.	**set accounting commands disable**
Step 5	Disable suppression of information for unknown users.	**set accounting suppress null-username disable**
Step 6	Verify the accounting configuration.	**show accounting**

This example shows how to disable stop-only accounting:

```
Console> (enable) set accounting connect disable
Accounting set to disable for connect events.
Console> (enable)

Console> (enable) set accounting exec disable
Accounting set to disable for exec events.
Console> (enable)

Console> (enable) set accounting system disable
Accounting set to disable for system events.
Console> (enable)

Console> (enable) set accounting commands disable
Accounting set to disable for commands-all events.
Console> (enable)
```

This example shows how to disable suppression of unknown users:

```
Console> (enable) set accounting suppress null-username disable
Accounting will be not be suppressed for user with no username.
Console> (enable)
```

This example shows how to verify the configuration:

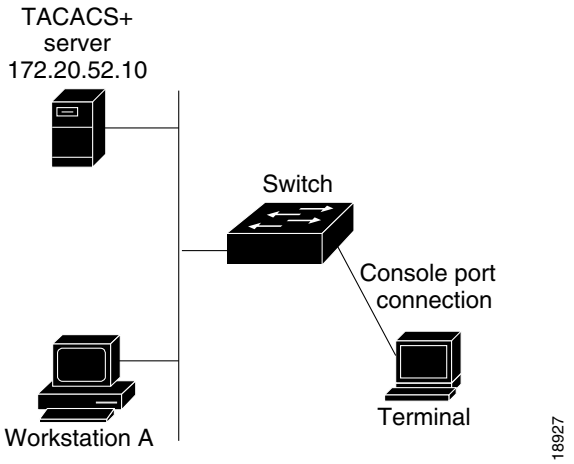
```
Console> (enable) show accounting
Event     Method  Mode
-----     ------- ----
exec:     -       -
connect:  -       -
system:   -       -
commands:
config:   -       -
all:      -       -

TACACS+ Suppress for no username: disabled
Update Frequency: new-info

Accounting information:
-----------------------
Active Accounted actions on tty0, User (null) Priv 0
Active Accounted actions on tty288091924, User (null) Priv 0
Overall Accounting Traffic:
          Starts   Stops  Active
          -----    -----  ------
Exec           0       0       0
Connect        0       0       0
Command        0       0       0
System         1       2       0
Console> (enable)
```

Accounting Example

Figure 39-5 shows a simple network topology using TACACS+.

When Workstation A initiates an accountable event on the switch, the switch gathers the event information and forwards the information to the server at the conclusion of the event. The accounting information is gathered at the conclusion of the event. Accounting is suspended for unknown users, and the system is updated every 120 minutes.

Figure 39-5 TACACS+ Example Network Topology



In this example, TACACS+ accounting is enabled for connection, EXEC, system, and all command accounting:

```
Console> (enable) set accounting connect enable stop-only tacacs+
Accounting set to enable for connect events in stop-only mode.
Console> (enable) set accounting exec enable stop-only tacacs+
Accounting set to enable for exec events in stop-only mode.
Console> (enable) set accounting commands enable all stop-only tacacs+
Accounting set to enable for commands-all events in stop-only mode.
Console> (enable) set accounting update periodic 120
Accounting updates will be periodic at 120 minute intervals.
Console> (enable) show accounting
Event     Method  Mode
-----     ------- ----
exec:     tacacs+ stop-only
connect:  tacacs+ stop-only
system:   tacacs+ stop-only
commands:
config:   -       -
all:      tacacs+ stop-only

TACACS+ Suppress for no username: enabled
Update Frequency: periodic, Interval = 120

Accounting information:
-----------------------
Active Accounted actions on tty0, User (null) Priv 0
Active Accounted actions on tty288091924, User (null) Priv 0
Overall Accounting Traffic:
          Starts   Stops  Active
          -----    -----  ------
Exec           0       0       0
Connect        0       0       0
Command        0       0       0
System         1       0       0
Console> (enable)
```

CHAPTER 40

Configuring 802.1X Authentication

This chapter describes how to configure IEEE 802.1X authentication on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

Note For information on configuring MAC address authentication bypass, see Chapter 41, "Configuring MAC Authentication Bypass."

Note For information on using port security to block input to an Ethernet, Fast Ethernet, or Gigabit Ethernet port when the MAC address of the station attempting to access the port is different from any of the MAC addresses that are specified for that port, see Chapter 38, "Configuring Port Security." That chapter also provides information on using port security to filter the traffic that is destined to or received from a specific host that is based on the host MAC address.

Note For information on configuring authentication, authorization, and accounting (AAA) to monitor and control access to the command-line interface (CLI) on the Catalyst 6500 series switches, see Chapter 39, "Configuring the Switch Access Using AAA."

Note For information on configuring Network Admission Control, see Chapter 44, "Configuring Network Admission Control."

Note For information on configuring network admission control, see Chapter 44, "Configuring Network Admission Control."

This chapter consists of these sections:

Understanding How 802.1X Authentication Works

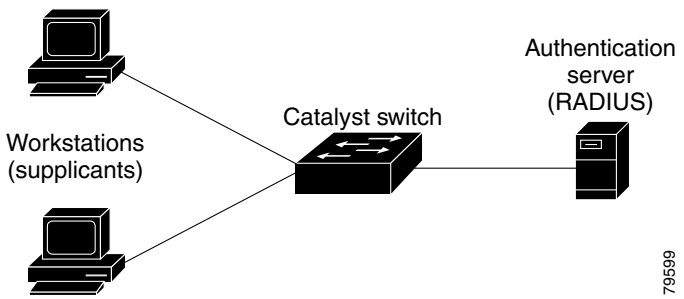
802.1X defines a client-server-based access control and authentication protocol that restricts unauthorized devices from connecting to a LAN through publicly accessible ports. 802.1X controls network access by creating two distinct virtual access points at each port. One access point is an uncontrolled port; the other is a controlled port. All traffic through the single port is available to both access points. 802.1X authenticates each user device that is connected to a switch port and assigns the port to a VLAN before making available any services that are offered by the switch or the LAN. Until the device is authenticated, 802.1X access control allows only Extensible Authentication Protocol over LAN (EAPOL) traffic through the port to which the device is connected. After authentication is successful, normal traffic can pass through the port. You can restrict the traffic in both directions, or you can restrict just the incoming traffic.

These sections provide the following information:

Device Roles

With 802.1X port-based authentication, the devices in the network have specific roles. (See Figure 40-1.)

Figure 40-1 802.1X Device Roles



- *Supplicant*—Requests access to the LAN and switch services and responds to requests from the switch. The workstation must be running 802.1X-compliant software.

  Note 802.1X uses the term *supplicant* for *client* or *host*. In this publication, we use *host* instead of *supplicant* because *host* is used in the Catalyst 6500 series CLI syntax.

- *Authentication server*—Performs the actual authentication of the host. The authentication server validates the identity of the host and notifies the switch if the host is authorized to access the LAN and switch services. Because the switch acts as the proxy, the authentication service is transparent to the host. In this release, the Remote Authentication Dial-In User Service (RADIUS) security system with Extensible Authentication Protocol (EAP) extensions is the only supported authentication server; it is available in Cisco Secure Access Control Server version 3.0. RADIUS operates in a client/server model in which secure authentication information is exchanged between the RADIUS server and one or more RADIUS clients.
- *Switch*—Controls the physical access to the network based on the authentication status of the host. The switch acts as an intermediary (proxy) between the host and the authentication server, requesting identity information from the host, verifying that information with the authentication server, and relaying a response to the host. The switch interacts with the RADIUS client. The RADIUS client encapsulates and decapsulates the EAP frames and interacts with the authentication server.

  When the switch receives the Extensible Authentication Protocol over LAN (EAPOL) frames and relays them to the authentication server, the Ethernet header is stripped and the remaining EAP frame is reencapsulated in the RADIUS format. The EAP frames are not modified or examined during encapsulation, and the authentication server must support EAP within the native frame format. When the switch receives the frames from the authentication server, the server's frame header is removed, leaving the EAP frame, which is then encapsulated for Ethernet and sent to the host.

Authentication Initiation and Message Exchange

The switch or the host can initiate authentication. If you enable authentication on a port by using the **set port dot1x** *mod/port* **port-control auto** command, the switch must initiate authentication when it determines that the port link state transitions from down to up. The switch sends an EAP-request/identity frame to the host to request its identity (typically, the switch sends an initial identity/request frame that is followed by one or more requests for authentication information). When the host receives the frame, it sends an EAP-response/identity frame.

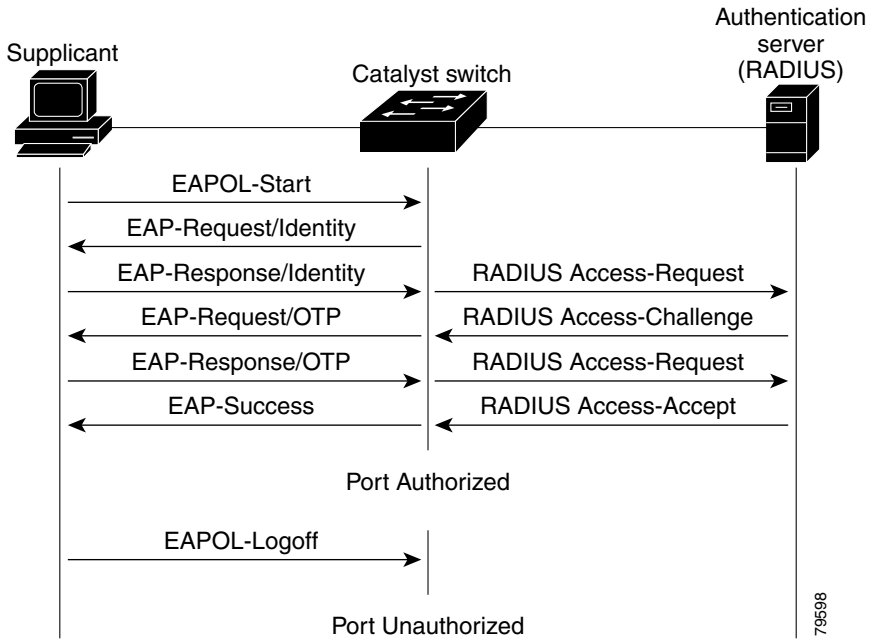
During bootup, if the host does not receive an EAP-request/identity frame from the switch, the host can initiate authentication by sending an EAPOL-start frame that prompts the switch to request the host's identity.

Note If 802.1X is not enabled or supported on the network access device, any of the EAPOL frames from the host are dropped. If the host does not receive an EAP-request/identity frame after three attempts to start authentication, the host transmits the frames as if the port is in the authorized state. A port that is in the authorized state means that the host has been successfully authenticated. For more information, see the "Ports in Authorized and Unauthorized States" section on page 40-4.

When the host supplies its identity, the switch acts as the intermediary, passing the EAP frames between the host and the authentication server until authentication succeeds or fails. If the authentication succeeds, the switch port becomes authorized. For more information, see the "Ports in Authorized and Unauthorized States" section on page 40-4.

The specific exchange of EAP frames depends on the authentication method that is being used. Figure 40-2 shows a message exchange that is initiated by the host using the One-Time-Password (OTP) authentication method with a RADIUS server.

Figure 40-2 Message Exchange



Ports in Authorized and Unauthorized States

The switch port state determines if the host is granted access to the network. The port starts in the *unauthorized* state. In this state, the port disallows all the ingress and egress traffic except for the 802.1X protocol packets. When a host is successfully authenticated, the port transitions to the *authorized* state, which allows all traffic for the host to flow normally.

If a host that does not support 802.1X is connected to an unauthorized 802.1X port, the switch requests the host's identity. In this situation, the host does not respond to the request, the port remains in the unauthorized state, and the host is not granted access to the network.

When an 802.1X-enabled host connects to a port that is not running the 802.1X protocol, the host initiates the authentication process by sending the EAPOL-start frame. When no response is received, the host sends the request for a fixed number of times. Because no response is received, the host begins sending frames as if the port is in the authorized state.

You control the port authorization state by using the **set port dot1x** *mod/port* **port-control** command and these keywords:

- **force-authorized**—Disables 802.1X authentication and causes the port to transition to the authorized state without any authentication exchange required. The port transmits and receives normal traffic without 802.1X-based authentication of the host. This setting is the default.
- **force-unauthorized**—Causes the port to remain in the unauthorized state, ignoring all attempts by the host to authenticate. The switch cannot provide authentication services to the host through the interface.
- **auto**—Enables 802.1X authentication and causes the port to begin in the unauthorized state, allowing only the EAPOL frames to be sent and received through the port. The authentication process begins when the link state of the port transitions from down to up or when an EAPOL-start frame is received. The switch requests the identity of the host and begins relaying the authentication messages between the host and the authentication server. Each host attempting to access the network is uniquely identified by the switch by using the host's MAC address.

If the host is successfully authenticated (receives an Accept frame from the authentication server), the port state changes to authorized, and all frames from the authenticated host are allowed through the port. If the authentication fails, the port remains in the unauthorized state, but authentication can be retried. If the switch cannot reach the authentication server, it can retransmit the request. If no response is received from the server after the specified number of attempts, authentication fails, and network access is not granted.

When a host logs off, the server sends an EAPOL-logoff message, causing the switch port to transition to the unauthorized state.

If the link state of a port transitions from up to down, or if an EAPOL-logoff frame is received, the port returns to the unauthorized state.

Table 40-1 defines the 802.1X terms.

Table 40-1 802.1X Terminology

Term	Definition
Authenticator PAE[1]	(Referred to as the "authenticator") entity at one end of a point-to-point LAN segment that enforces host authentication. The authenticator is independent of the actual authentication method and functions only as a pass-through for the authentication exchange. It communicates with the host, submits the information from the host to the authentication server, and authorizes the host when instructed to do so by the authentication server.
Authentication server	Entity that provides the authentication service for the authenticator PAE. It checks the credentials of the host PAE and then notifies its client, the authenticator PAE, whether the host PAE is authorized to access the LAN/switch services.
Authorized state	Status of the port after the host PAE is authorized.
Both	Bidirectional flow control, incoming and outgoing, at an unauthorized switch port.
Controlled port	Secured access point.
EAP	Extensible Authentication Protocol.
EAPOL[2]	Encapsulated EAP messages that can be handled directly by a LAN MAC service.

Table 40-1 802.1X Terminology (continued)

Term	Definition
In	Flow control only on incoming frames in an unauthorized switch port.
Port	Single point of attachment to the LAN infrastructure (for example, MAC bridge ports).
PAE	Port access entity protocol object that is associated with a specific system port.
PDU	Protocol data unit.
RADIUS	Remote Access Dial-In User Service.
Supplicant[3] PAE	Entity that requests access to the LAN/switch services and responds to the information requests from the authenticator.
Unauthorized state	Status of the port before the supplicant PAE is authorized.
Uncontrolled port	Unsecured access point that allows the uncontrolled exchange of PDUs.

1. PAE = port access entity
2. EAPOL = Extensible Authorization Protocol over LAN
3. 802.1X uses the term *supplicant* for *client* or *host*. This publication uses *host* instead of *supplicant* because *host* is used in the Catalyst 6500 series CLI syntax.

Authentication Server

The frames that are exchanged between the authenticator and the authentication server are dependent on the authentication mechanism, so they are not defined by 802.1X. You can use other protocols, but we recommend that you use RADIUS for authentication, particularly when the authentication server is located remotely, because RADIUS has extensions that support the encapsulation of EAP frames built into it.

802.1X Parameters Configurable on the Switch

You can configure these 802.1X parameters on the switch:

- Specify Force-Authorized, Force-Unauthorized, or Automatic 802.1X port control
- Specify single authentication, multiple authentication, and multiple host authentication
- Enable or disable system authentication control
- Specify the quiet time interval
- Specify the authenticator to host retransmission time interval
- Specify the back-end authenticator to host retransmission time interval
- Specify the back-end authenticator to authentication server retransmission time interval
- Specify the number of frames that are retransmitted from the back-end authenticator to the host
- Specify the automatic host reauthentication time interval
- Specify the port shutdown timeout period after a security violation
- Enable or disable automatic host reauthentication

Understanding How 802.1X VLAN Assignments Using a RADIUS Server Work

In the supervisor engine software releases prior to software release 7.2(2), once the 802.1X host is authenticated, it joins an NVRAM-configured VLAN. With software release 7.2(2) and later releases, after authentication, an 802.1X host can receive its VLAN assignment from the RADIUS server.

The VLAN assignment feature allows you to restrict users to a specific VLAN. For example, you could put the guest users in a VLAN with limited access to the network.

The 802.1X authenticated ports are assigned to a VLAN based on the username of the host that is connected to the port. This feature works with the RADIUS server that has a database of username-to-VLAN mappings.

After a successful 802.1X authentication of the port, the RADIUS server sends the VLAN in which the user needs to be given access. The 802.1X port behavior with the VLAN assignment feature is as follows:

- At linkup, an 802.1X port is placed in its original NVRAM-configured VLAN.
- After linkup, the port can be put in the RADIUS-supplied VLAN if the RADIUS-supplied VLAN is valid and active in the management domain.
- If the port is currently in a different VLAN, it is moved to the RADIUS-supplied VLAN.
- If the RADIUS-supplied VLAN is not active in the management domain, the port is put in an inactive state.
- If the RADIUS-supplied VLAN is invalid or there is a problem with the port hardware, the port is moved to the 802.1X unauthorized state.
- When you enable the multiple hosts option on an 802.1X port, all the hosts are placed in the same RADIUS-supplied VLAN that is received by the first authenticated user.
- When an 802.1X-configured module goes down, all the Enhanced Address Recognition Logic (EARL) entries are cleared for the 802.1X ports.
- When an 802.1X-configured module comes up, all the 802.1X ports are configured in the NVRAM-configured VLANs.
- When an 802.1X-configured module's configuration is cleared, all the 802.1X ports are moved to the NVRAM-configured VLAN and all the EARL entries for the 802.1X ports are cleared.
- When an 802.1X port moves from an authorized to an unauthorized state, the port is moved to the NVRAM-configured VLAN.

In order for the "802.1X VLAN assignment using a RADIUS server" feature to successfully complete, the RADIUS server must return these three RFC 2868 attributes to the authenticator (the Cisco switch to which the host attaches):

- [64] Tunnel-Type = VLAN
- [65] Tunnel-Medium-Type = 802
- [81] Tunnel-Private-Group-Id = VLAN NAME or VLAN ID (VLAN number)

Attribute [64] must contain the value "VLAN" (type 13). Attribute [65] must contain the value "802" (type 6). Attribute [81] specifies the VLAN name or VLAN ID in which the successfully authenticated 802.1X host is placed.

Understanding How 802.1X Authentication with DHCP Works

The 802.1X authentication support for the Dynamic Host Configuration Protocol (DHCP) allows the DHCP server to assign the IP addresses to the different classes of end users by adding the authenticated user identity into the DHCP discovery process. This feature allows you to secure the IP addresses given to the end users for accounting purposes and to grant the services that are based on the Layer 3 criteria. Once the RADIUS server authenticates the supplicant, the DHCP server keeps an authenticated user identity that is associated with the IP address lease. This authenticated user identity is then added to the DHCP discovery process so that the different addresses can be assigned to the different classes of users.

After the successful 802.1X authentications between the supplicant and the RADIUS server, the switch puts the port in the forwarding state and stores the attributes that it receives from the RADIUS server. These attributes are used to map to an address pool in the DHCP server. Because the switch can act as a DHCP Relay Agent, it can receive the DHCP messages and regenerate those messages for transmission on another interface. When the supplicant does DHCP discovery (following authentication), the DHCP Relay Agent on the supervisor engine receives the packet and adds the stored attributes that it received from the RADIUS server to the DHCP discovery packet and submits the discovery broadcast again. The mapping of user-to-IP address can be on a one-to-one, one-to-many, or many-to-many basis. The one-to-many mapping allows the same user to authenticate through the 802.1X hosts on multiple ports.

Understanding How 802.1X Authentication on Ports Configured for Auxiliary VLAN Traffic Works

You can enable 802.1X on a Multiple VLAN Access Port (MVAP), and you can enable an auxiliary VLAN ID on an 802.1X port.

The ports that are configured for 802.1X authentication and an auxiliary VLAN must be in single-host authentication mode to forward the auxiliary VLAN-tagged packets from an IP phone. Because the IP phones do not have host PAE capability, when the auxiliary VLAN-tagged packets are received on a port that is configured for 802.1X authentication from the IP phone, the packets are forwarded as authorized traffic.

A host PAE that is connected behind an IP phone will be authenticated. Only the traffic from the host PAE behind the IP phone is forwarded after authentication.

Note If a new host PAE is connected to an IP phone that is connected to an 802.1X-enabled auxiliary VLAN port, after removing the old host, the new host PAE will be authenticated. Only the traffic from the new host PAE is forwarded after authentication.

Understanding How 802.1X Authentication for the Guest VLAN Works

This section describes the 802.1X authentication for the guest VLANs.

A guest VLAN enables the non-802.1X capable hosts to access the networks that use 802.1X authentication. You can use the guest VLANs while you are upgrading your system to support the 802.1X authentication.

When you configure a VLAN as an 802.1X guest VLAN, all the non-802.1X capable hosts are put in this VLAN. You can configure any VLAN (except for the private VLANs and RSPAN VLANs) as a guest VLAN. If a port is already forwarding on the guest VLAN and you enable 802.1X support on the network interface of the host, the port is immediately moved out of the guest VLAN and the authenticator waits for authentication to occur.

Note In software release 8.6(1) and later releases, a private VLAN and a secondary VLAN can be configured as the guest VLAN. For more information, see the "Configuring 802.1X Authentication with Private VLANs" section on page 40-41.

Enabling 802.1X authentication on a port starts the 802.1X protocol. If the host fails to respond to the packets from the authenticator within a certain amount of time, the authenticator puts the port in the guest VLAN.

The guest VLANs are supported in both single-authentication mode and multiple-host mode.

Note Contrast the guest VLAN feature with the authentication failure VLAN feature. On a traditional 802.1X port, the switch does not provide access to the network until the supplicant that is connected to the port is authenticated by verifying its identity information with an authentication server. With an authentication failure VLAN, you can configure the authentication failure VLAN on a per-port basis and after three failed 802.1X authentication attempts by the supplicant, the port is moved to the authentication failure VLAN where the supplicant can access the network.

An authentication failure VLAN is independent of the guest VLAN. However, the guest VLAN can be the same VLAN as the authentication failure VLAN. If you do not want to differentiate between the non-802.1X capable hosts and the authentication failed hosts, you may configure both hosts to the same VLAN (either a guest VLAN or an authentication failure VLAN).

For more information, see the "Configuring the Authentication Failure VLAN" section on page 40-38.

Usage Guidelines for 802.1X Authentication with the Guest VLANs on Windows-XP Hosts

This section describes the usage guidelines for configuring 802.1X authentication with the guest VLANs on Windows-XP hosts:

- If a guest VLAN is enabled on a port, that port cannot be configured as a unidirectional port, and conversely, a unidirectional port cannot be configured in a guest VLAN.
- If the host fails to respond to the authenticator, the port remains in the connecting state for 180 seconds. After this time, the login/password window does not appear on the host. The workaround is to have the user unplug and then reconnect the network interface cable.

- The hosts that respond with an incorrect login/password fail authentication are not put in the guest VLAN. The first time that a host fails authentication, the quiet-period timer starts and no activity occurs for the duration of the quiet-period timer. When the quiet-period timer expires, the host is presented with the login/password window. If the host fails authentication for the second time, the quiet-period timer starts again and no activity occurs for the duration of the quiet-period timer. The host is presented with the login/password window a third time. If the host fails the third time, the port is put in the connecting and unauthorized states. The workaround to this problem is to have the user unplug and then reconnect the network interface cable.
- If a host does not respond to the username and password authentication requests from the Authenticator PAE, it is placed in a guest VLAN.

Note The guest VLANs are limited to the local switch and are not propagated through VTP.

Understanding How 802.1X Authentication with Port Security Works

802.1X authentication is compatible with the port security feature (for more information, see Chapter 38, "Configuring Port Security"). If you enable port security for only one MAC address on a specific port, only that MAC address authenticates through a RADIUS server. The users that are connected through all other MAC addresses are denied access. If you enable port security for multiple MAC addresses, each address needs to authenticate through the 802.1X RADIUS server.

Note When 802.1X authentication and port security are enabled on any 802.1X port, the 802.1X authentication takes precedence over the port security on the port. The host is authenticated first and is then secured by port security.

You can enable port security for any 802.1X mode (single-authentication mode, multiple-host mode, or multiple-authentication mode). Only one mode can be enabled on a port at a time. The default port mode is single-authentication mode.

You can disable port security for single-authentication mode and multiple-host mode. You cannot disable port security for multiple-authentication mode.

When 802.1X authentication is enabled on a port that is also enabled for MAC address-based port security, 802.1X authentication does not occur on the port unless the maximum allowable number of MAC addresses has been configured. If you configure fewer addresses than the maximum allowable number of MAC addresses on a port that is also configured for 802.1X single-host mode authentication, the system generates a message asking if you want the configured MAC addresses to be removed. If you answer "yes" to this message, the MAC addresses that you configured for MAC address-based port security are removed and the port is authenticated using 802.1X authentication. If 802.1X authentication is enabled for any other mode, no message is created and the MAC addresses are retained.

In the multiple-authentication mode, all connected hosts are authenticated using 802.1X and secured using port security. 802.1X authenticates the MAC address and then gives the MAC address to port security to secure it. When a MAC address sends an EAPOL logoff packet, the MAC address is cleared from the port security tables.

Understanding How 802.1X Authentication with ARP Traffic Inspection Works

Note This feature is available only with Supervisor Engine 2 with PFC2, Supervisor Engine 720 with PFC3A/PFC3B/PFC3BXL, and Supervisor Engine 32 with PFC3B/PFC3BXL.

ARP traffic inspection allows you to configure a set of order-dependent rules within the security ACL (VACL) framework to prevent ARP table attacks. ARP traffic inspection complements the 802.1X port authentication protocol, which first binds the MAC address of the authenticated client to the port, eliminating the possibility of spoofing additional MAC addresses by adding an IP to MAC address binding for additional spoof proofing.

You can use 802.1X authentication with ARP traffic inspection to provide an additional layer of port and user security by eliminating the possibility of malicious users/hosts corrupting the ARP tables of the other hosts. After a successful 802.1X supplicant authentication, ARP traffic inspection, which binds the supplicant's IP address and MAC address, is invoked and eliminates the spoofing possibility.

ARP is a simple protocol that does not have an authentication mechanism so there is no means to ensure that the ARP requests and replies are genuine. Without an authentication mechanism, a malicious user/host can corrupt the ARP tables of the other hosts on the same VLAN in a Layer 2 network or bridge domain.

For example, user/Host A (the malicious user) can send the unsolicited ARP replies (or the gratuitous ARP packets) to the other hosts on the subnet with the IP address of the default router and the MAC address of Host A.With some earlier operating systems, even if a host already has a static ARP entry for the default router, the newly advertised binding from Host A is learned. If Host A enables IP forwarding and forwards all packets from the "spoofed" hosts to the router and vice versa, then Host A can carry out a man-in-the-middle attack (for example, using the program dsniff) without the spoofed hosts realizing that all of their traffic is being sniffed.

In addition, ARP inspection can drop the packets where the source Ethernet MAC address (in the Ethernet header) does not match the source MAC address in the ARP header. You can enable (or disable) this feature through the CLI by entering the **set security acl arp-inspection match-mac** {**enable** [**drop** [**log**]] | **disable**} command.

To configure ARP traffic inspection, see the "Inspecting ARP Traffic" section on page 15-30.

Default Authentication Configuration

Table 40-2 shows the default 802.1X authentication configuration.

Table 40-2 802.1X Authentication Default Configuration

Feature	Default Value
PAE Capability	Authenticator only
Protocol Version	1
802.1X port control	Force-authorized
802.1X multiple hosts	Disabled
802.1X system authentication control	Enabled
802.1X quiet period time	60 seconds

Table 40-2 802.1X Authentication Default Configuration (continued)

Feature	Default Value
802.1X authenticator to host retransmission time	30 seconds
802.1X back-end authenticator to host retransmission time	30 seconds
802.1X back-end authenticator to authentication server retransmission time	30 seconds
802.1X number of frames that are retransmitted from back-end authenticator to the host	2
802.1X automatic host reauthentication time	3600 seconds
802.1X automatic authenticator reauthentication of the host	Disabled
802.1X shutdown timeout period	300 seconds
802.1X RADIUS accounting	Disabled
802.1X RADIUS VLAN assignment	Enabled
802.1X RADIUS keepalive state	Enabled

Authentication Configuration Guidelines

This section provides the guidelines for configuring 802.1X authentication on the switch:

- 802.1X will work with other protocols, but we recommend that you use RADIUS with a remotely located authentication server.
- 802.1X is supported only on the Ethernet ports.
- Software release 7.5(1) supports two in-band management interfaces, sc0 and sc1. 802.1X authentication always uses the sc0 interface as the identifier for the authenticator when communicating with the RADIUS server. 802.1X authentication is not supported with the sc1 interface.
- You cannot enable 802.1X on a trunk port until you turn off trunking on that port. You cannot enable trunking on an 802.1X port.
- You cannot enable 802.1X on a dynamic port until you turn off dynamic VLAN on that port. You cannot enable dynamic VLAN on an 802.1X port.
- You cannot enable 802.1X on a channeling port until you turn off channeling on that port. You cannot enable channeling on an 802.1X port.
- You cannot enable 802.1X on a switched port analyzer (SPAN) destination port. You cannot configure SPAN destination on an 802.1X port. However, you can configure an 802.1X port as a SPAN source port.
- You cannot set the auxiliary VLAN to **dot1p** or **untagged,** and the auxiliary VLAN should not be equal to the native VLAN on the 802.1X-enabled port.
- You cannot enable the multiple-authentication option on an 802.1X-enabled auxiliary VLAN port. We recommend that you do not enable the multiple-host option on an 802.1X-enabled auxiliary port.
- Do not assign a guest VLAN equal to an auxiliary VLAN because an 802.1X-enabled auxiliary VLAN port will not be put into the guest VLAN if the auxiliary VLAN on the port is the same as the guest VLAN.
- On an 802.1X-enabled port, an administratively configured VLAN cannot be equal to an auxiliary VLAN.

- The private VLANs and 802.1X configurations are mutually exclusive of one another.

Note Software release 8.6(1) and later releases provide support for configuring 802.1X with private VLANs. For more information, see the "Configuring 802.1X Authentication with Private VLANs" section on page 40-41.

- With a PFC3A/PFC3B/PFC3BXL, you can use the **set rate-limit l2port-security** command to enable, disable, or set the 802.1X port security rate limiters globally on the switch. For more information on configuring rate limiting, see the "Configuring Layer 2 PDU Rate Limiting on the Switch" section on page 7-61.

Configuring 802.1X Authentication on the Switch

These sections describe how to configure 802.1X authentication on the switch:

Note For information on using a RADIUS server for VLAN assignment, see the "Understanding How 802.1X VLAN Assignments Using a RADIUS Server Work" section on page 40-7.

Enabling 802.1X Authentication Globally

You must enable 802.1X authentication for the entire system before you can configure it for the individual ports. After you globally enable 802.1X authentication, you can configure the individual ports for 802.1X authentication if the port meets the specific requirements that are required by 802.1X. To enable 802.1X authentication for the individual ports, see the "Enabling 802.1X Authentication for Individual Ports" section on page 40-15.

To enable 802.1X authentication globally, perform this task in privileged mode:

Task	Command
Globally enable 802.1X authentication.	**set dot1x system-auth-control enable**

This example shows how to enable 802.1X authentication globally:

```
Console> (enable) set dot1x system-auth-control enable
dot1x system-auth-control enabled.
```

Disabling 802.1X Authentication Globally

When 802.1X authentication is enabled for the entire system, you can disable it globally. When 802.1X authentication is disabled globally, it is no longer available at any port (even ports that were previously configured for it).

To disable 802.1X authentication globally, perform this task in privileged mode:

Task	Command
Globally disable 802.1X authentication.	**set dot1x system-auth-control disable**

This example shows how to disable 802.1X authentication globally:

```
Console> (enable) set dot1x system-auth-control disable
dot1x system-auth-control disabled.
```

Enabling 802.1X Authentication for Individual Ports

After 802.1X authentication is globally enabled, you must enable 802.1X authentication from the console for the individual ports. To enable 802.1X authentication globally, see the "Enabling 802.1X Authentication Globally" section on page 40-14.

Note You must specify at least one RADIUS server before you can enable 802.1X authentication on the switch. For more information, see Chapter 21, "Configuring the Switch Access Using AAA."

To enable 802.1X authentication for access to the switch, perform this task in privileged mode:

	Task	Command
Step 1	Enable 802.1X control on a specific port.	**set port dot1x** *mod/port* **port-control auto**
Step 2	Verify the 802.1X configuration.	**show port dot1x** *mod/port*

This example shows how to enable 802.1X authentication on port 1 in module 3 and verify the configuration:

```
Console> (enable) set port dot1x 3/1 port-control auto
Port 3/1 dot1x port-control is set to auto.
Trunking disabled for port 3/1 due to Dot1x feature.
Spantree port fast start option enabled for port 3/1.
Console> (enable) show port dot1x 3/1
Port  Auth-State          BEnd-State Port-Control        Port-Status
----- ------------------- ---------- ------------------- -------------
 3/1  connecting          idle       auto                unauthorized

Port  Port-Mode     Re-authentication   Shutdown-timeout   Control-Mode
                                                          admin   oper
----- ------------- -----------------   ----------------   ---------------
 3/1  SingleAuth    disabled            disabled           Both    Both
Console> (enable)
```

Note To clear the current state machines for a new authentication, enter the **set port dot1x** *mod/port* **initialize** command.

Enabling 802.1X with Inaccessible Authentication Bypass

You can enable 802.1X inaccessible authentication bypass on a per-port basis. This feature allows you to specify a port as critical. When a port is specified as a critical port, 802.1X attempts to authenticate the port in the normal way. If attempts to reach the authentication server fail, the port is still given access to the network in the administratively configured VLAN or the port's native VLAN. You can configure a port as critical only if it is in single-authentication mode.

After a critical port obtains access to the network, if the authentication server becomes available, the critical port returns to the unauthorized state, the normal authentication process restarts, and the critical port moves into the RADIUS server-specified VLAN after the port is authenticated. At this point, you must initialize the port manually using the **set port dot1x** *mod/port* **initialize** command.

If the authentication server goes down after a host has already been authenticated through the normal authentication process, the switch checks if the port is a critical port. If the switch determines that the port is a critical port, the normal reauthentication process is temporarily disabled for the port and the port is given network access until the authentication server becomes active and restarts the authentication process.

To specify a port as a critical port, perform this task in privileged mode:

	Task	Command
Step 1	Specify a port as a critical port.	**set port dot1x** *mod/port* **critical** {**enable** | **disable**}
Step 2	Verify the 802.1X configuration.	**show port dot1x** *mod/port*

This example shows how to specify a port as a critical port:

```
Console> (enable) set port dot1x 5/48 critical enable
Port 5/48 critical-port option is enabled
Console> (enable)
```

This example shows how to verify the 802.1X configuration:

```
Console> (enable) show port dot1x 5/48
Port  Auth-State          BEnd-State Port-Control        Port-Status
----- ------------------- ---------- ------------------- -------------
 5/48 -                   -          force-authorized    -

Port  Port-Mode     Re-authentication   Shutdown-timeout   Control-Mode
                                                          admin   oper
----- ------------- -----------------   ----------------   ---------------
 5/48 SingleAuth    disabled            disabled           Both    -

Port  Posture-Token Critical Termination action Session-timeout
----- ------------- -------- ------------------ ---------------
 5/48 -             YES      -                  -
Console> (enable)
```

Enabling Multiple 802.1X Authentications

You can specify multiple authentications so that more than one host can gain access to an 802.1X port. Cisco-proprietary multiple authentication allows multiple dot1x-hosts on a port; every host is authenticated separately. Use these guidelines when enabling multiple 802.1X authentications:

- The traffic from the non-802.1X hosts on multiple authenticated ports is blocked.
- You cannot enable a guest VLAN on multiple authenticated ports.
- You cannot enable multiple authentication on a MVAP.
- Multiple authenticated ports go into the port VLAN and will not go into a RADIUS-assigned VLAN.
- You need to enable port security on a port before you can enable multiple authentications on the port.
- You cannot disable port security on a multiple authenticated port.
- The port security timers are used on multiple authenticated ports. The reauthentication timers are not used on multiple authenticated ports.

To enable multiple 802.1X authentications, perform this task in privileged mode:

	Task	Command
Step 1	Enable multiple 802.1X authentications on a specific port.	**set port dot1x** *mod/port* **multiple-authentication** {**enable** | **disable**}
Step 2	Verify the 802.1X configuration.	**show port dot1x** *mod/port*

This example shows how to enable multiple 802.1X authentications on port 1 in module 3 and verify the configuration:

```
Console> (enable) set port dot1x 3/1 multiple-authentication enable
PortSecurity should be enabled on port 3/1, before enabling Multiple-authentication
Port Security not enabled on 3/1.
Console> (enable) set port security 3/1 enable
Port 3/1 security enabled.
Console> (enable) set port dot1x 3/1 multiple-authentication enable
Port 3/1 Multiple-authentication option enabled
Console> (enable) show port dot1x 3/1
Port  Auth-State          BEnd-State Port-Control        Port-Status
----- ------------------- ---------- ------------------- -------------
 3/1  connecting          idle       auto                unauthorized

Port  Port-Mode     Re-authentication   Shutdown-timeout   Control-Mode
                                                          admin   oper
----- ------------- -----------------   ----------------   ---------------
 3/1  MultiAuth     disabled            disabled           Both    Both
Console> (enable)
```

Setting and Enabling Automatic Reauthentication of the Host

You can specify how often 802.1X authentication reauthenticates the host if you do so before you enable automatic 802.1X host reauthentication. If you do not specify a time period before you enable host reauthentication, 802.1X defaults to 3600 seconds (the valid values are from 1–65535 seconds).

You can enable automatic 802.1X host reauthentication for the hosts that are connected to a specific port. To manually reauthenticate the host that is connected to a specific port, see the "Manually Reauthenticating the Host" section on page 40-18.

To set how often 802.1X authentication reauthenticates the host and enable automatic 802.1X reauthentication, perform this task in privileged mode:

	Task	Command
Step 1	Set the time constant for reauthenticating the host.	**set dot1x re-authperiod** *seconds*
Step 2	Enable reauthentication.	**set port dot1x** *mod/port* **re-authentication enable**
Step 3	Verify the 802.1X configuration.	**show port dot1x** *mod/port*

This example shows how to set automatic reauthentication to 7200 seconds, enable 802.1X reauthentication on port 3/1, and verify the configuration:

```
Console> (enable) set dot1x re-authperiod 7200
dot1x re-authperiod set to 7200 seconds
Console> (enable) set port dot1x 3/1 re-authentication enable
Port 3/1 Dot1x re-authentication enabled.
Console> (enable) show port dot1x 3/1
Port  Auth-State          BEnd-State Port-Control        Port-Status
----- ------------------- ---------- ------------------- -------------
 3/1  connecting          idle       auto                unauthorized

Port  Port-Mode     Re-authentication   Shutdown-timeout   Control-Mode
                                                           admin   oper
----- ------------ -----------------   ----------------   ---------------
 3/1  MultiAuth     enabled             disabled           Both    Both
Console> (enable)
```

Manually Reauthenticating the Host

You can manually reauthenticate the host that is connected to a specific port at any time. When you want to configure automatic 802.1X host reauthentication, see the "Setting and Enabling Automatic Reauthentication of the Host" section on page 40-17.

To manually reauthenticate a host that is connected to a specific port, perform this task in privileged mode:

Task	Command
Manually reauthenticate the host that is connected to a specific port.	**set port dot1x** *mod/port* **re-authenticate**

This example shows how to manually reauthenticate the host that is connected to port 1 on module 3:

```
Console> (enable) set port dot1x 3/1 re-authenticate
Port 3/1 re-authenticating...
dot1x re-authentication successful...
dot1x port 3/1 authorized.
Console> (enable)
```

Enabling Multiple Hosts

You can enable a specific port to allow multiple-user access. When a port is enabled for multiple users, and a host that is connected to that port is authorized successfully, any host (with any MAC address) is allowed to send and receive the traffic on that port. If you connect multiple hosts to that port through a hub, you can reduce the security level on that port.

To enable access for multiple hosts on a specific port, perform this task in privileged mode:

Task	Command
Enable multiple hosts on a specific port.	**set port dot1x** *mod/port* **multiple-host enable**

This example shows how to enable access for multiple hosts on port 1 on module 3:

```
Console> (enable) set port dot1x 3/1 multiple-host enable
Port 3/1 Multiple-host option enabled.
Console> (enable)
```

Disabling Multiple Hosts

You can disable multiple-user access on any port where it is enabled.

To disable access for multiple hosts on a specific port, perform this task in privileged mode:

Task	Command
Disable multiple hosts on a specific port.	**set port dot1x** *mod/port* **multiple-host disable**

This example shows how to disable access for multiple hosts on port 1 on module 3:

```
Console> (enable) set port dot1x 3/1 multiple-host disable
Port 3/1 Multiple-host option disabled.
Console> (enable)
```

Setting the Quiet Period

When the authenticator cannot authenticate the host, it remains idle for a set period of time and then tries again. The idle time is determined by the quiet-period value. (The default is 60 seconds.) You may set the value from 0–65535 seconds.

To set the value for the quiet period, perform this task in privileged mode:

Task	Command
Set the quiet-period value.	**set dot1x quiet-period** *seconds*

This example shows how to set the quiet period to 45 seconds:

```
Console> (enable) set dot1x quiet-period 45
dot1x quiet-period set to 45 seconds.
Console> (enable)
```

Setting the Shutdown Timeout Period

If a port is shut down because of a security violation, you must either manually reenable it or configure the shutdown timeout period after which the port can be enabled again.

To set the period of time that a port will be disabled after a security violation, perform this task in privileged mode:

Task	Command
Set the shutdown timeout period.	**set dot1x shutdown-timeout** *1- 65535 seconds*

This example shows how to set the shutdown timeout period:

```
Console> (enable) set dot1x shutdown-timeout 300
dot1x shutdown-timeout set to 300 seconds.
Console> (enable)
```

Setting the Authenticator-to-Host Retransmission Time for EAP-Request/Identity Frames

The host notifies the authenticator that it received the EAP-request/identity frame. When the authenticator does not receive this notification, the authenticator waits a set period of time and then retransmits the frame. You may set the amount of time that the authenticator waits for notification from 1–65535 seconds. (The default is 30 seconds.)

To set the authenticator-to-host retransmission time for the EAP-request/identity frames, perform this task in privileged mode:

Task	Command
Set the authenticator-to-host retransmission time for EAP-request/identity frames.	**set dot1x tx-period** *seconds*

This example shows how to set the authenticator-to-host retransmission time for the EAP-request/identity frame to 15 seconds:

```
Console> (enable) set dot1x tx-period 15
dot1x tx-period set to 15 seconds.
Console> (enable)
```

Setting the Back-End Authenticator-to-Host Retransmission Time for the EAP-Request Frames

The host notifies the back-end authenticator that it received the EAP-request frame. When the back-end authenticator does not receive this notification, the back-end authenticator waits a set period of time and then retransmits the frame. You may set the amount of time that the back-end authenticator waits for notification from 1–65535 seconds. (The default is 30 seconds.)

To set the back-end authenticator-to-host retransmission time for the EAP-request frames, perform this task in privileged mode:

Task	Command
Set the back-end authenticator-to-host retransmission time for the EAP-request frame.	**set dot1x supp-timeout** *seconds*

This example shows how to set the back-end authenticator-to-host retransmission time for the EAP-request frame to 15 seconds:

```
Console> (enable) set dot1x supp-timeout 15
dot1x supp-timeout set to 15 seconds.
Console> (enable)
```

Setting the Back-End Authenticator-to-Authentication-Server Retransmission Time for the Transport Layer Packets

The authentication server notifies the back-end authenticator each time that it receives a transport layer packet. When the back-end authenticator does *not* receive a notification after sending a packet, the back-end authenticator waits a set period of time and then retransmits the packet. You may set the amount of time that the back-end authenticator waits for notification from 1–65535 seconds. (The default is 30 seconds.)

To set the value for the retransmission of transport layer packets from the back-end authenticator to the authentication server, perform this task in privileged mode:

Task	Command
Set the back-end authenticator-to-authentication-server retransmission time for the transport layer packets.	**set dot1x server-timeout** *seconds*

This example shows how to set the value for the retransmission time for the transport layer packets that are sent from the back-end authenticator to the authentication server to 15 seconds:

```
Console> (enable) set dot1x server-timeout 15
dot1x server-timeout set to 15 seconds.
Console> (enable)
```

Setting the Back-End Authenticator-to-Host Frame-Retransmission Number

The authentication server notifies the back-end authenticator each time that it receives a specific number of frames. When the back-end authenticator does not receive this notification after sending the frames, the back-end authenticator waits a set period of time and then retransmits the frames. You may set the number of frames that the back-end authenticator retransmits from 1–10 (the default is 2).

To set the number of frames that are retransmitted from the back-end authenticator to the host, perform this task in privileged mode:

Task	Command
Set the back-end authenticator-to-host frame retransmission number.	**set dot1x max-req** *count*

This example shows how to set the number of retransmitted frames that are sent from the back-end authenticator to the host to 4:

```
Console> (enable) set dot1x max-req 4
dot1x max-req set to 4.
Console> (enable)
```

Setting the Critical Recovery Delay for an Authentication Feature

You can set the critical recovery delay for each authentication feature. By default, critical recovery delay is disabled. The critical recovery delay can be set between 1–10000 milliseconds. Ports enabled with the critical recovery delay feature will be moved to the “critical” state when the RADIUS server is not

available to authenticate. The ports moved to critical state are initalized when the RADIUS server comes online and the RADIUS auto-initialization feature is enabled. During the initialization process, the ports that were moved to the critical state are initialized after the configured critical recovery delay interval.

For example, if there are 10 ports enabled with dot1x and moved to the critical state, the ports are initialized when the RADIUS server comes online. If you configure a delay of 10 milliseconds, the initialization for each port is delayed by 10 milliseconds before the initialization process begins. After each 10-millisecond period is completed, the next port initializes until all the ports have gone through the initialization process.

Task	Command
Set the critical recovery delay feature.	**set [dot1x | mac-auth-bypass | eou | web-auth] critical-recovery-delay** *time*

This example shows how to set the critical recovery delay to 10 milliseconds for dot1x:

```
Console> (enable) set dot1x critical-recovery-delay 10
Dot1x critical recovery delay set to 10 milliseconds.
Console> (enable)
```

Resetting the 802.1X Configuration Parameters to the Default Values

You can reset the 802.1X configuration parameters to the default values with a single command, which also globally disables 802.1X.

To reset the 802.1X configuration parameters to the default values, perform this task in privileged mode:

	Task	Command
Step 1	Reset the 802.1X configuration parameters to the default values and globally disable 802.1X.	**clear dot1x config**
Step 2	Verify the 802.1X configuration.	**show dot1x**

This example shows how to reset the 802.1X configuration parameters to the default values and verify the configuration:

```
Console> (enable) clear dot1x config
This command will disable dot1x on all ports and take dot1x parameter values back to
factory defaults.
Do you want to continue (y/n) [n]?
Console> (enable) show dot1x
PAE Capability             Authenticator Only
Protocol Version           1
system-auth-control        enabled
max-req                    2
quiet-period               45 seconds
radius-accounting          disabled
radius-vlan-assignment     enabled
radius-keepalive state     enabled
re-authperiod              7200 seconds
server-timeout             30 seconds
shutdown-timeout           300 seconds
supp-timeout               30 seconds
tx-period                  30 seconds
```

```
Console> (enable)
```

Enabling 802.1X Authentication for the DHCP Relay Agent

To enable the DCHP Relay Agent to send 802.1X parameters for a particular VLAN to the DHCP server, perform this task in privileged mode:

Note The management VLAN (the VLAN that is configured on the sc0 or sc1 interfaces) cannot be mapped to an ACL that has a dot1x-dhcp ACE. You cannot use the **clear config interface** command when VLAN 1 or VLAN 2 is mapped to an ACL that has a dot1x-dhcp ACE.

	Task	Command
Step 1	Enable 802.1X authentication for the DHCP Relay Agent. **Note** This command creates an ACE entry with the given ACL name. The ACL can have other ACE entries but DHCP ACE entries are given priority.	**set security acl ip** *acl_name* **permit dot1x-dhcp**
Step 2	Verify the 802.1X configuration.	**show dot1x**

This example shows how to create an ACL entry for the 802.1X DHCP relay traffic:

```
Console> (enable) set security acl ip dhcp_relay permit dot1x_dhcp
Successfully configured Dot1x Dhcp ACL for dhcp_relay. Use ‘commit’ command to save
changes
```

This example shows how to configure the ACL to allow other traffic than DHCP on an existing ACL entry:

```
Console> (enable) set security acl ip dhcp_relay permit any
dhcp_relay editbuffer modified. Use ’commit’ command to apply changes.
console> (enable)
```

This example shows how to commit the ACE to NVRAM:

```
Console> (enable) commit security acl dhcp_relay
Commit operation in progress
ACL ‘dhcp_relay’ successfully committed.
```

This example shows how to map the VLANs that should be applied to dhcp-relay-acl:

```
Console> (enable) set security acl map dhcp_relay 1-3,20
Mapping in progress...
ACL dhcp_relay successfully mapped to VLAN 1.
ACL dhcp_relay successfully mapped to VLAN 2.
ACL dhcp_relay successfully mapped to VLAN 3.
ACL dhcp_relay successfully mapped to VLAN 20.
```

The DHCP Relay Agent Information field is added in the DHCP packet that is forwarded from the client to the server. The VLANs that are not mapped to “dhcp-relay-acl” and all DHCP packets are switched as usual without any modifications.

Disabling 802.1X Authentication for the DHCP Relay Agent

To disable the DCHP Relay Agent from sending the 802.1X parameters for a particular VLAN to the DHCP server, perform this task in privileged mode:

	Task	Command
Step 1	Disable 802.1X authentication for the DHCP Relay Agent.	**clear security acl map dhcp_relay** *vlan_ID*
Step 2	Verify the 802.1X configuration.	**show dot1x**

This example shows how to configure the DHCP Relay Agent to stop sending the 802.1X authentication parameters for VLANs 1–3 and 20 and verify the configuration:

```
Console> (enable) clear security acl map dhcp_relay 1-3,20
Successfully cleared mapping between ACL dhcp_relay and VLAN 1.
Successfully cleared mapping between ACL dhcp_relay and VLAN 2.
Successfully cleared mapping between ACL dhcp_relay and VLAN 3.
Successfully cleared mapping between ACL dhcp_relay and VLAN 20.
```

Adding Hosts to an 802.1X Guest VLAN

Typically, the guest VLANs support minimal services and provide minimal network access. The hosts can be added to the guest VLAN only when the **set port dot1x** *mod/port* **port-control auto** command option is used. If you change the **set port dot1x** *mod/port* **port-control** command option from **auto** to **force-authorized** or **force-unauthorized**, the host is removed from the guest VLAN and added back to the port VLAN.

To add a port to an 802.1X guest VLAN, perform this task in privileged mode:

	Task	Command
Step 1	Configure an active VLAN as an 802.1X guest VLAN.	**set port dot1x** *mod/port* **guest-vlan** {*vlan* | **none**}
Step 2	Verify the per-port 802.1X guest VLAN configuration.	**show port dot1x guest-vlan**

This example shows how to add port 3/1 to 802.1X guest VLAN 200:

```
Console> (enable) set port dot1x 3/1 guest-vlan 200
Port 3/1 is Multiple-authentication enabled, guest-vlan can not be enabled
Console> (enable) set port dot1x 3/1 multiple-authentication disable
Port 3/1 Multiple-authentication option disabled
Console> (enable) set port dot1x 3/1 guest-vlan 200
Port 3/1 Guest Vlan is set to 200
Console> (enable) show port dot1x guest-vlan
Guest-Vlan    Status   Mod/Ports
------------- -------- ------------------
200           active   3/1
none          none     2/1-2,3/2-48,8/1-8
Console> (enable)
```

This example shows how to remove the port from the guest VLAN:

```
Console> (enable) set port dot1x 3/1 guest-vlan none
Port 3/1 Guest Vlan is cleared
Console> (enable)
```

Configuring an 802.1X Unidirectional Controlled Port

802.1X allows you to use wake-on LAN technology (also referred to as remote wake-up) to perform the unattended system backups or software upgrades on the hosts that are attached to the switch.

When you configure a unidirectional controlled port, the port allows outbound-only traffic prior to host authentication. This behavior enables a management station to send the wake-on LAN frames to selected hosts that trigger the host to power up and boot, authenticate, and then perform the unattended operation.

Note The wake-on LAN technology requires specific hardware for the host that is outside the scope of this publication.

Prior to software release 8.3(1), the 802.1X bridge ports were configured by default to a bidirectional state where the control was exerted on protocol exchanges in both directions on the unauthorized ports. With the unidirectional controlled port feature, you can configure the 802.1X-capable ports to be in unidirectional (**in** keyword) or bidirectional (**both** keyword) states using the **set port dot1x** *mod/port* **port-control-direction** command.

Unidirectional State

When you configure a port as a unidirectional port (**in** keyword) and set the port to **auto** using the **set port dot1x** *mod/port* **port-control auto** command, the bridge port is moved into the spanning-tree forwarding state where all the traffic to the port is redirected to the supervisor engine for processing. With the wake-on LAN functionality, when the connected host is in sleeping mode or a power-down state, the host does not exchange the traffic with any other devices in the network. The hosts that are connected to the unidirectional port cannot send the traffic out into the network; they can only receive the traffic from the other devices in the network. If the unidirectional port sees any kind of incoming traffic, the port returns to the bidirectional (default) state and the spanning-tree state is moved to the blocking state where both the incoming and outgoing traffic are dropped. The authenticator system on the port moves the port into the initialize state and no traffic is allowed other than the EAPOL packet exchanges. When the port is returned to the bidirectional state, a 5-minute timer is started and if the port is not authenticated before the timer runs out, the port switches back to a unidirectional port.

Bidirectional State

When you configure a port as a bidirectional port (**both** keyword) and set the port to **auto** using the **set port dot1x** *mod/port* **port-control auto** command, the port is access controlled in both directions. This state disables the reception of any incoming packets and the transmission of outgoing packets on the port. When the port is configured as a bidirectional port, it behaves as it did in software releases prior to release 8.3(1); the port is in the spanning-tree blocking state and the normal authentication process is followed.

Configuration Guidelines

This section provides the guidelines for configuring 802.1X unidirectional ports:

- Auxiliary VLANs—To support auxiliary VLANs on a port when you configure the port as a unidirectional port, the auxiliary VLAN is moved to the spanning-tree forwarding state to ensure that the connected IP phone is operational immediately. To prevent any disturbance of the incoming traffic, initially the port VLAN is also moved to the spanning-tree forwarding state and then if any traffic is seen on the port VLAN, the port is moved to the spanning-tree blocking state to drop all additional traffic. The connected host is then requested to get authorized to send any traffic.
- Guest VLANs—The guest VLANs are supported only on the ports that are configured as bidirectional ports. If a guest VLAN is enabled on a port, that port cannot be configured as a unidirectional port, and conversely, a unidirectional port cannot be configured in a guest VLAN.
- Port mode—The port mode (single-authentication mode, multiple-host mode, or multiple-authentication mode) for a port configured as a unidirectional port must be single-authentication mode (the default port mode).

Using the CLI to Configure an 802.1X Unidirectional or Bidirectional Port

If you specify the **in** keyword, all the incoming traffic is dropped and the outgoing traffic is allowed. If you specify the **both** keyword (the default), all the receiving traffic and transmitting traffic on the port is dropped. To configure a port as an 802.1X unidirectional port or bidirectional port, perform this task in privileged mode:

Task	Command
Configure a port as an 802.1X unidirectional port or bidirectional port.	**set port dot1x** *mod/port* **port-control-direction** [**both** | **in**]

These examples show how to set a port to unidirectional or bidirectional states and verify the configuration:

```
Console> (enable) set port dot1x 3/1 port-control-direction both
Port 3/1 Port Control Direction set to Both.
Console> (enable) set port dot1x 3/1 port-control-direction in
Port 3/1 Port Control Direction set to In.
Console> (enable) show port dot1x 3/1
Port  Auth-State          BEnd-State Port-Control        Port-Status
----- ------------------- ---------- ------------------- -------------
 3/1  connecting          idle       auto                unauthorized

Port  Port-Mode     Re-authentication   Shutdown-timeout   Control-Mode
                                                           admin   oper
----- ------------- -----------------   ----------------   ---------------
 3/1  SingleAuth    enabled             disabled           In      Both
Console> (enable)
```

Configuring 802.1X with ACL Assignments

These sections describe how to configure 802.1X with ACL assignments:

- Overview, page 40-27
- 802.1X with ACL Assignments Configuration Guidelines, page 40-28

Overview

When you configure 802.1X with ACL assignments, the identity-based ACLs are used to dynamically assign an access control policy to an interface that is based on the user's 802.1X authentication. This feature restricts the users to certain network segments, limits the access to the sensitive servers, and restricts the protocols and applications that may be used. This feature also allows you to provide very specific identity-based security without compromising user mobility or significantly increasing the administrative overhead.

When you configure 802.1X with ACL assignments, you eliminate the problem of creating, modifying, and removing the access control policies that are based on the IP/MAC addresses whenever the user's physical location changes in the network. This feature allows you to create the identity-based security access policies rather than the VLAN-based policies (VACLs) or the port-based policies (PACLs) without compromising user mobility. With this feature, the user does not have to rely on the network administrator to enforce the access policy changes whenever the user's physical location and/or connection to the network changes.

The new **group** *group_name* keyword is used to classify the policy as a group. A group is a set of users (their IP addresses) to which the policy applies. Prior to this feature, if you wanted to permit the IP access to a set of users, you had to specify each user's IP address in the ACL ACE and there could only be one IP address per ACE. With this new feature, you specify a *group_name* in the ACE, such as **set security acl ip** *grpacl* **permit ip group** *ip-permit-group* **any**, where the *ip-permit-group* is a group and all the users that are part of that group are authenticated. After a successful user authentication and after the user's IP address is obtained, if the user is part of the group, the user's IP address is added to the group and a new ACE is created and installed in the hardware (PFC). The ACL grows and shrinks dynamically upon user authentication and logoff; the ACL is dynamic and the policy is installed only for the authenticated and valid users.

When you configure 802.1X with ACL assignments, you can automatically configure the QoS ACLs and VACLs to a user once the user is authenticated. The RADIUS server sends a QoS VLAN-based ACL, QoS port-based ACL, or VACL policy name with the authentication success packet. The policy that is associated with the policy name is already configured on the switch through the CLI. The policy is converted into a set of ACEs and then installed on the switch.

You can apply the ACLs to an IP address. Because the 802.1X authentication is done on a username and can be tied to a MAC address—but the IP address is not known at the time of authentication (DHCP is started by the host only after a successful authentication)—the ACE installation occurs only after the IP address is known either through DHCP snooping or dynamic ARP inspection.

When you configure 802.1X with ACL assignments, you perform these two main configuration tasks:

- Associate and configure the group names for the users in the RADIUS server
- Configure, commit, and map the ACLs on the switch for the groups using the switch CLI

After you configure the 802.1X ACL assignments, the switch does the following:

- Authenticates the user(s)
- Uses DHCP snooping or dynamic ARP inspection to obtain the IP address of the user(s)
- Expands the ACL using the IP address(es) and programs the PFC

802.1X with ACL Assignments Configuration Guidelines

This section provides the guidelines for configuring 802.1X with ACL assignments:

- The port mode (single-authentication mode, multiple-host mode, or multiple-authentication mode) for a port that is configured for 802.1X with ACL assignments must be single-authentication mode (the default port mode).
- Dynamically learned IP addresses (obtained through DHCP snooping or dynamic ARP inspection) are used to expand the group name. 802.1X with ACL assignments is also supported with static IP addresses (the static IP address should also be configured in the RADIUS server).
- The groups are policy groups. An example of a policy group would be a policy such as "deny http access" that applies to a set of IP addresses.
- The user is never permanently tied to a group, and a user can be part of multiple policy groups simultaneously. If you want to define more than one policy, for example, if you want both "deny http access" *and* "deny ftp access," you can define two policy groups—one policy group as "http deny" and another policy group as "ftp deny."
- The RADIUS server can send all the policies that have to be applied to a particular user in the authentication success packet, and the user can be added to all those groups on the switch. If a policy group sent by the RADIUS server is not configured on the switch, the policy is either ignored or the port goes into the unauthorized state. If the RADIUS server sends a group ID that is not present in any ACL on the switch, authentication fails.
- With software release 8.3(1) and later releases, you can load balance the 802.1X-authenticated users that are configured under one group name by distributing them evenly between the VLANs. For more configuration information, see the "Configuring 802.1X User Distribution" section on page 40-32.

Using the CLI to Configure 802.1X with ACL Assignments

Note This section describes the CLI introduced in software release 8.3(1), which is used to configure 802.1X with ACL assignments. For more information on configuring the ACLs, see Chapter 15, "Configuring Access Control."

To configure 802.1X with ACL assignments, perform this task in privileged mode:

Task	Command
Configure 802.1X with ACL assignments.	**set security acl ip** {*acl_name*} {**permit** | **deny** | **redirect** {*mod_num/port_num*}} [**ip**] {*src_ip_spec* | [**group** {*group_name*}} {*dest_ip_spec* | [**group**} [**precedence** {*precedence*}] [**tos** {*tos*}] [**fragment**] [**capture**] [**log**] [**before** {*editbuffer_index*} | **modify** {*editbuffer_index*}]

This example shows how to specify a group name for an 802.1X group and verify that the group was configured:

```
Console> (enable) set security acl ip grpacl permit ip group ip-permit-group any
grpacl editbuffer modified. Use 'commit' command to apply changes.
Console> (enable) commit security acl grpacl
```

```
ACL commit in progress.

ACL 'grpacl' successfully committed.
Console> (enable)

Console> (enable) show dot1x group all
Group Manager Info

Current Group Count = 1

------------------------------------------------------------
Info of Group ip-permit-group
User Count  = 0
Console> (enable)
```

Configuring 802.1X with QoS ACLs

These sections describe how to configure 802.1X with QoS ACLs:

- 802.1X with QoS ACLs Configuration Guidelines, page 40-29
- 802.1X with QoS ACLs Configuration Example, page 40-30
- Configuring the RADIUS Server, page 40-31

The RADIUS server sends a policy name to the 802.1X client. The policy is already defined and committed on the switch. The user is able to fully utilize all existing QoS features when defining the QoS policy. The 802.1X client interacts with the QoS subsystem and applies the policy on an interface after authentication has been made. The policy is removed when the authenticated client leaves the interface. If 802.1X has attached a policy to an interface, it is still possible for you to unmap the policy directly through the switch CLI.

802.1X with QoS ACLs Configuration Guidelines

This section describes the guidelines for configuring 802.1X with QoS ACLs:

- If a QoS policy misconfiguration exists and 802.1X attempts to authenticate a user on an interface, the authentication will fail.
- If you misconfigure a QoS policy after 802.1X has properly authenticated the interface, authentication will fail when reauthentication is attempted on the interface with that same QoS policy.
- If multiple QoS policies are applied at the same time (input and output policies), authentication will fail if any of the QoS policies fail.
- If you apply a port-based policy and a VLAN-based policy to the same interface, the authentication will fail.
- The 802.1X security and QoS policies are applied only when an 802.1X user logs in. If you change the 802.1X security and/or QoS policy on the switch or the RADIUS server, the changes are not applied until the 802.1X user reauthenticates. If reauthentication is enabled (nondefault), the policy will take effect usually within one hour. If reauthentication is disabled (default), the policy changes will not take effect until each 802.1X user logs out and logs back in.
- The existing QoS commands are used to create and show the QoS policy information. The commands include but are not limited to the **set qos enable**, **set qos acl**, and **commit qos acl** commands. Scheduling commands and port-based QoS commands may also be used to build the dynamic QoS policy.

- After you define a QoS policy on the switch, you should map the policy to a VLAN or port (using the **set qos acl map** command) and verify that the policy mapping succeeds. After verification, clear the ACL mapping and configure 802.1X on the interface.

Note Be careful when you name the QoS ACL. The QoS ACL name must match the policy name specified on the RADIUS server.

802.1X with QoS ACLs Configuration Example

In the following example, QoS is enabled and an 802.1X QoS policy (Dot1xDscp5Policy) is created. The policy is then committed. The same policy name (Dot1xDscp5Policy) is then configured on the RADIUS server. After a period of time, you can see that the policy is applied to port 3/1 after 802.1X has authenticated a client and applied the policy. You can see that the policy mapping is not found in the configuration (config) display of the mapping command: it is found only in the run-time configuration.

The AV-pairs at the RADIUS server require the following input—qos:inpacl=Dot1xDscp5Policy. After supplicant authentication on port 3/1, the QoS run-time mapping to port 3/1 occurs.

The other options for the AV-pairs are as follows—qos:invacl=<policy-name> and qos:outpacl=<policy-name>.

If the policy name in the AV-pairs does not match a policy name in the switch, the supplicant is not authenticated.

```
Console> (enable) set qos enable
QoS is enabled.
Console> (enable) set qos acl ip Dot1xDscp5Policy dscp 5 any
Dot1xDscp5Policy editbuffer modified. Use 'commit' command to apply changes.
Console> (enable) commit qos acl all

QoS ACL 'Dot1xDscp5Policy' successfully committed.
Console> (enable) show qos acl map config Dot1xDscp5Policy
QoS ACL mappings on input side:
ACL name                         Type Vlans
-------------------------------- ---- ---------------------------------
Dot1xDscp5Policy                   IP
ACL name                         Type Ports
-------------------------------- ---- ---------------------------------
Dot1xDscp5Policy                   IP
QoS ACL mappings on output side:
ACL name                         Type Vlans
-------------------------------- ---- ---------------------------------
Dot1xDscp5Policy                   IP
Console> (enable)

<<< Dot1x Authenticates a client on 3/1 and applies Dot1xDscp5Policy >>>

Console> (enable) show qos acl map runtime Dot1xDscp5Policy
QoS ACL mappings on input side:
ACL name                         Type Vlans
-------------------------------- ---- ---------------------------------
Dot1xDscp5Policy                   IP
ACL name                         Type Ports
-------------------------------- ---- ---------------------------------
Dot1xDscp5Policy                   IP 3/1
QoS ACL mappings on output side:
ACL name                         Type Vlans
-------------------------------- ---- ---------------------------------
Dot1xDscp5Policy                   IP
Console> (enable) show qos acl map config Dot1xDscp5Policy
```

```
QoS ACL mappings on input side:
ACL name                         Type Vlans
-------------------------------- ---- --------------------------------
Dot1xDscp5Policy                   IP
ACL name                         Type Ports
-------------------------------- ---- --------------------------------
Dot1xDscp5Policy                   IP
QoS ACL mappings on output side:
ACL name                         Type Vlans
-------------------------------- ---- --------------------------------
Dot1xDscp5Policy                   IP
Console> (enable)
```

This example shows that the dynamic QoS policy information is displayed using the **show qos acl map** command. When you use the **runtime** keyword, you can see which dynamic policies have been applied to which interfaces. The **config** keyword does not show the dynamic QoS policy mapping.

```
Console> (enable) show qos acl map config Dot1xDscp5Policy
QoS ACL mappings on input side:
ACL name                         Type Vlans
-------------------------------- ---- --------------------------------
Dot1xDscp5Policy                   IP
ACL name                         Type Ports
-------------------------------- ---- --------------------------------
Dot1xDscp5Policy                   IP
QoS ACL mappings on output side:
ACL name                         Type Vlans
-------------------------------- ---- --------------------------------
Dot1xDscp5Policy                   IP
Console> (enable) show qos acl map runtime Dot1xDscp5Policy
QoS ACL mappings on input side:
ACL name                         Type Vlans
-------------------------------- ---- --------------------------------
Dot1xDscp5Policy                   IP
ACL name                         Type Ports
-------------------------------- ---- --------------------------------
Dot1xDscp5Policy                   IP 3/1
QoS ACL mappings on output side:
ACL name                         Type Vlans
-------------------------------- ---- --------------------------------
Dot1xDscp5Policy                   IP
Console> (enable)
```

Configuring the RADIUS Server

Using Cisco Secure Access Control Server (ACS) 3.x or higher, you need to configure the QoS policy name associated with an authenticated client. To configure the RADIUS server, perform these steps from the ACS home page:

Step 1 Select **network configuration**.

Step 2 Click on the NAS IP on which to turn on the RADUIS IOS/PIX style of attributes.You will get the Authenticate Using field.

Step 3 Select the **IOS/PIX** option and submit.

Step 4 Select **interface config**.

Step 5 Select **RADIUS (IOS/PIX)**.

Step 6 Check both boxes before the AV-pair option. The first option itself is AV-pair.

The AV-pair box appears for every user. Check the box and then type the AV-pair strings in the window. The strings in this case represent the QoS policy name that you wish to associate with each user. If you are sending multiple AV-pair strings, you need to separate them with a new line so that each AV-pair is sent as a different 26/9/1 attribute.

Configuring 802.1X User Distribution

When you configure 802.1X user distribution, you can distribute the users that have the same group name across multiple VLANs. Before software release 8.3(1), the RADIUS VLAN assignment feature that was supported by 802.1X took the VLAN number that was obtained from the RADIUS server and added all the users to that VLAN. With software release 8.3(1) and later releases, you can load balance the 802.1X-authenticated users that are configured under one group name by distributing them evenly between the VLANs.

Use these two methods to load balance the users between the different VLANs. The VLANs are either supplied by the RADIUS server or configured under a VLAN group name through the switch CLI:

- Configure the RADIUS server to send more than one VLAN name for a user. The multiple VLAN names can be sent as part of the response to the user. The 802.1X user distribution tracks all the users in a particular VLAN and achieves load balancing by moving the authorized user to the least populated VLAN.
- Configure the RADIUS server to send a VLAN group name for a user. The VLAN group name can be sent as part of the response to the user. The selected VLAN group name is searched among the VLAN group names that you configured using the Catalyst CLI (see the "Using the CLI to Configure 802.1X User Distribution" section on page 40-33). If the VLAN group name is found, the corresponding VLANs that are configured under this VLAN group name are searched to find the least populated VLAN and load balancing is achieved by moving the corresponding authorized user to that VLAN.

802.1X User Distribution Configuration Guidelines

This section provides the guidelines for configuring the 802.1X user distribution feature:

- Ensure that at least one VLAN is mapped to the VLAN group.
- You can map more than one VLAN to a VLAN group.
- You can modify the VLAN group by adding or deleting a VLAN.
- When you clear an existing VLAN from the VLAN group name, none of the ports that are authenticated in the VLAN are cleared but the mappings are removed from the existing VLAN group.
- If you clear the last VLAN from the VLAN group name, the VLAN group is deleted.
- You can clear a VLAN group even when the active VLANs are mapped to the group. When you clear a VLAN group, none of the ports or users that are in the authenticated state in any VLAN within the group are cleared, but the VLAN mappings to the VLAN group are cleared.
- If you enter the **set dot1x radius-vlan-assignment disable** command, the VLAN information that is sent from the RADIUS server is ignored and the port stays in the NVRAM-configured VLAN. This command is used to enable or disable the VLAN assignment feature globally. When the command is enabled, the switch uses the tunnel attributes to extract the VLAN name in the RADIUS Access-Accept message. The command is enabled by default.

Using the CLI to Configure 802.1X User Distribution

To configure a VLAN group and map a VLAN to it, perform these tasks in privileged mode:

Task	Command
Configure a VLAN group and map a single VLAN or a range of VLANs to it.	**set dot1x vlan-group** {*vlan-group-name*} {*vlans*}
Verify the configuration.	**show dot1x vlan-group** [**all** | {*vlan-group-name*}]
Clear the VLAN group configuration or elements of the VLAN group configuration.	**clear dot1x vlan-group** [**all** {*vlan-group-name*} [{*vlans*}]]

This example shows how to configure the VLAN groups, map the VLANs to the groups, and verify that the VLAN groups were configured and mapped to the specified VLANs:

```
Console> (enable) set dot1x vlan-group eng-dept 10
Vlan group name eng-dept is successfully configured and mapped to vlan 10
Console> (enable) set dot1x vlan-group hr-dept 20
Vlan group name hr-dept is successfully configured and mapped to vlan 20
Console> (enable) show dot1x vlan-group eng-dept
Group Name                     Vlans Mapped
-------------                  --------------
eng-dept                       10
Console> (enable) show dot1x vlan-group all
Group Name                     Vlans Mapped
-------------                  --------------
eng-dept                       10
hr-dept                        20
Console> (enable)
```

This example shows how to add a VLAN to an existing VLAN group and verify that the VLAN was added:

```
Console> (enable) set dot1x vlan-group eng-dept 30
Vlan 30 is successfully mapped to vlan group eng-dept.
Console> (enable) show dot1x vlan-group eng-dept
Group Name                     Vlans Mapped
-------------                  --------------
eng-dept                       10,30
Console> (enable)
```

This example shows how to clear a VLAN from a VLAN group:

```
Console> (enable) clear dot1x vlan-group eng-dept 10
Vlan 10 is successfully cleared from vlan group eng-dept.
Console> (enable)
```

This example shows that when all the VLANs are cleared from a VLAN group, the VLAN group is cleared:

```
Console> (enable) clear dot1x vlan-group eng-dept 30
Vlan 30 is successfully cleared from vlan group eng-dept.
Warning: No more vlans mapped to this group, vlan group is cleared.
Console> (enable) show dot1x vlan-group eng-dept
Vlan group eng-dept doesn't exist, can not display.
Console> (enable)
```

This example shows how to clear all the existing VLAN groups:

```
Console> (enable) clear dot1x vlan-group all
Console> (enable) show dot1x vlan-group all
No vlan groups are present for display.
Console> (enable)
```

Enabling and Disabling 802.1X RADIUS Accounting and Tracking

You can use 802.1X RADIUS accounting and tracking to send the 802.1X user accounting information to the RADIUS server. The feature uses UDP port number 1813.

An 802.1X accounting packet can indicate the following information to the RADIUS server:

- When a user successfully authenticates
- When a user logs off
- When the link goes down on an 802.1X port
- When a reauthentication succeeds
- When a reauthentication fails

The attributes of the accounting packets are as follows (some attributes are optional):

- Attribute [1] USERNAME—The username that is going to be authenticated.
- Attribute [4] NAS-IP—The IP address of the switch that initiated the authentication/accounting session (typically, this is the sc0 interface IP address).
- Attribute [40] ACCT-STATUS-TYPE—START/STOP/INTERIM
  - START is sent when the authentication succeeds and the port is moved to the authorized state.
  - STOP is sent when the user sends a logoff, when the link goes down, or when reauthentication fails.
  - INTERIM is sent when a reauthentication succeeds.
- Attribute [44] ACCT-SESSION-ID—The unique session identifier that is associated with every accounting session.

The accounting packet format is as follows:

```
<NAS-IP> <user-id> <date> <time> <random16bit#>
```

An example of the accounting packet format is as follows:

```
9.9.150.140 rameshp 31/07/2003 12:40:00 12345
```

The attributes listed above are common regardless of the ACCT-STATUS-TYPE attribute (for START/STOP/INTERIM).

These attributes are specific to the INTERIM updates:

- Attribute [8] FRAMED-IP-ADDRESS—The IP address that is assigned to the user (this address can be obtained through a static assignment or through DHCP).

- Attribute [81] TUNNEL-PRIVATE-GROUP-ID—Actual VLAN name that is sent by the RADIUS server.

  CISCO-AV-PAIRS sent along with the above attribute in "Interim Accounting Request" are as follows:

  - AAA: ip-addr-method—Sent whether the IP assignment is through DHCP or statically configured.
  - AAA: vlan-assign-method—Device local or RADIUS assigned.

  The type is "device local" when the RADIUS server does not send a VLAN. In that case, the administratively-configured port VLAN is the VLAN for the user. If the RADIUS server sent the VLAN, the type is "RADIUS assigned."

These attributes are specific to the STOP packets:

- Attribute [49] ACCT-TERMINATION-CAUSE—The cause can be due to a user logoff, a port going down, reauthentication failures, and so on.
- CISCO-AV-PAIRS
  - Cisco:Input-Octets—A 64-byte integer that provides the number of bytes of ingress traffic that is received on the port.
  - Cisco:Output-Octets—A 64-byte integer that provides the number of bytes of egress traffic that is forwarded from the port.

Using the CLI to Enable and Disable 802.1X RADIUS Accounting and Tracking

To enable or disable 802.1X RADIUS accounting and tracking globally, perform this task in privileged mode (the default is disabled):

Task	Command
Enable or disable 802.1X RADIUS accounting and tracking globally.	**set dot1x radius-accounting** {**enable** | **disable**}

This example shows how to enable or disable 802.1X RADIUS accounting and tracking globally:

```
Console> (enable) set dot1x radius-accounting enable
dot1x radius-accounting enabled.
Console> (enable) set dot1x radius-accounting disable
dot1x radius-accounting disabled.
Console> (enable)
```

Enabling and Disabling RADIUS Keepalive

Use the **set radius-keepalive enable** command to check if configured RADIUS servers are alive. When the command is enabled, the switch sends out a test username for authentication. In reply to the test username, the RADIUS servers send an access rejection. To turn off authentication attempts that test the RADIUS servers, enter the **set radius-keepalive disable** command. If you disable this feature, the switch does not check the status of the servers, and the RADIUS server logs do not record the test attempts.

Note In software releases 7.5 through 8.2, the command that you used to enable or disable the RADIUS keepalive feature was the **set feature dot1x-radius-keepalive** command. In software release 8.3 through 8.5, the command that you used to enable or disable the RADIUS keepalive feature was the **set dot1x radius-keepalive** command. In software release 8.6 and later releases, the command to enable or disable the RADIUS keepalive feature is the **set radius keepalive** command.

To enable or disable the RADIUS keepalive feature, perform this task in privileged mode (the default is enabled):

Task	Command
Enable or disable the RADIUS keepalive feature.	**set radius keepalive** {**enable** | **disable**}

This example shows how to globally enable the RADIUS keepalive feature:

```
Console> (enable) set radius keepalive enable
Radius Keepalive enabled.
Console> (enable)
```

To set the RADIUS keepalive time in seconds, perform this task in privileged mode (the default is 60 seconds):

Task	Command
Set the RADIUS keepalive time. The time can be set from 1–6,000 seconds. The default is 60 seconds.	**set radius keepalive time** *seconds*

This example shows how to set the RADIUS keepalive time:

```
Console> (enable) set radius keepalive time 120
Radius keepalive time set to 120 seconds.
Console> (enable)
```

Configuring the Authenticated Identity-to-Port Description Mappings

You can use authenticated identity-to-port description mapping to assign a port name to the 802.1X port based on the information that is received from the RADIUS server. This feature uses an AV-pair "Supplicant Name" to uniquely assign a port name for an authenticated user. Currently, there is support only for the Cisco-supported AV-pairs that are sent from the authentication server; the other vendor-specific AV-pairs are ignored.

Enter the **show port dot1x name-mapping** command to display the name of the port that is received from the RADIUS server. If the switch receives an authenticated port name that is greater than or equal to 20 characters, the name is truncated to 19 characters and a # sign is appended to the name (allowing a total of 20 characters that is compatible with the **set port name** command). When you enter the **set port name** command, the end result is the same as if you had used the authenticated identity-to-port description mapping; the difference is that this feature assigns the name dynamically upon 802.1X authentication. An example of a dynamically assigned port name is as follows:

```
Console> (enable) show port dot1x name-mapping 5/1
 Port Port Name                 802.1X Port Name
 ---- ------------------        ------------------
 5/1  Cube-C1/2                 User1
```

Configuring the DNS Resolution for a RADIUS Server Configuration

When you configure the DNS resolution for a RADIUS server, you can configure the RADIUS server using a DNS name in addition to the IP addresses. The switch automatically resolves the DNS name using a DNS server that is configured to associate a DNS name with an IP address. The configured DNS name can coexist with the other IP addresses that are configured as primary or secondary. The DNS name is stored in NVRAM. You must enable the RADIUS keepalive feature for the DNS resolution to work. DNS resolution allows you to modify the IP address of the RADIUS server transparently without the knowledge of the switch. The switch can then resolve the DNS name with the modified IP address.

The switch resolves the DNS name a second time (reresolution) to the IP address during the initial configuration of the DNS name, when 802.1X is disabled and enabled, during the 802.1X port authentication, or if the request to the RADIUS server times out. The reresolution checks if the DNS name-to-IP address mapping is changed on the DNS server side.

Enter the **show config** or **show radius** commands to display the DNS name if the DNS name is configured in place of an IP address for the RADIUS server. You can configure a maximum of three RADIUS servers. To display the configured RADIUS server parameters, enter the **show radius** command as follows:

```
Console> (enable) show radius
RADIUS Deadtime:              0 minutes
RADIUS Key:                   cisco
RADIUS Retransmit:            2
RADIUS Timeout:               5 seconds
Framed-Ip Address Transmit:   Disabled

RADIUS-Server                      Status  Auth-port Acct-port Resolved IP Address
---------------------------------- ------- --------- --------- -------------------
9.9.150.16                         primary 1812      1813
cat6k-sup2                                 1812      1813      9.9.150.20
cat6k-sup3                                 1812      1813      9.9.150.21
Console> (enable)
```

Configuring the Authentication Failure VLAN

On a traditional 802.1X port, the switch does not provide access to the network until the supplicant that is connected to the port is authenticated by verifying its identity information with an authentication server. With the authentication failure VLAN feature, you can configure the authentication failure VLAN on a per-port basis and that after three failed 802.1X authentication attempts by the supplicant, the port is moved to the authentication failure VLAN where the supplicant can access the network.

Note Contrast an authentication failure VLAN with a guest VLAN. A guest VLAN enables the non-802.1X capable hosts to access the networks that use 802.1X authentication. You can use the guest VLANs while you are upgrading your system to support the 802.1X authentication. Typically, the guest VLANs support minimal services and provide minimal network access.

An authentication failure VLAN is independent of a guest VLAN. However, the guest VLAN can be the same VLAN as the authentication failure VLAN. If you do not want to differentiate between the non-802.1X capable hosts and the authentication failed hosts, you may configure both hosts to the same VLAN (either a guest VLAN or an authentication failure VLAN).

For more information, see the "Understanding How 802.1X Authentication for the Guest VLAN Works" section on page 40-9.

Authentication Failure VLAN Configuration Guidelines and Restrictions

This section describes the configuration guidelines and restrictions for configuring the authentication failure VLAN:

- After three failed 802.1X authentication attempts by the supplicant, the port is moved to the authentication failure VLAN where the supplicant can access the network. These three attempts introduce a delay of 3 minutes before the port is enabled in the authentication failure VLAN and the EAP success packet is sent to the supplicant (1 minute per failed attempt based on the default quiet period of 60 seconds after each failed attempt).
- The number of failed 802.1X authentication attempts is counted from the time of the linkup to the point where the port is moved into the authentication failure VLAN. When the port moves into the authentication failure VLAN, the failed-attempts counter is reset.
- Only the authenticated failed users are moved to the authentication failure VLAN.
- The authentication failure VLAN is supported only in the single-authentication mode (the default port mode).
- The authentication failure VLAN is not supported on a port that is configured as a unidirectional port.
- The supplicant's MAC address is added to the CAM table and only its MAC address is allowed on the authentication failure VLAN port. Any new MAC address appearing on the port is treated as a security violation.
- The authentication failure VLAN port cannot be part of an RSPAN VLAN or a private VLAN.

Note In software release 8.6(1) and later releases, a private VLAN and secondary VLAN can be configured as the guest VLAN or authentication failure VLAN. For more information, see the "Configuring 802.1X Authentication with Private VLANs" section on page 40-41.

- On multiple VLAN access ports (MVAPs), the authentication failure VLAN and the auxiliary VLAN cannot be the same VLAN.
- The authentication failure VLAN and port security features do not conflict with each other. Additionally, other security features, such as Dynamic ARP Inspection (DAI), Dynamic Host Configuration Protocol (DHCP) snooping, and IP source guard can be enabled and disabled independently on the authentication failure VLAN.
- An authentication failure VLAN is independent of a guest VLAN. However, the guest VLAN can be the same VLAN as the authentication failure VLAN. If you do not want to differentiate between the non-802.1X capable hosts and the authentication failed hosts, you may configure both hosts to the same VLAN (either a guest VLAN or an authentication failure VLAN).
- High availability is supported with an authentication failure VLAN.

Creating an Authentication Failure VLAN and Adding 802.1X Ports

To create an authentication failure VLAN and add 802.1X ports to the VLAN, perform this task in privileged mode:

Task	Command
Create an authentication failure VLAN and add 802.1X ports to the VLAN.	**set port dot1x** *mod/ports* **auth-fail-vlan** {**none** | *vlan*}

This example shows how to create the authentication failure VLAN (VLAN 81) and add port 3/33:

```
Console> (enable) set port dot1x 3/33 auth-fail-vlan 81
Port 3/33 Auth Fail Vlan is set to 81
Console> (enable)
```

This example shows how to display the authentication failure VLAN configuration:

```
Console> (enable) show port dot1x auth-fail-vlan
Auth-Fail-Vlan Status   Mod/Ports
-------------- -------- ------------------
81             active   3/33
none           none     1/1-2,2/1-2,3/1-32,3/34-48
Console> (enable)
```

This example shows how to clear a port from an authentication failure VLAN:

```
Console> (enable) set port dot1x 3/33 auth-fail-vlan none
Port 3/33 Auth Fail Vlan is cleared
Console> (enable)
```

This example shows how to list the active users and ports in an authentication failure VLAN:

```
Console> (enable) show dot1x auth-fail-users
Username       Mod/Port   Auth-Fail-Vlan
--------       --------   --------------
testuser       3/33       81
Console> (enable)
```

Configuring a RADIUS Server Failover

Before software release 8.4(1), when the active RADIUS server went down or was unreachable, the 802.1X authentication timed out before the backup RADIUS server could become active. With software release 8.4(1) and later releases, some RADIUS server timer values are now configurable and the **show radius** command has been enhanced to show the active RADIUS server.

Enter the following commands to prevent a RADIUS server failover:

- **set dot1x max-req**—Specifies the maximum number of times that the state machine retransmits an EAP-Request frame to the supplicant before it times out the authentication session; the valid values are from 1 to 10. The default is 2. An example is as follows:

```
Console> (enable) set dot1x max-req 8
dot1x max-req set to 8.
Console> (enable)
```

- **set dot1x server-timeout—**Specifies the time constant for the retransmission of packets by the back-end authenticator to the authentication server; the valid values are from 1 to 65535 seconds. When the authentication server does not notify the back-end authenticator that it received specific packets, the back-end authenticator waits a period of time (set by entering the **server-timeout** *seconds* parameter), and then retransmits the packets. The default is 30. An example is as follows:

```
Console> (enable) set dot1x server-timeout 100
dot1x server-timeout set to 100 seconds.
Console> (enable)
```

Ener the **show radius** command to display the RADIUS server configuration and to show which RADIUS server is active as follows:

```
Console> (enable) show radius
Active RADIUS Server:        81.81.81.20
RADIUS Deadtime:             1 minutes
RADIUS Key:                  cisco
RADIUS Retransmit:           2
RADIUS Timeout:              5 seconds
Framed-Ip Address Transmit:  Disabled

RADIUS-Server                    Status  Auth-port Acct-port Resolved IP Address
-------------------------------- ------- --------- --------- -------------------
81.81.81.20                      primary 1812      1813
10.6.89.200                              1812      1813
10.6.98.35                               1812      1813
Console> (enable)
```

Configuring 802.1X Authentication with Private VLANs

Note For more information on private VLANs, see the "Configuring Private VLANs on the Switch" section on page 11-19.

These sections describe how to configure 802.1X authentication with private VLANs:

Overview

Private VLANs provide a subnet conservation mechanism that allows a port to be conditionally operational in a VLAN pair without trunking. A private VLAN is composed of an associated primary VLAN and a secondary VLAN. A primary VLAN can participate in multiple private VLANs, with each primary VLAN having a different secondary VLAN associated with it. A secondary VLAN must belong to only one private VLAN. The secondary VLAN must be associated with only one primary VLAN. Secondary VLAN types are community, isolated, and two-way.

Before software release 8.6(1), an 802.1X port could not be configured in a private VLAN and a private VLAN port could not participate in 802.1X. With software release 8.6(1) and later releases, you can enable isolated private VLANs for 802.1X ports that are assigned to a guest VLAN through 802.1X authentication.

With guest VLANs, you might have ports from different customers residing in the same guest VLAN if the supplicant is identified as incapable of 802.1X before becoming 802.1X capable. With this behavior, the traffic from one customer might be accessible to every other customer. To avoid this situation, you can select different guest VLANs for each port; however, this action consumes multiple VLANs. With the isolated private VLAN approach, you can configure multiple ports in a VLAN pair and suppress the traffic interchange between the ports in the same secondary VLAN.

Port VLANs and 802.1X VLANs

With 802.1X, a port can be in a preauthenticated or post-authenticated state. In both states, the port is associated with a VLAN. The VLANs are referred to as the port VLAN and the 802.1X VLAN. The port VLAN is the VLAN of the port before a new VLAN has been assigned by 802.1X. The 802.1X VLAN of the port is the VLAN that is assigned to the port by 802.1X. The port operates in its port VLAN if it has not been enabled for 802.1X. Once the port is enabled for 802.1X, the port continues to be associated with its port VLAN although it stops forwarding traffic on the port VLAN. Once 802.1X assigns a new VLAN to the port, the port becomes operationally associated with the new VLAN (the 802.1X VLAN). If no VLAN is supplied by the RADIUS server, the port becomes operational in its port VLAN. A summary of port VLAN and 802.1X VLAN behavior follows:

- The port VLAN behavior is as follows:
  - Used by ports before 802.1X is enabled
  - Used as a nonoperational port VLAN after 802.1X is enabled

  - Used as a nonoperational port VLAN before the port reaches an 802.1X state (authenticated, guest VLAN, or authentication failure VLAN)
  - Used as an operational VLAN in the authenticated state if no VLAN is provided by the RADIUS server
  - Can be a private VLAN
- The 802.1X VLAN behavior is as follows:
  - Used as an operational port VLAN after 802.1X moves the port to an 802.1X state (authenticated, guest VLAN, or authentication failure VLAN)
  - Can be a private VLAN

Configuration Guidelines

This section provides the guidelines for configuring 802.1X authentication with private VLANs:

- No changes to the existing CLI are required for configuring 802.1X authentication with private VLANs.
- When you add an 802.1X port to a VLAN (RADIUS-assigned VLAN, guest VLAN, or authentication failure VLAN), the following checks are automatically made:
  - It is verified that the private VLAN is a secondary VLAN
  - It is verified that the secondary VLAN is associated to a valid primary VLAN

  If any of the checks fail, an error message is generated and the port is not placed in the private VLAN.
- Promiscuous ports and the sc0 interface cannot participate in 802.1X.
- When you configure an 802.1X port in a private VLAN, BPDU guard is automatically enabled, trunking is set to off, and the port retains these settings after being removed from the private VLAN.
- IP phone ports that support 802.1X cannot be private VLAN ports.

Configuring 802.1X Authentication with Private VLANs

These sections describe and provide examples on configuring 802.1X authentication with private VLANs:

Creating Private VLANs

This example shows how to create private VLANs:

```
Console> (enable) set vlan 800 pvlan-type primary
VTP advertisements transmitting temporarily stopped,
and will resume after the command finishes.
Vlan 800 configuration successful
Console> (enable) set vlan 801 pvlan-type community
VTP advertisements transmitting temporarily stopped,
and will resume after the command finishes.
Vlan 801 configuration successful
Console> (enable) set pvlan 800 801
Host mode set to enable for ports
BPDU guard set to enable for ports
Trunk mode set to off for ports
Successfully set association between 800 and 801.
Console> (enable) set vlan 400 pvlan-type primary
VTP advertisements transmitting temporarily stopped,
and will resume after the command finishes.
Vlan 400 configuration successful
Console> (enable) set vlan 401 pvlan-type isolated
VTP advertisements transmitting temporarily stopped,
and will resume after the command finishes.
Vlan 401 configuration successful
Console> (enable) set pvlan 400 401
Host mode set to enable for ports
BPDU guard set to enable for ports
Trunk mode set to off for ports
Successfully set association between 400 and 401.
Console> (enable) set vlan 200 pvlan-type primary
VTP advertisements transmitting temporarily stopped,
and will resume after the command finishes.
Vlan 200 configuration successful
Console> (enable) set vlan 201 pvlan-type twoway-community
VTP advertisements transmitting temporarily stopped,
and will resume after the command finishes.
Vlan 201 configuration successful
Console> (enable) set pvlan 200 201
Host mode set to enable for ports
BPDU guard set to enable for ports
Trunk mode set to off for ports
Successfully set association between 200 and 201.
Console> (enable)
```

Verifying the Private VLAN Configuration

This example shows how to verify the private VLAN configuration:

```
Console> (enable) show pvlan
Primary Secondary Secondary-Type Ports
------- --------- ---------------- ------------
200 201 twoway-community
400 401 isolated
800 801 community
Console> (enable)
```

Verifying the Pre-802.1X Port Settings

This example shows how to verify the pre-802.1X port settings:

```
Console> (enable) show port 2/2
* = Configured MAC Address
# = 802.1X Authenticated Port Name.
Port Name Status Vlan Duplex Speed Type
----- -------------------- ---------- ---------- ------ ---------- ------------
2/2 connected 999 a-half a-10 10/100BaseTX
Port AuxiliaryVlan AuxVlan-Status
----- ------------ --------------
2/2 none none
Port Security Violation Shutdown-Time Age-Time Max-Addr Trap IfIndex
----- -------- --------- ------------ -------- -------- -------- -------
2/2 disabled shutdown 0 0 1 disabled 61
Port Flooding on Address Limit Last-Src-Addr Vlan TimerType
----- ------------------------- ----------------- ---- ----------
2/2 Enabled - - Absolute
Port Num-Addr Secure-Src-Addr Vlan Age-Left Shutdown/Time-Left
----- -------- ----------------- ---- -------- ------------------
2/2 0 - - - - -
Port 802.1X Auth-State 802.1X Port-Status
----- ------------------ ------------------
2/2 force-authorized authorized
Port Broadcast-Limit Multicast Unicast Total-Drop Action
-------- --------------- --------- ------- -------------------- ------------
2/2 - - - 0 drop-packets
Port Send FlowControl Receive FlowControl RxPause TxPause
admin oper admin oper
----- -------- -------- --------- --------- ---------- ----------
2/2 off off off off 0 0
Port Status Channel Admin Ch
Mode Group Id
----- ---------- -------------------- ----- -----
2/2 connected off 2 0
Port Status ErrDisable Reason Port ErrDisableTimeout Action on Timeout
---- ---------- ------------------- ---------------------- -----------------
2/2 connected - Enable No Change
Port Align-Err FCS-Err Xmit-Err Rcv-Err UnderSize
----- ---------- ---------- ---------- ---------- ---------
2/2 0 0 0 0 0
Port Single-Col Multi-Coll Late-Coll Excess-Col Carri-Sen Runts Giants
----- ---------- ---------- ---------- ---------- --------- --------- ---------
2/2 3 3 0 3 0 0 0
Port Last-Time-Cleared
----- --------------------------
2/2 Mon Oct 3 2005, 12:42:26
Idle Detection
---------------
Console> (enable) show port dot1x 2/2
Port Auth-State BEnd-State Port-Control Port-Status
----- ------------------- ---------- ------------------- -------------
2/2 force-authorized idle force-authorized authorized
Port Port-Mode Re-authentication Shutdown-timeout Control-Mode
admin oper
----- ------------- ----------------- ---------------- ---------------
2/2 SingleAuth disabled disabled Both Both
Port Posture-Token Critical Termination action Session-timeout
----- ------------- -------- ------------------ ---------------
2/2 - NO NoReAuth -
Console> (enable)
```

Assigning Private VLANs to 802.1X

This example shows how to assign private VLANs to 802.1X:

```
Console> (enable) set port dot1x 2/2 port-control auto
Port 2/2 dot1x port-control is set to auto.
Trunking disabled for port 2/2 due to Dot1x feature.
Spantree port fast start option enabled for port 2/2.
Console> (enable) set port dot1x 2/2 initialize
Port 2/2 dot1x initializing ...
Console> (enable) set port dot1x 2/2 port-control auto
Port 2/2 dot1x port-control is set to auto.
Trunking disabled for port 2/2 due to Dot1x feature.
Spantree port fast start option enabled for port 2/2.
Console> (enable) set port dot1x 2/2 initialize
Port 2/2 dot1x initializing ...
Console> (enable) set port dot1x 2/2 guest-vlan 401
Port 2/2 Guest Vlan is set to 401
Console> (enable) set port dot1x 2/2 auth-fail-vlan 201
Port 2/2 Auth Fail Vlan is set to 201
Console> (enable)
```

Verifying the Config-Time 802.1X Private VLAN Settings

This example shows how to verify the config-time 802.1x private VLAN settings:

```
Console> (enable) show port 2/2
* = Configured MAC Address
# = 802.1X Authenticated Port Name.
Port Name Status Vlan Duplex Speed Type
----- -------------------- ---------- ---------- ------ ----------- ------------
2/2 connected 999 a-half a-10 10/100BaseTX
<...snip...>
Console> (enable) show port dot1x 2/2
Port Auth-State BEnd-State Port-Control Port-Status
----- ------------------ ---------- ------------------- -------------
2/2 connecting idle auto unauthorized
<...snip...>
Console> (enable) show pvlan
Primary Secondary Secondary-Type Ports
------- --------- ---------------- ------------
200 201 twoway-community
400 401 isolated
800 801 community
Console> (enable)
```

Verifying the Run-Time 802.1X-Assigned Private VLAN Settings

This example shows how to verify the run-time 802.1X-assigned private VLAN settings:

```
Console> (enable) show port dot1x guest-vlan
Guest-Vlan Status Mod/Ports
------------ -------- ------------------
401 active 2/2
none none 2/1,2/3-48,3/1-48,5/1-2
Console> (enable)  show port dot1x auth-fail-vlan
Auth-Fail-Vlan Status Mod/Ports
------------- -------- ------------------
201 active 2/2
none none 2/1,2/3-48,3/1-48,5/1-2
Console> (enable)
```

Example1: Guest VLAN is an isolated private VLAN (VLANs 400, 401)

```
Console> (enable) show port 2/2
* = Configured MAC Address
# = 802.1X Authenticated Port Name.
Port Name Status Vlan Duplex Speed Type
----- ------------------- ---------- ---------- ------ ---------- ------------
2/2 connected guest-400,401 a-half a-10 10/100BaseTX
<...snip...>
Console> (enable) show port dot1x 2/2
Port Auth-State BEnd-State Port-Control Port-Status
----- ------------------- ---------- ------------------- -------------
2/2 guest-vlan idle auto authorized
<...snip...>
Console> (enable) show pvlan
Primary Secondary Secondary-Type Ports
------- --------- ---------------- ------------
200 201 twoway-community
400 401 isolated 2/2
800 801 community
Console> (enable)
```

Example 2: 802.1X authentication failure VLAN is a two-way community private VLAN (VLANs 200, 201)

```
Console> (enable) show port 2/2
* = Configured MAC Address
# = 802.1X Authenticated Port Name.
Port Name Status Vlan Duplex Speed Type
----- ------------------- ---------- ---------- ------ ---------- ------------
2/2 connected fail-200,201 a-half a-10 10/100BaseTX
<...snip...>
Console> (enable)  clear port dot1x 2/2
dot1x port statistics cleared successfully for port
Console> (enable)  show port dot1x 2/2
Port Auth-State BEnd-State Port-Control Port-Status
----- ------------------- ---------- ------------------- -------------
2/2 auth-fail idle auto authorized
<...snip...>
Console> (enable) show pvlan
Primary Secondary Secondary-Type Ports
------- --------- ---------------- ------------
200 201 twoway-community 2/2
400 401 isolated
800 801 community
Console> (enable)
```

Example 3: 802.1X RADIUS-supplied VLAN is a community private VLAN (VLANs 800, 801)

```
Console> (enable) show port 2/2
* = Configured MAC Address
# = 802.1X Authenticated Port Name.
Port Name Status Vlan Duplex Speed Type
----- ------------------- ---------- ---------- ------ ---------- ------------
2/2 connected dot1x-800,801 a-half a-10 10/100BaseTX
Port AuxiliaryVlan AuxVlan-Status
----- ------------- --------------
2/2 none none
Port Security Violation Shutdown-Time Age-Time Max-Addr Trap IfIndex
----- -------- --------- ------------- -------- -------- -------- -------
2/2 disabled shutdown 0 0 1 disabled 61
Port Flooding on Address Limit Last-Src-Addr Vlan TimerType
----- ------------------------ ----------------- ---- ----------
```

```
2/2 Enabled - - Absolute
Port Num-Addr Secure-Src-Addr Vlan Age-Left Shutdown/Time-Left
----- -------- ----------------- ---- -------- ------------------
2/2 0 - - - - -
<...snip...>
Console> (enable) show port dot1x 2/2
Port Auth-State BEnd-State Port-Control Port-Status
----- ------------------- ---------- ------------------- -------------
2/2 authenticated idle auto authorized
<...snip...>
Console> (enable) show pvlan
Primary Secondary Secondary-Type Ports
------- --------- ---------------- ------------
200 201 twoway-community
400 401 isolated
800 801 community 2/2
Console> (enable)
```

Using the show Commands

Use these **show** commands to access the information about 802.1X authentication and its configuration:

- **show port dot1x ?**
- **show port dot1x**
- **show port dot1x statistics**
- **show dot1x**
- **show cam static**

To display the usage options for the **show port dot1x** command, perform this task in normal mode:

Task	Command
Display the usage options for the **show port dot1x** command.	**show port dot1x ?**

This example shows how to display the usage options for the **show port dot1x** command:

```
Console> (enable) show port dot1x ?
  guest-vlan                Show Port guest vlan information
  statistics                Show statistic information
  <mod>                     Module number
  <mod/port>                Module number and Port number(s)
  |                         Output modifiers
  <cr>
```

To display the values for all the parameters that are associated with the authenticator PAE and back-end authenticator on a specific port on a specific module, perform this task in normal mode:

Task	Command
Display the values for all the configurable and current state parameters that are associated with the authenticator PAE and back-end authenticator on a specific port on a specific module.	**show port dot1x** *mod/port*

This example shows how to display the values for all the parameters that are associated with the authenticator PAE and back-end authenticator on port 1 on module 3:

```
Console> (enable) show port dot1x 3/1
Port  Auth-State          BEnd-State Port-Control        Port-Status
----- ------------------- ---------- ------------------- -------------
 3/1  connecting          idle       auto                unauthorized

Port  Port-Mode     Re-authentication   Shutdown-timeout   Control-Mode
                                                           admin   oper
----- ------------ -----------------   ----------------   ---------------
 3/1  SingleAuth    enabled             disabled           In      Both
Console> (enable)
```

To display the statistics for the different types of EAP frames that are transmitted and received by the authenticator on a specific port on a specific module, perform this task in normal mode:

Task	Command
Display the statistics for the different types of EAP frames that are transmitted and received by the authenticator on a specific port on a specific module.	**show port dot1x statistics** *mod/port*

This example shows how to display the statistics for the different types of EAP frames that are transmitted and received by the authenticator on port 1 on module 3:

```
Console> (enable) show port dot1x statistics 3/1
Port   Tx_Req/Id Tx_Req Tx_Total Rx_Start Rx_Logoff Rx_Resp/Id Rx_Resp
-----  --------- ------ -------- -------- --------- ---------- -------
 3/1   43        0      43       0        0         0          0

Port   Rx_Invalid Rx_Len_Err Rx_Total Last_Rx_Frm_Ver Last_Rx_Frm_Src_Mac
-----  ---------- ---------- -------- --------------- -------------------
 3/1   2          0          2        0               00-00-00-00-00-00
Console> (enable)
```

To display the global 802.1X parameters, perform this task in normal mode:

Task	Command
Display the PAE capabilities, protocol version, system-auth-control, and other global dot1x parameters.	**show dot1x**

This example shows how to display the global 802.1X parameters:

```
Console> (enable) show dot1x
PAE Capability             Authenticator Only
Protocol Version           1
system-auth-control        enabled
max-req                    2
quiet-period               60 seconds
radius-accounting          disabled
radius-vlan-assignment     enabled
radius-keepalive state     enabled
re-authperiod              7200 seconds
server-timeout             30 seconds
shutdown-timeout           300 seconds
```

```
supp-timeout                30 seconds
tx-period                   30 seconds

Console> (enable)
```

To display the 802.1X authenticated MAC addresses, perform this task in normal mode:

Task	Command
Display the 802.1X authenticated MAC addresses.	**show cam static**

This example shows how to display the 802.1X authenticated MAC addresses. In this example, both 802.1X and port security are enabled:

```
Console> (enable) show cam static 8/17
* = Static Entry. + = Permanent Entry. # = System Entry. R = Router Entry.
X = Port Security Entry $ = Dot1x Security Entry

VLAN  Dest MAC/Route Des    [CoS]  Destination Ports or VCs / [Protocol Type]
----  ------------------    -----  -------------------------------------------
12    00-40-ca-13-ae-bf  $          8/17
17    00-30-94-c2-c3-c1  X          8/17
Total Matching CAM Entries Displayed  =2
Console> (enable)
```

CHAPTER 41

Configuring MAC Authentication Bypass

This chapter describes how to configure MAC authentication bypass on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

Note For information on configuring IEEE 802.1X authentication, see Chapter 40, "Configuring 802.1X Authentication."

Note For information on using port security to block input to an Ethernet, Fast Ethernet, or Gigabit Ethernet port when the MAC address of the station attempting to access the port is different from any of the MAC addresses that are specified for that port, see Chapter 38, "Configuring Port Security." That chapter also provides information on using port security to filter the traffic that is destined to or received from a specific host that is based on the host MAC address.

Note For information on configuring authentication, authorization, and accounting (AAA) to monitor and control access to the command-line interface (CLI) on the Catalyst 6500 series switches, see Chapter 39, "Configuring the Switch Access Using AAA."

Note For information on configuring network admission control, see Chapter 44, "Configuring Network Admission Control."

This chapter consists of these sections:

Understanding How MAC Authentication Bypass Works

These sections describe how MAC authentication bypass works on the Catalyst 6500 series switches:

Overview

MAC authentication bypass is an alternative to 802.1X that allows network access to devices (such as printers and IP phones) that do not have the 802.1X supplicant capability. MAC authentication bypass uses the MAC address of the connecting device to grant or deny network access.

To support MAC authentication bypass, the RADIUS authentication server maintains a database of MAC addresses for devices that require access to the network. MAC authentication bypass generates a RADIUS request with a MAC address in the calling-station-id (attribute 31) and service-type (attribute 6) with value 10.

To get the device's MAC address, the switch port needs to be in the forwarding state in a VLAN. If the port is not in the forwarding state in a VLAN, unicast traffic cannot enter or exit the switch. Because the switch port is brought up in the native VLAN with learning disabled on the port, the packets are redirected to the supervisor engine. When the supervisor engine sees a new MAC address, it installs a content-addressable memory (CAM) entry with a trap bit that is set to protect the supervisor engine from unnecessary flooding from that MAC address. The supervisor engine does not redirect further packets until the MAC authentication is finished. After a successful authentication, the RADIUS server sends a VLAN, and the port is moved to that VLAN. The trap entry is removed after a successful authentication. The port that is moved to the RADIUS server-specified VLAN behaves like any other switch port. If a MAC authentication fails, the port is moved into the authentication failure VLAN (if that VLAN is configured). (For information on authentication failure VLANs, see the "Configuring the Authentication Failure VLAN" section on page 40-38.)

Understanding Reauthentication of MAC Addresses

In the reauthentication mode, a port stays in the RADIUS server-specified VLAN and tries to reauthenticate itself. If the reauthentication is successful, the port stays in the RADIUS server-specified VLAN. If the reauthentication is not successful, the port is either moved back to the authentication failure VLAN (if that VLAN is configured), or the port is moved from its existing VLAN to an administratively configured VLAN. Periodic reauthentication can be attempted for the failed port. The failed port's MAC address CAM entry on the previously authenticated VLAN is removed and the initialization process forces the port to automatically go into the administratively configured VLAN where it attempts to reauthenticate itself. If reauthentication is successful, the port is moved to the RADIUS server-specified VLAN.

The RADIUS server-specified timers can also trigger reauthentication. RADIUS server attributes 27 and 29 control the reauthentication behavior. Attribute 27 (session timeout) specifies the time after which authentication should be tried again, and attribute 29 (termination action) specifies whether the behavior should be one of the following:

- Initialize—The existing session is disrupted until the reauthentication results are available.
- Reauthenticate—The existing session is not disrupted while reauthentication is attempted.

Understanding MAC Authentication Bypass States

This section describes the following MAC authentication bypass states:

- Waiting—In the waiting state, the switch waits to receive the MAC address that needs to be authenticated, learning is disabled, and the idle timer starts. The port is in the forwarding state to receive unicast traffic, and all Layer 2 entries on the port are cleared. The port transitions to the other state if there are other features configured but only after receiving an authentication result (the result could be success or failure). If traffic is not seen, the port remains in the waiting state.
- Authenticating—When the switch learns the port's MAC address from a redirected packet, the MAC authentication bypass state machine transitions to authenticating. In this state, the RADIUS request is built and sent to the RADIUS server and the switch waits for the RADIUS server response. If there is a successful authentication, the port moves to the authenticated state where the RADIUS server-specified VLAN is configured on the port, a static CAM entry is installed on the RADIUS server-specified VLAN, and the trap entries on the old VLAN are removed. If authentication fails, the port moves to the AuthFail State. If there is a RADIUS timeout or initialization, the port moves to the waiting state again.
- Authenticated—In the authenticated state, the RADIUS-received policy (VLAN) is configured on the port. The port then transitions to the waiting state in case there is an initialization and moves to the authenticating state if it receives a reauthenticate event. In the authenticated state, the trap entry on the port is removed from the old VLAN and the static CAM entry is installed on the new VLAN.
- AuthFail—In the AuthFail state, the port waits for "auth-fail-timeout" seconds before moving to the waiting state if no other features are configured. If fallback features are configured (such as web-based proxy authentication, 802.1X, or the authentication failure VLAN), the port moves to those states. A trap still exists in the AuthFail state, so a MAC address cannot authenticate itself again for auth-fail timeout seconds. When a port moves to the waiting state from the AuthFail state, the trap entries are cleared and the port starts the authentication process again.
- Finished—The finished state is entered after MAC authentication bypass fails to authenticate a host and if there are other features configured on the port that can potentially grant access (such as web-based proxy authentication, 802.1X, or the authentication failure VLAN). The finished state involves authorizing/bringing up the port and installing any policy required by the other features. For example, if the guest VLAN is configured, the port might be added to the guest VLAN. If web-based proxy authentication is configured, policies might be installed to allow Dynamic Host Configuration Protocol (DHCP), Domain Name System (DNS), and access control entries (ACEs) for HTTP redirection and so on. If other features are not configured, the port roams in the waiting, authenticating, AuthFail, and waiting states in case of an authentication failure or the port stays in the waiting state until it sees traffic.

Understanding MAC Authentication Bypass Events

This section describes the following MAC authentication bypass events:

- AuthenticateMac—This event is posted by the redirected packet processing component when it sees a MAC address on the port. This event is posted to the MAC authentication bypass state machine when it is in the waiting state.
- Initialize—This event is triggered by the CLI and can be received in any state. Upon reception of this event, the port is moved to the waiting state and any required cleanup is performed (such as unauthorizing the port, cleaning up any static/trap CAM entries, and so on).
- Reauthenticate—This event is received because either a session-timeout expired or because of a CLI trigger (executive command entered from the CLI). This event is accepted only when the port is in the authenticated state; otherwise, it is ignored. If this event is CLI driven, you are informed that the CLI can be accepted only if the port is in the authenticated state.
- Authentication success—This event is posted when there is an authentication success from the RADIUS server. This event, which is accepted only when the port is in the authenticating state, transitions the port to the authenticated state.
- Authentication failure—This event is posted when there is an authentication failure received from the RADIUS server. This event, which is accepted only when the port is in the authenticating state, transitions the port to the AuthFail state.
- RADIUS timeout—This event is received when the RADIUS server is not responding. This event, which is accepted only when the port is in the authenticating state, transitions the port to the waiting state after the maximum number of retries expire and the RADIUS server does not respond.
- AuthFail timeout—This event is received when the port is in the AuthFail state because of a RADIUS server authentication failure and there are no other potential features configured to bring the port up. This event transitions the port to the waiting state, and the port starts the authentication process again.
- Security violation—This event can be received in any state other than the waiting state. This event is posted if a second MAC address is seen on a port. The action taken for a security violation depends on the global violation mode configured and can either restrict a MAC address or shut down the port.

MAC Authentication Bypass Configuration Guidelines and Restrictions

This section provides the guidelines and restrictions for configuring MAC authentication bypass:

- Security violations—With MAC authentication bypass, only one host is supported per port. If more than one host appears on a port, it is a security violation and the port shuts down. With auxiliary VLAN ports, the one host per-port restriction only applies to hosts on the data VLAN; there is no restriction on the number of hosts on the auxiliary (voice) VLAN.
- Policy enforcement—MAC authentication bypass supports all policy enforcement mechanisms that are supported with 802.1X.
- DHCP snooping—MAC authentication bypass is independent of DHCP snooping. Until a MAC address successfully authenticates, no traffic is allowed from the MAC address (because of the trap entry), and the traffic that triggers the MAC authentication could be any type of traffic, including DHCP.

- 802.1X—MAC authentication bypass is an independent feature but when used in combination with 802.1X, acts as a fallback for authenticating MAC addresses. When both MAC authentication bypass and 802.1X are configured on a port, the port tries to authenticate using 802.1X. If the host does not respond to the EAPOL requests, instead of continuing the authentication attempts, the 802.1X port is moved to the MAC authentication bypass state, where the authentication is attempted using MAC authentication bypass.
- Authentication failure VLAN—When 802.1X authentication fails, irrespective of whether MAC authentication bypass is configured, if the authentication failure VLAN is configured, the port is moved to the authentication failure VLAN. The authentication failure VLAN is only for 802.1X authentication failed users and not a generic authentication failure VLAN for MAC authentication bypass. For more information on the authentication failure VLAN, see the "Configuring the Authentication Failure VLAN" section on page 40-38.
- Guest VLAN—The 802.1X guest VLAN and MAC authentication bypass work together but with some changes to the existing guest VLAN behavior. When both the MAC authentication bypass and the guest VLAN are configured and no Extensible Authentication Protocol over LAN (EAPOL) packets are received on a port, the 802.1X state machine is moved to the MAC authentication bypass state where it puts the port to forwarding in the native VLAN and disables learning. If the guest VLAN is not configured, the port remains in the MAC authentication bypass state where it waits for a MAC address on the port. For more information on guest VLANs, see the "Understanding How 802.1X Authentication for the Guest VLAN Works" section on page 40-9.
- Port security—When a new MAC address is redirected, the MAC authentication bypass function sees the MAC address before port security. If the MAC address is successfully authenticated, the port security feature is informed of the newly learned MAC address. In the inband path, the MAC authentication bypass function starts before any port security functions begin.
- Auxiliary VLANs—MAC authentication bypass is supported with auxiliary (voice) VLANs. MAC authentication bypass is restricted to those MAC addresses that appear on the port VLAN only. All IP phone MAC addresses that are learned through Cisco Discovery Protocol (CDP) are allowed on the auxiliary VLAN.
- Dynamic ARP Inspection (DAI)—Works with MAC authentication bypass.
- VLAN Membership Policy Server (VMPS)—MAC authentication bypass and VMPS are mutually exclusive features. The CLI prevents you from configuring both features at the same time.
- LAN port IP—When you configure both MAC authentication bypass and LAN port IP, the MAC authentication bypass function runs first. After authentication, the MAC authentication bypass feature triggers the LAN port IP function. The hosts in the LAN port IP exception list are authenticated using MAC authentication bypass (if configured) before access is provided.
- Web-based proxy authentication— When both MAC authentication bypass and web-based proxy authentication are configured on an interface, MAC authentication bypass starts before the web-based proxy authentication because MAC authentication bypass is a feature in Layer 2. A feature in Layer 2 is always attempted before a feature in Layer 3.
- RADIUS accounting—RADIUS accounting is supported.
- SNMP support—All required set and get calls are exported to SNMP. The SNMP support for MAC authentication bypass is scheduled for a future software release.
- High availability—High availability is supported. The MAC authentication bypass initial state and end state (authorized and unauthorized) of the port are synchronized to the standby supervisor engine. Intermediate states are not synchronized.

Configuring MAC Authentication Bypass

These sections describe how to configure MAC authentication bypass:

Enabling or Disabling MAC Authentication Bypass Globally

The default is disabled. To enable or disable MAC authentication bypass globally, perform this task in privileged mode:

Task	Command
Enable or disable MAC authentication bypass globally.	**set mac-auth-bypass** {**disable** | **enable**}

This example shows how to enable MAC authentication bypass globally:

```
Console> (enable) set mac-auth-bypass enable
Mac-Auth-Bypass enabled globally.
Console> (enable)
```

Enabling or Disabling MAC Authentication Bypass on a Port

When you enable or disable MAC authentication bypass on a port, you automatically enable or disable PortFast on the same port. The default is enabled.

To enable or disable MAC authentication bypass on a port, perform this task in privileged mode:

Task	Command
Enable or disable MAC authentication bypass on a port.	**set port mac-auth-bypass** *mod/port* {**disable** | **enable**}

This example shows how to enable MAC authentication bypass on a port:

```
Console> (enable) set port mac-auth-bypass 3/1 enable
MAC-Auth-Bypass successfully enabled on 3/1.
Console> (enable)
```

Initializing the MAC Authentication Bypass State for a Port

To initialize the MAC authentication bypass state for a port so that the port can participate in authentication again, perform this task in privileged mode:

Task	Command
Initialize the MAC authentication bypass state for a port so the port can participate in authentication again.	**set port mac-auth-bypass** *mod/port* **initialize**

This example shows how to initialize the MAC authentication bypass state for a port so that the port can participate in authentication again:

```
Console> (enable) set port mac-auth-bypass 3/1 initialize
Mac-Auth-Bypass successfully Initialized 3/1.
Console> (enable)
```

Reauthenticating the MAC Address for a Port

To reauthenticate the MAC address for a port, perform this task in privileged mode:

Task	Command
Reauthenticate the MAC address for a port.	**set port mac-auth-bypass** *mod/port* **reauthenticate**

This example shows how to reauthenticate the MAC address for a port:

```
Console> (enable) set port mac-auth-bypass 3/1 reauthenticate
Reauthenticating MAC address 00-00-00-00-00-01 on port 3/1 using Mac-Auth-Bypass.
Console> (enable)
```

Specifying the Shutdown Timeout Period

If there is a security violation on a port, the port shuts down. Use the global **set mac-auth-bypass shutdown-timeout** *seconds* command to specify the time (in seconds) the ports are shut down before they are automatically reenabled. The range is from 30 to 65535 seconds. The default is 60 seconds. If you specify a shutdown timeout period of 0 seconds, the automatic port enable function is disabled and you will have to reenable the ports manually.

To specify the shutdown timeout period, perform this task in privileged mode:

Task	Command
Specify the shutdown timeout period.	**set mac-auth-bypass shutdown-timeout** *seconds*

This example shows how to specify the shutdown timeout period:

```
Console> (enable) set mac-auth-bypass shutdown-timeout 40
Shutdown Timeout set to 40 seconds.
Console> (enable)
```

Specifying the AuthFail Timeout Period

The global **set mac-auth-bypass auth-fail-timeout** *seconds* command specifies the time (in seconds) that ports wait in the authentication failure (AuthFail) state before trying authentication again. The range is from 5 to 65535 seconds. The default is 60 seconds.

To specify the AuthFail timeout period, perform this task in privileged mode:

Task	Command
Specify the AuthFail timeout period.	**set mac-auth-bypass auth-fail-timeout** *seconds*

This example shows how to specify the AuthFail timeout period:

```
Console> (enable) set mac-auth-bypass auth-fail-timeout 60
Authfail Timeout set to 60 seconds.
Console> (enable)
```

Specifying the Reauthentication Timeout Period

The global **set mac-auth-bypass reauth-timeout** *seconds* command specifies the time (in seconds) that elapse before reauthentication is triggered after global reauthentication is enabled. The range is from 300 to 65535 seconds. The default is 3600 seconds.

To specify the reauthentication timeout period, perform this task in privileged mode:

Task	Command
Specify the reauthentication timeout period.	**set mac-auth-bypass reauth-timeout** *seconds*

This example shows how to specify the reauthentication timeout period:

```
Console> (enable) set mac-auth-bypass reauth-timeout 400
Reauth Timeout set to 400 seconds.
Console> (enable)
```

Enabling or Disabling Reauthentication

Enabling the global **set mac-auth-bypass re-authentication** command returns all MAC authentication bypass values to their defaults. The default is disabled.

To enable or disable MAC authentication bypass reauthentication globally, perform this task in privileged mode:

Task	Command
Enable or disable MAC authentication bypass reauthentication globally.	**set mac-auth-bypass reauthentication** {**disable** | **enable**}

This example shows how to enable MAC authentication bypass reauthentication globally:

```
Console> (enable) set mac-auth-bypass reauthentication enable
Global reauthentication mode enabled.
Console> (enable)
```

Specifying the Security Violation Mode

If there is a security violation on a port, the port goes into restricted mode or is shut down. In restricted mode, the MAC address that causes the security violation is added as a trap entry into the forwarding table. The default is shutdown.

To specify the security violation mode globally, perform this task in privileged mode:

Task	Command
Specify the security violation mode globally.	**set mac-auth-bypass violation** {**restrict** | **shutdown**}

This example shows how to specify “restricted” for the security violation mode:

```
Console> (enable) set mac-auth-bypass violation restrict
Mac-Auth-Bypass security violation mode set to restrict.
Console> (enable)
```

Enabling or Disabling MAC Authentication Bypass RADIUS Accounting

The default is disabled. To enable or disable MAC authentication bypass RADIUS accounting, perform these tasks in privileged mode:

Task	Command
Enable or disable MAC authentication bypass RADIUS accounting.	**set mac-auth-bypass radius-accounting** {**disable** | **enable**}
Verify the MAC authentication bypass RADIUS accounting state.	**show mac-auth-bypass config**

This example shows how to enable MAC authentication bypass RADIUS accounting:

```
Console> (enable) set mac-auth-bypass radius-accounting enable
Radius Accounting for MacAuth enabled.
Console> (enable)
```

This example shows how to verify the MAC authentication bypass RADIUS accounting state:

```
Console> (enable) show mac-auth-bypass config
Mac-Auth-Bypass Global Config
-----------------------------
Mac-Auth-Bypass Status      = Enabled
AuthFail Timeout            = 60
RadiusAccounting            = Enabled
Reauthentication            = Disabled
Reauth Timeout              = 3600
Shutdown Timeout            = 60
Violation mode              = Shutdown
Console> (enable)
```

Configuring a PVLAN on a MAC Authentication Bypass-Enabled Port

To configure a PVLAN on a MAC authentication bypass-enabled port, perform these tasks in enabled mode:

Task	Command
Configure MAC authentication bypass.	**set mac-auth-bypass {enable | disable}**
Configure a PVLAN on a MAC authentication bypass-enabled port.	**set port mac-auth-bypass** *mod/port* **{enable | disable}**
Configure the PVLAN on the port.	**set pvlan** *primary vlan secondary vlan mod/port*

This example shows how to configure MAC authentication bypass-enabled on PVLAN port 3/13:

```
Console> (enable) set mac-auth-bypass enable
Mac-Auth-Bypass enabled globally.
Console> (enable) set port mac-auth-bypass 3/13 enable
Mac-Auth-Bypass successfully enabled on port(s) 3/13
Console> (enable) show port mac-auth-bypass 3/13
Port  Mac-Auth-Bypass State MAC Address       Auth-State         Vlan
----- --------------------- ----------------- -----------------  -----
 3/13 Enabled               00-00-00-00-00-00 waiting            25

Port  Termination action Session Timeout Shutdown/Time-Left
----- ------------------ --------------- ------------------
 3/13 initialize         3600            NO       -

Port  PolicyGroups
----- --------------------------------------------------
 3/13  -

Port  Critical Critical-Status
----- -------- ---------------
 3/13 Enabled  -
Console> (enable) set pvlan 12 30 3/13
Host mode set to enable for port 3/13.
BPDU guard set to enable for port 3/13.
Trunk mode set to off for ports 3/13
```

```
Successfully set the following ports to Private Vlan 12,30:
3/13
Console> (enable)
```

Configuring MAC Authentication Bypass on a PVLAN Port

To configure MAC authentication bypass on a PVLAN port, perform these tasks in enabled mode:

Task	Command
Configure the PVLAN on the port.	**set pvlan** *primary vlan secondary vlan mod/port*
Configure MAC authentication bypass.	**set mac-auth-bypass {enable | disable}**
Configure a PVLAN on a MAC authentication bypass-enabled port.	**set port mac-auth-bypass** *mod/port* **{enable | disable}**

This example shows how to configure MAC authentication bypass-enabled on PVLAN port 3/13:

```
Console> (enable) set pvlan 12 30 3/13
Successfully set the following ports to Private Vlan 12,30:
3/13
Console> (enable) set mac-auth-bypass enable
Mac-Auth-Bypass enabled globally.
Console> (enable) set port mac-auth-bypass 3/13 enable
Mac-Auth-Bypass successfully enabled on port(s) 3/13
Console> (enable)
```

Displaying MAC Authentication Bypass Information

The **show port mac-auth-bypass** {*mod/port*} command displays the port state (such as authenticating, authenticated, and waiting to learn the source MAC address), and the port's RADIUS server-specified VLAN.

To display MAC authentication bypass information, perform these tasks in normal mode:

Task	Command
Display MAC authentication bypass information for all ports in the switch that have MAC authentication bypass enabled or for a single port.	**show port mac-auth-bypass** [*mod/port*]
Display MAC authentication bypass information for all ports in the switch that have MAC authentication bypass enabled or for the port with the specified MAC address.	**show mac-auth-bypass** {**all** | **config** | *mac_address*}

This example shows how to display MAC authentication bypass information for port 5/1:

```
Console> (enable) show port mac-auth-bypass 5/1
Port  Mac-Auth-Bypass State MAC Address       Auth-State     Vlan
----- --------------------- ----------------- -------------- -----
 5/1  Disabled              -                 -              1
```

```
Port  Termination action Session Timeout Shutdown/Time-Left
----- ------------------ --------------- ------------------
 5/1  -                  3600            -        -
Console> (enable)
```

This example shows how to display MAC authentication bypass information for all ports in the switch that have MAC authentication bypass enabled:

```
Console> (enable) show mac-auth-bypass all

Port  Mac-Auth-Bypass State MAC Address       Auth-State     Vlan
----- -------------------- ----------------- -------------- -----
 5/1  Disabled             -                 -              1
 5/2  Enabled              00-00-00-00-00-00 waiting        1
 5/3  Enabled              00-00-00-00-00-00 waiting        1
 5/4  Enabled              00-00-00-00-00-00 waiting        1
 5/5  Enabled              00-00-00-00-00-00 waiting        1
 5/6  Enabled              00-00-00-00-00-00 waiting        1
 5/7  Enabled              00-00-00-00-00-00 waiting        1
 5/8  Enabled              00-00-00-00-00-00 waiting        1
.
.
.
Port  Termination action Session Timeout Shutdown/Time-Left
----- ------------------ --------------- ------------------
 5/1  -                  3600            -        -
 5/2  reauthenticate     3600            NO       -
 5/3  reauthenticate     3600            NO       -
 5/4  reauthenticate     3600            NO       -
 5/5  reauthenticate     3600            NO       -
 5/6  reauthenticate     3600            NO       -
 5/7  reauthenticate     3600            NO       -
 5/8  reauthenticate     3600            NO       -
.
.
.
Console> (enable)
```

Displaying the MAC Authentication Bypass Global Configuration

The **show mac-auth-bypass config** command displays MAC authentication bypass global configuration settings including the timer values, violation mode, global reauthentication mode, and so on.

To display MAC authentication bypass global configuration settings, perform this task in normal mode:

Task	Command
Display MAC authentication bypass global configuration settings.	**show mac-auth-bypass** {**all** | **config** | *mac_address*}

This example shows how to display MAC authentication bypass global configuration settings:

```
Console> (enable) show mac-auth-bypass config
Mac-Auth-Bypass Global Config
-----------------------------
Mac-Auth-Bypass Status      = Enabled
AuthFail Timeout            = 60
RadiusAccounting            = Enabled
Reauthentication            = Disabled
Reauth Timeout              = 3600
```

```
Shutdown Timeout            = 60
Violation mode              = Shutdown
Console> (enable)
```

Configuring MAC Authentication Bypass with ACL Assignments

MAC authentication bypass(MAB)-enabled ports support ACL assignments similar to 802.1X-enabled ports. For more information, see "Configuring 802.1X with ACL Assignments" section on page 40-26.

The ACLs must be predefined and committed on the switch. ACL mapping by MAB is a runtime configuration and does not reflect in the NVRAM. The mapping is removed when the MAB static CAM entry is removed or at reauth, if the RADIUS sends a different or no ACL to map.

Configuring MAC Authentication Bypass with QoS ACLs

MAC authentication bypass-enabled ports support ACLs sent by RADIUS and QoS policies-based authentication similar to QoS policies on 802.1X-enabled ports. For more information, see "Configuring 802.1X with QoS ACLs" section on page 40-29.

When configuring MAB with QoS ACLs, follow these guidelines:

- The QoS ACLs must be predefined and committed on the switch.
- If more than one QoS ACL of the same attribute type (*invacl*, *outvacl*, or *inpacl*) is sent to the MAB port, only the first ACL for an attribute type is configured.
- The minimum acceptable reauthentication timeout for MAB has been reduced to 30 from 300 seconds. The default is 30 seconds.
- Dynamically applied QoS ACLs cannot be removed using commands. They are automatically removed when MAB initializes.

This example shows how to display the QoS ACLs information for a MAB-enabled port:

```
Console (enable)> show port mac-auth-bypass 3/13
Port  Mac-Auth-Bypass State MAC Address       Auth-State         Vlan
----- -------------------- ----------------- -----------------  -----
3/13 Enabled               00-11-22-33-01-87 authenticated      391

Port  Termination action Session Timeout Shutdown/Time-Left
----- ------------------ --------------- ------------------
3/13 initialize         3600            NO       -

Port  PolicyGroups
----- ------------------------------------------------------------------
3/13  -

Port  Security ACL                     Sec ACL Type      QoS ACL Type
----- ------------------------------- ----------------- ----------------
3/13 my_security_pacl                 Pacl              Vacl

Port  QoS Ingress Policy               QoS Egress Policy
----- ------------------------------- ---------------------------------
3/13 my_qos_invacl                    my_qos_outvacl

Port  Critical Critical-Status
----- -------- ---------------
3/13 Disabled -
```

Configuring Agentless Hosts for NAC Auditing with MAB

Note Catalyst 6500 series software release 8.7(1) and later releases support NAC auditing for agentless hosts with MAC authentication bypass enabled. This feature is not supported on Supervisor Engine 2 and for agentless hosts with 802.1X enabled on other supervisor engines.

These sections describe how to audit agentless hosts with MAC authentication bypass enabled:

NAC Agentless Hosts Auditing Overview

Network Admission Control (NAC) enables the posture of an endpoint device to check for compliance with the security policy before the device accesses the protected areas of a network. NAC allows the host posture to be determined using either the Posture Agent (PA), or using the audit server for agentless hosts if the PA is not installed on the host.

Several methods in NAC allow network access to hosts that cannot perform authentication because of the lack of posture agent. Agentless hosts are such as printers, scanners, and hosts with unsupported operating systems. One method is to use an external audit server with agentless hosts connected to MAC authentication bypass-enabled NAD ports. To determine the posture, the MAC address must be registered, and shared profiles and admission policies must be created on a centralized ACS server.

Audit servers have the ability to probe and scan the clientless devices for security compliance, vulnerabilities, and threats. The result of the audit sever can influence access servers to make host specific network access policy decisions rather than enforce a common restrictive policy for all nonresponsive hosts.

Configuring the Switch

For the NAC audit server to determine the posture of agentless hosts, perform these tasks in privileged mode:

	Task	Command
Step 1	Enable MAC authentication bypass globally on the switch.	**set mac-auth-bypass enable**
Step 2	Enable MAC authentication bypass reauthentication on the switch.	**set mac-auth-bypass reauthentication enable**
Step 3	Enable MAC authentication bypass on a per-port basis.	**set port mac-auth-bypass** *mod/port* **enable**

When configuring the switch, follow these guidelines:

- The switch must have a RADIUS configuration and be connected to the Cisco Secure ACS server.
- If the audit configuration is removed from the network access profile (NAP) of MAB, the port needs to be reinitialized.
- The session-timeout value must be greater than the time required for the DACL to download all the ACLs and it must be determined based on other audit requirements.

Configuring the Cisco Secure ACS Server

For auditing agentless hosts, the switch must be connected to a Cisco Secure ACS server and a third-party NAC audit server such as Qualys. When the audit server is installed and running, configure the audit server information on the ACS server. Cisco Secure ACS server 4.1 or later is required for this feature to function properly.

To configure the ACS server with NAC agentless hosts and NAC audit server information, perform these steps:

Step 1 Import the NAC audit vendor trusted root CA to the certificate store on ACS by using the **CSUtil** tool.

Step 2 Import an audit device-type attribute file for the NAC audit server by using **CSUtil**.

Step 3 Import NAC attribute-value pairs for the audit vendor by using **CSUtil**.

Step 4 Enable posture validation on the ACS.

Step 5 Configure the external audit server on ACS using the external posture validation audit server setup page on the ACS.

Step 6 Define shared profile components.

Step 7 Configure network access profile (NAP) authorization policy.

Note In the NAP profile, configure MAB, specify the audit server, DACL or shared RAC policies to be applied for the various posture tokens, and the fail open policy to be applied when the audit server cannot communicate with the host.

Step 8 Configure the hosts to be audited, and device-type retrieval and mapping for audit vendors who have a device attribute in the RADIUS dictionary using the external audit server posture validation setup page on the ACS.

Step 9 Set up a device group policy on the ACS.

For more information about auditing agentless hosts, and detailed steps to complete each of these tasks, refer to the following documents:

- *Configuration Guide for CISCO Secure ACS*
- *NAC Framework Configuration Guide*
- *NAC Audit Vendor Configuration Guide*

Installing and Configuring the NAC Audit Server

For information regarding installing and configuring the NAC audit server, refer to the NAC Audit vendor documentation shipped with the audit server. Ensure that the audit server is physically connected to the switch before you install and configure it.

Displaying the Agentless Host Posture Tokens

The agentless host is evaluated on the number of vulnerabilities found and their severity levels. This vulnerability information is taken from the cached audit report, and the posture token is determined by the evaluation method settings on the NAC audit server.

The agentless host can hold any of the following posture agents:

- **Infected**—When at least one Severity 5 vulnerability is detected. Infected host audit reports are cached and expire after 5 minutes.
- **Quarantine**—When at least one Severity 4 vulnerability is detected. Quarantine host audit reports are cached and expire after 10 minutes.
- **Check-up**—When at least one Severity 3 vulnerability is detected. Check-up host audit reports are cached and expire after 1 hour.
- **Healthy**—When no severity 5, 4, or 3 vulnerabilities are detected. Healthy host audit reports are cached and expire after 24 hours.
- **Unknown**—When nonexisting and dead hosts do not respond to probes. Unknown host audit reports are cached and expire after 12 hours.

Note There will be a delay in traffic because of auditing and the host would hold a transition posture token during such delay.

This example shows how to display the posture tokens of a MAC authentication bypass-enabled port:

```
Console> (enable) show port mac-auth-bypass 6/25
Port  Mac-Auth-Bypass State   MAC Address  Auth-State   Vlan
----  ----------------------  -----------  ----------   -------
6/25  Disabled                -            -            5

Port  Termination action   Session Timeout  Shutdown/Time-Left
----  -------------------  ---------------  ------------------
6/25  -                    3600             NO       -

Port  PolicyGroups
----  ------------------------------------------------------
6/25  -

Port  Critical       Critical-Status
----  -----------    ---------------
6/25  Disabled       -

Port  Session-id
----  --------------------------------
6/25  000015a90000099a000019ba000003e1

Port  Posture -Token   Url-Redirect
----  --------------   -------------------
6/25  Healthy          http://10.76.255.100:2002
```

Interaction of Agentless Host Audit with Security Features

This section describes the behavior of NAC audit with other security features:

- 802.1X—When ACS audits a 802.1X-authenticated port, it checks for the MAB configuration. ACS audits the port only if MAB is enabled, otherwise it considers the port to be part of a guest VLAN.
- MAB—Regardless of how MAB is triggered, audit runs unless MAB fails.
- Layer 3 features—Not affected by MAB-enabled agentless host audit.
- Critical-Auth—Because there is no RADIUS server, no interaction is possible and the old posture (if any) is maintained.
- PVLAN—No effect.

CHAPTER 42

Configuring Web-Based Proxy Authentication

This chapter describes how to configure web-based proxy authentication on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

Note For information on configuring IEEE 802.1X authentication, see Chapter 40, "Configuring 802.1X Authentication."

Note For information on configuring MAC address authentication bypass, see Chapter 41, "Configuring MAC Authentication Bypass."

Note For information on using port security to block input to an Ethernet, Fast Ethernet, or Gigabit Ethernet port when the MAC address of the station attempting to access the port is different from any of the MAC addresses that are specified for that port, see Chapter 38, "Configuring Port Security." That chapter also provides information on using port security to filter the traffic that is destined to or received from a specific host that is based on the host MAC address.

Note For information on configuring authentication, authorization, and accounting (AAA) to monitor and control access to the command-line interface (CLI) on the Catalyst 6500 series switches, see Chapter 39, "Configuring the Switch Access Using AAA."

Note For information on configuring network admission control, see Chapter 44, "Configuring Network Admission Control."

This chapter consists of these sections:

Understanding How Web-Based Proxy Authentication Works

The Catalyst 6500 series switch provides web-based proxy authentication in cases where the network client does not have IEEE 802.1X host support. Web-based proxy authentication is authentication through a standard web-based interface (HTTP/HTTPS) of the front-end systems for client identity and credential input.

With 802.1X port-based authentication, a *supplicant* is required to provide access to the LAN and switch services and respond to requests from the switch.

Note 802.1X uses the term *supplicant* for *client* or *host*. In this publication, we use *host* instead of *supplicant* because *host* is used in the Catalyst 6500 series CLI syntax.

Web-based proxy authentication supports full 802.1X authentication and provides support for nonhost-capable clients.

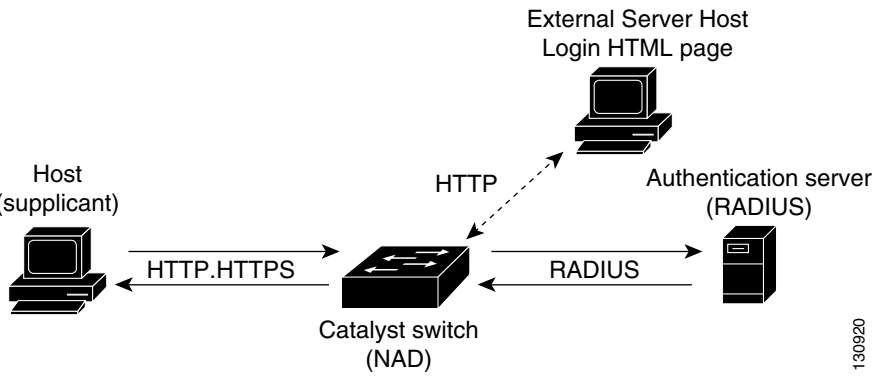
See the "Configuring 802.1X Authentication" chapter for 802.1X authentication information.

These sections describe how web-based proxy authentication works:

Device Roles

Web-based proxy authentication provides authentication through a standard web-based interface as shown in Figure 42-1.

Figure 42-1 Device-integrated Web-Based Proxy Authentication



Host (Supplicant)—Once you enable web-based proxy authentication, the host can request access to the LAN and switch services and respond to requests from the switch.

Switch—The network access device (NAD), or the Catalyst 6500 series switch, hosts all the HTML pages when the host is connected to the switch port that is enabled for web-based authentication. The login web page is hosted on an external web server. When the host receives an IP address, the web browser is opened. When an HTTP packet is intercepted, the URL redirects the client to the location of the external login web page URL. You can directly download the login page from the external web server. If an external login page is not configured, a default login page is sent.

The credentials, which include the username, password, and any other options, are input at the host. The host then submits the page. The Catalyst 6500 series switch intercepts this HTTP POST request, establishes the connection, and retrieves the POST request. Once the POST request is retrieved, the Catalyst 6500 series switch processes the web page and extracts the credentials.

Authentication server—The server validates the identity of the host and notifies the switch if the host is authorized to access the LAN and switch services. Because the switch acts as the proxy, the authentication service is transparent to the host. The Remote Authentication Dial-In User Service (RADIUS) security system with Extensible Authentication Protocol (EAP) extensions is the only supported authentication server; it is available in Cisco Secure Access Control Server version 3.0. RADIUS operates in a client/server model in which secure authentication information is exchanged between the RADIUS server and one or more RADIUS clients.

Authentication Initiation and Message Exchange

The host is connected to the switch port that needs to perform web authentication. When the host receives an IP address, a web browser is opened. When an HTTP packet is intercepted, the network access device (NAD) establishes the TCP connection with the host and sends the login page if it is stored locally on the switch, or the URL redirects the client to the location of the external login page URL so that the client directly downloads the login page from the external web server.

You can enter the credentials including the username, the password, and any other options and submit the page from the host. The NAD intercepts this information, establishes a connection, and retrieves the request. The NAD then processes the web page information and extracts the credentials, which are authenticated using an external AAA server (RADIUS). Based on the results of the authentication, the NAD sends an authentication success or an authentication failure page to the client as follows:

- If the authentication succeeds, NAD updates the policy-based ACLs (PBACLs) with the new policy groups that are received from RADIUS for this host. The URL redirects the client to the URL that the client initially tried to access.
- If the authentication fails, the NAD sends a Login-fail web page to the host, that lists the login-fail and input fields. If an external login-fail page is specified, the NAD URL redirects the client to the location of the login-fail page.

If the login or login-fail page points to an external web server, then the default policy allows HTTP access to this web server even before the host is authenticated.

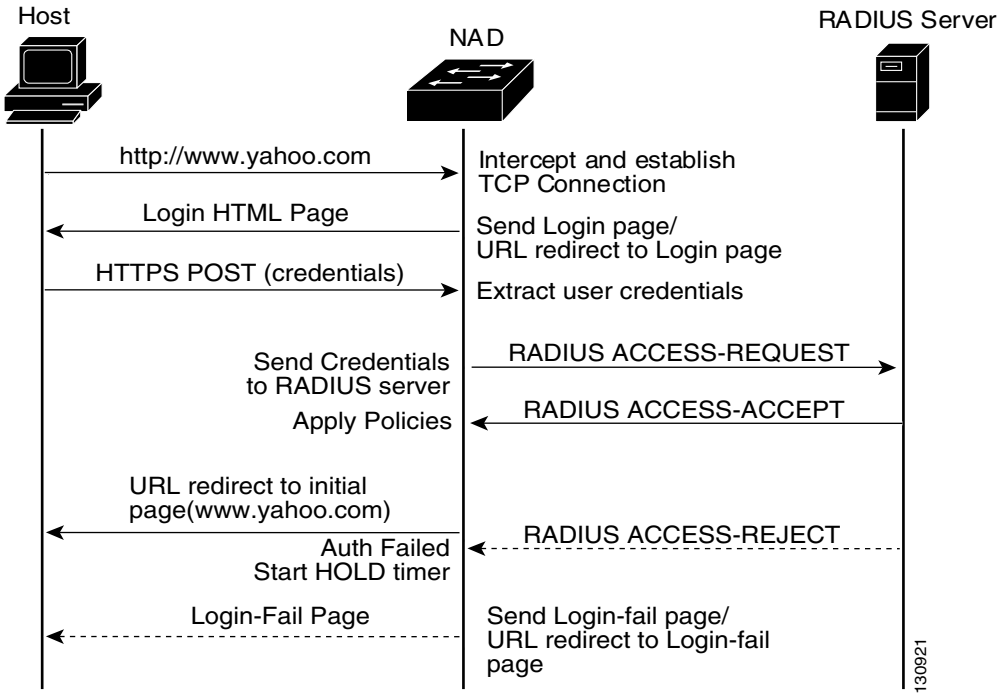
Note If the default policy does not allow HTTP access and external pages, the client cannot download these web pages and web-based proxy authentication does not work.

The login/login-fail page contains the same variable names and types for the username, passwords, and any other fields that the NAD is programmed to process. A default page is used in the absence of a configured login file on the NAD.

The initial login page is sent using HTTP and HTTPS and is used for submitting user credentials to the Catalyst 6500 series switch. Until HTTPS functionality is fully operational, HTTP is used for credential transfer.

The authentication initiation and message exchange sequence of events is shown in Figure 42-2.

Figure 42-2 Authentication Initiation and Message Exchange



Host Detection and HTTP Traffic Interception

Address Resolution Protocol (ARP) inspection is used to address hosts with static IP addresses assigned. When ARP inspection receives any ARP request on a web-authenticated port, web-based proxy authentication is triggered for a host IP address. If web-based proxy authentication is enabled on a port that is operational, the web-based proxy authentication is initiated on all IP addresses in the Dynamic Host Configuration Protocol (DHCP) snooping table. If a DHCP snooping entry does not exist, web-based proxy authentication is not triggered until a DHCP snooping entry is created or an ARP request is received.

Once the host is detected, the HTTP traffic from the host is intercepted and redirected to the supervisor engine. This process is called URL redirection. To configure URL redirection, you must configure an ACL to redirect all TCP port 80 ingress traffic to the supervisor engine by entering the **permit url-redirect** command. The **permit url-redirect** command redirects all TCP port 80 traffic to the supervisor engine.

Any ACL that is mapped to a port/port-VLAN with this access control entry (ACE) redirects all the HTTP/HTTPS protocol packets that match the ACE criteria to the supervisor engine.

If you enable web-based proxy authentication without configuring this ACE, the HTTP/HTTPS packets are not intercepted and authentication is not initiated. The host traffic in this scenario is controlled by the default policy that is configured on the port/VLAN.

Web-based proxy authentication notifies URL redirection through the software when a new host is detected and provides a callback function for the intercepted HTTP packets.

Access Control

Access control is provided by PBACLs. You can use a PBACL to configure the intercept, default, and host-specific ACLs.

PBACLs are mapped to a VLAN. All ports in the VLAN have the default access specified by the PBACL only.

Note We recommend that you enable web-based proxy authentication on all ports in the VLAN.

Supported HTML Pages for Web-Based Proxy Authentication

This section describes the following HTML pages required to support web-based proxy authentication:

- Login Page, page 42-5
- Success Page, page 42-6
- Login-Fail Page, page 42-6

Login Page

The login page displays at the client in response to the first URL intercept. Web-based proxy authentication supports a customized login page. The customized login page needs the URL (HTTP only) of the login page. The login page contains the following fields:

- Username—character string
- Password—character string
- Radio button with the following options:
  - I have a registered account
  - I have a Guest account
  - I don’t have an account

Note The submit button in the login page points to the HTTPS URL if the switch supports the HTTPS protocol. If HTTPS is not supported, the login page points to the HTTP URL.

A default login page is sent if a customized login page is not specified.

Success Page

The success page is an auto-redirection page that automatically redirects the client browser to the URL that you tried to access initially. The success page is not displayed, it is auto-redirected to the original page.

Login-Fail Page

The login-fail page, which contains information about the authentication failure, allows you to reenter the credentials if an authentication fails. The login-fail page contains all the fields of a login page and information about the authentication failure.

Note An authentication failure can occur if you enter the wrong username/password or if you select the "I don't have an account" option and the switch does not have default policies configured for this option.

A default login-fail page displays if a customized login-fail page is not specified.

Multiple Hosts Per Port

Web-based proxy authentication authenticates all the hosts (IP addresses) that are seen on the port. The maximum number of hosts supported on a port is 32.

A new web-based proxy authentication state is created for every new host that is seen on the port. If you enable web-based proxy authentication on a port that has multiple DHCP bindings already created, web-based proxy authentication is initialized for all IP addresses.

High Availability

Web-based proxy authentication supports high availability. Only the information from the authenticated hosts is synchronized to the standby supervisor engine. All authenticated hosts remain authenticated upon a switchover. The notification from unauthenticated or authentication in-progress hosts is not synchronized. Web-based proxy authentication initializes these hosts upon a switchover and authentication restarts.

For example, if you entered the credentials and submitted a login page, and the switch sent the credentials to RADIUS and was waiting for a response, if the switchover occurs, the credentials that you entered are lost and the login page is resent to the host when you try to access any URL. You must reenter the credentials.

Host State

The host state determines if the host is granted access to the network. The host states are as follows:

- Initialize—Occurs when the IP address of the host is registered with URL redirection for redirecting any HTTP packet from this host to the supervisor engine. After receiving the first HTTP-intercepted packet, the host state changes to the connecting state.
- Connecting—Occurs when the login page displays to the client and waits for a response from the client. When the host receives the HTTP POST response, the host state changes to the authenticating state.

- Authenticating—Occurs when the host response (HTTP POST message) is processed and you can extract the credentials. The credentials are then authenticated with the external RADIUS server as follows:
  - If the HTTP response fails, the state changes to the Parse-error state. For example, this state could occur if the external login page specified does not conform to the variable/field names that the switch is programmed to process.
  - If the authentication succeeds, the state changes to the Authenticated state. If the authentication fails and the retry count is less than the maximum configured, the state changes to the Authentication-Fail state or the Held state.
- Authenticated—Occurs upon a successful authentication. In the Authenticated state, the RADIUS attributes are processed and the policies are applied and returned to the host. No HTTP packets are intercepted and redirected to the supervisor engine. The state changes to the session-timeout state when the session timer expires.
- Authentication-Fail—Occurs when RADIUS sends an accept-reject and a Login-Fail page with authentication failure information embedded in it.
- Parse-Error—Occurs upon a failure to extract user credentials from the HTTP Post message. A standard login page that is stored internally in the network access device is sent to the client. The state changes to the Authenticating state when the host receives a HTTP Post response.
- Session-timeout—Occurs when the session timer expires. The user policies are removed and the state changes to the Initialize state.
- Held—Occurs when the authentication retry count exceeds the configured maximum number of retry attempts. No HTTP packets are intercepted. Port initialize and DHCP binding removal removes the Held state designation.

Interaction with Other Features

Web-based proxy authentication interacts with these features as follows:

- DHCP snooping—You can enable web-based proxy authentication and DHCP snooping on the same port/VLAN. The default access control list (ACL) for web-based proxy authentication has an ACE that allows DHCP snooping. The creation of DHCP snooping binding triggers web-based proxy authentication.
- Dynamic ARP inspection (DAI)—You can enable web-based proxy authentication and DAI on the same port/VLAN. The default ACL requires an ACE to allow ARP inspection. A host has static IP addresses configured. ARP inspection triggers web-based proxy authentication.
- IP source guard (IPSG)—You can enable web-based proxy authentication and IPSG on the same port. IPSG uses a PACL for access policy, and web-based proxy authentication uses a PBACL for access policy. The port ACL mode must be in merge mode in order for IPSG to work with web-based proxy authentication.
- 802.1X—Web-based proxy authentication and 802.1X are independent identity authentication protocols with 802.1X at Layer 2 and web-based proxy authentication at Layer 3. You can enable web-based proxy authentication with 802.1X. When you configure both web-based proxy authentication and 802.1X on a port, the port attempts to authenticate using 802.1X. After successful authentication, it receives policies from RADIUS. If a policy allows all web (HTTP/HTTPS) traffic, then web-based proxy authentication does not occur. The host is not authenticated if the 802.1X policies allow web traffic. If the 802.1X policies do not allow web traffic, then web-based proxy authentication occurs when the host sends the first HTTP/HTTPS packet that is not allowed by the policy. The packet is intercepted by the URL redirect ACE.

- MAC-Authentication Bypass—MAC-Authentication Bypass is a Layer 2 authentication that uses a MAC address. There is no actual authentication with MAC-Authentication Bypass. When you configure web-based proxy authentication on an interface that has MAC-Authentication Bypass configured, web-based proxy authentication occurs when the MAC-Authentication Bypass completes. MAC-Authentication Bypass adds the port to a VLAN and gets an IP address using DHCP, which triggers web-based proxy authentication.
- Port Security—When you enable port security and web-based proxy authentication on a port, the hosts that are secured by port security are web authenticated.
- Voice VLAN ID (VVID)—Web-based proxy authentication and VVID support is restricted to port-VLAN hosts.
- Guest VLAN—At the completion of the 802.1X authentication or MAC-Authentication Bypass, a port is added to the guest VLAN based on the 802.1X or the MAC-Authentication Bypass authentication result. The port receives an IP address using DHCP in the guest VLAN. Web-based proxy authentication occurs after the IP address is received.
- Auth-Fail-VLAN—You can enable web-based proxy authentication and the authentication-fail VLAN on the same port/VLAN.
- Network Admission Control (NAC)—You can enable web-based proxy authentication and NAC LAN port IP on the same port/VLAN. NAC with LAN port IP is independent of web-based proxy authentication; LAN port IP posture validation can happen before web-based proxy authentication.

Default Web-Based Proxy Authentication Configuration

Table 42-1 shows the default web-based proxy authentication configuration settings.

Table 42-1 Web-Based Proxy Authentication Default Configuration

Feature	Default Value
Port access entity (PAE) capability	Authenticator only
Web-based proxy authentication—Global	Disabled
Web-based proxy authentication—Per port	Disabled
Global session timeout	3600 seconds
Quiet timeout	60 seconds
Login attempts	3 attempts

Web-Based Authentication Guidelines and Restrictions

This section provides the guidelines and restrictions for configuring web-based proxy authentication:

- Web-based authentication is not supported on trunk or port-channel interfaces.
- Because PBACL will be mapped to a VLAN, all ports in the VLAN have default access specified by the PBACLs default policy. We recommend that you enable web-based authentication on all the ports in the VLAN.

- Before you enable web-based proxy authentication on a port, you must map a PBACL with the following ACEs to the VLAN:
  - DHCP snooping
  - ARP inspection
  - Allow DNS
  - Policy config
  - URL Redirect
  - Default policy
- Before you enable web-based proxy authentication on a port, you must enable ARP inspection for the static IP hosts and configure the static ARP inspection rules.

This example shows how to configure a typical ACL with these ACEs:

```
permit dhcp-snooping
permit arp-inspection <ip_addr> <hwaddr>
permit udp any eq dns any                    [permit DNS]
permit tcp any eq domain any                 [permit DNS w/TCP]
<Policy configuration>
permit ip group Exception ExpServers
permit ip group Engineer EngServers
permit ip group Manager MgrServers
permit ip group Admin any
permit url-redirect                          [permit URL redirection]
deny ip any any                              [Default policy]
```

When the host first comes up, there are no policies configured for the host IP and all host traffic, except for the HTTP traffic that is controlled by the default policy and configured in the PBACL. The HTTP traffic is redirected to the supervisor engine. Web-based proxy authentication registers this IP with URL redirection when it receives a trigger from DHCP or ARP. The URL redirection module on the supervisor engine receives the packet and passes it to web-based proxy authentication.

After successful authentication, web-based proxy authentication adds the host IP to the groups that are received from RADIUS, expands the PBACL, and updates the Ternary Content Addressable Memory (TCAM). The host traffic is controlled by the policy configuration. Because the HTTP redirection ACE is at the end, it will not be affected if the host policies are in place. Once the host policies are removed (after the session timeout has been exceeded), the host traffic is again subjected to the default policy and HTTP traffic gets redirected to the supervisor engine.

Configuring Web-Based Proxy Authentication

This section describes how to configure web-based proxy authentication:

Enabling or Disabling Web-Based Proxy Authentication Globally

You must enable web-based proxy authentication for the entire system before you can configure it for the individual ports. After you enable web-based proxy authentication globally, you can configure the individual ports for web-based proxy authentication. To enable web-based proxy authentication for the individual ports, see the "Enabling or Disabling Web-Based Proxy Authentication on a Port" section on page 42-10.

To enable or disable web-based authentication globally, perform these tasks in privileged mode:

Task	Command
Globally enable web-based proxy authentication.	**set web-auth enable**
Globally disable web-based proxy authentication.	**set web-auth disable**

This example shows how to enable web-based proxy authentication globally:

```
Console> (enable) set web-auth enable
enabled web-auth
Console> (enable)
```

This example shows how to disable web-based proxy authentication globally:

```
Console> (enable) set web-auth disable
disabled web-auth
Console> (enable)
```

Enabling or Disabling Web-Based Proxy Authentication on a Port

You can enable web-based proxy authentication for individual ports after you enable web-based proxy authentication globally. To enable web-based proxy authentication globally, see the "Enabling or Disabling Web-Based Proxy Authentication Globally" section on page 42-10.

Note If you have disabled web-based proxy authentication globally, web-based proxy authentication on a port may not start but will be stored in the configuration.

To enable or disable web-based authentication on a port, perform these tasks in privileged mode:

Task	Command
Enable web-based proxy authentication on a port.	**set port web-auth** *mod/port* **enable**
Disable web-based proxy authentication on a port.	**set port web-auth** *mod/port* **disable**

This example shows how to enable web-based proxy authentication on a port:

```
Console> (enable) set port web-auth 1/1 enable
web-authentication successfully enabled on Interface 1/1.
Console> (enable)
```

This example shows how to disable web-based proxy authentication on a port:

```
Console> (enable) set port web-auth 1/1 disable
web-authentication successfully disabled on Interface 1/1.
Console> (enable)
```

Initializing Web-Based Proxy Authentication on a Port

When you initialize the port with the **set port web-auth initialize** command, you are returning the port to the first state. In this state, the IP address of the host is registered with URL redirection for redirecting any HTTP packet from this host to the supervisor engine.

If you specify the *ip_addr* argument, web-based proxy authentication is initialized for that host only. If you do not specify the *ip_addr* argument, web-based proxy authentication is initialized for all hosts.

You must enable web-based proxy authentication globally and on the individual port before you can initialize a web-based proxy authentication port for authentication again.

To initialize a web-based proxy authentication port for authentication again, perform this task in privileged mode:

Task	Command
Initialize a web-based proxy authentication port for authentication again.	**set port web-auth** *mod/port* **initialize** [*ip_addr*]

This example shows how to initialize web-based proxy authentication again for all hosts on a port:

```
Console> (enable) set port web-auth 2/1 initialize
Initialized web-authentication for all hosts on port 2/1.
Console> (enable)
```

This example shows how to initialize web-based proxy authentication again for a specific host:

```
Console> (enable) set port web-auth 2/1 initialize 10.1.1.1
Initialized web-authentication for host 10.1.1.1 on port 2/1.
Console> (enable)
```

Configuring the Login Page URL

When you enter the URL, use the url = **http://***string*.

To configure the URL for the login page, perform this task in privileged mode:

Task	Command
Configure the URL for the login page.	**set web-auth login-page url** *url*

This example shows how to configure the URL for the login page:

```
Console> (enable) set web-auth login-page url http://proxyauth.cisco.com/login.html
web-auth login-page configured.
Console> (enable)
```

Configuring the Login-Fail Page URL

When you enter the URL, use this format, url = **http://***string*.

To configure the URL for the login-fail page, perform this task in privileged mode:

Task	Command
Configure the URL for the login-fail page.	**set web-auth login-fail-page url** *url*

This example shows how to configure the URL for the login-fail page:

```
Console> (enable) set web-auth login-fail-page url http://proxyauth.cisco.com/login.html
web-auth login fail page configured.
Console> (enable)
```

Specifying the Session Timeout Period

You can specify the amount of time that this session is valid. After the time has been exceeded, the web-authenticated session is terminated. The RADIUS-supplied session timeout takes precedence over the locally configured value.

To specify the timeout period for the global web-based proxy authentication sessions, perform this task in privileged mode:

Task	Command
Specify the timeout period for the global web-based proxy authentication sessions.	**set web-auth session-timeout** *seconds*

This example shows how to specify the timeout period for the global web-based proxy authentication sessions:

```
Console> (enable) set web-auth session-timeout 20
web-authentication session-timeout set to 20 seconds.
Console> (enable)
```

Specifying the Quiet Period

When the authenticator cannot authenticate the host, it remains idle for a set period of time and then tries again. The idle time is determined by the quiet-period value. The default is 60 seconds. You may set the *seconds* value from 0 to 65535 seconds.

To specify the duration of the quiet period, perform this task in privileged mode:

Task	Command
Specify the quiet period.	**set web-auth quiet-timeout** *seconds*

This example shows how to specify the quiet period:

```
Console> (enable) set web-auth quiet-timeout 20
web-authentication quiet-timeout set to 20 seconds.
Console> (enable)
```

Specifying the Maximum Login Attempts

You can specify the maximum number of unsuccessful login attempts allowed before blocking the user.

To specify the maximum number of login attempts, perform this task in privileged mode:

Task	Command
Specify the maximum number of login attempts.	**set web-auth login-attempts** *count*

This example shows how to specify the maximum number of login attempts:

```
Console> (enable) set web-auth login-attempts
web-authentication max retry count set to <count>
Console> (enable)
```

Displaying Web-Based Proxy Authentication Information

This section describes how you can display the following web-based proxy authentication information:

- Displaying Summary of Session Information, page 42-13
- Displaying Per-Port Information, page 42-14

Displaying Summary of Session Information

If you specify the **vlan** *vlan_id* keyword and argument, a summary of information for the specified VLAN is displayed.

In the command output display, the following applies:

- The * indicates the RADIUS assigned value.
- The State field displays the current web-authentication state for the given host.

To display a summary of information about the web-based proxy authentication session, perform this task in normal mode:

Task	Command
Display a summary of information for the web-based proxy authentication session.	**show web-auth summary** [**vlan** *vlan_id*]

This example shows how to display a summary of information about the web-based proxy authentication session:

```
Console> (enable) show web-auth summary
Web-authentication enabled globally
Login-page location url http://proxyauth.cisco.com/login.html
Login-fail-page location url http://proxyauth.cisco.com/loginfail.html
session-timeout : 3600 secs
quiet timeout : 60 secs
Max Login attempt count: 3
-------------------------------------------------------------------------------------------
--------------------------------
IP Address                  Interface       Web Auth State
     Session-Timeout   Leftover-Session-Time       VLAN
-------------------------------------------------------------------------------------------
-----------------------------------------------
9.9.150. 1                       1/1             Authenticated
              * 7200                           200                             100
9.9.150.2
                       1/2             Authenticating               3600
        -                              100
9.9.150.3                        1/3             Authentication-fai
         3600                           -                                100
9.9.160.10                      1/4             Held
               3600                            -                                200
9.9.170.15                      1/5
            Connecting                     3600                             -
               300
Console> (enable)
```

This example shows how to display a summary of information about the web-based proxy authentication session for a specific VLAN:

```
Console> (enable) show web-auth summary vlan 100
-------------------------------------------------------------------------------------------
--------------------------------
IP Address                  Interface       Web Auth State
     Session-Timeout   Leftover-Session-Time
-------------------------------------------------------------------------------------------
--------------------------------
9.9.150. 1                       1/1             Authenticated
             * 7200                           200
9.9.150.2                        1/2
Authenticating                3600                             -
9.9.150.3                        1/3             Held
                              3600                             -
Console> (enable)
```

Displaying Per-Port Information

The **show port web-auth** command displays the following information:

- IP address of the host.
- Current state.
- Session-timeout. The time displayed is the configured timeout if not supplied by RADIUS.
- Leftover session timeout value.

To display information about a web-based proxy authentication port, perform this task in normal mode:

Task	Command
Display information about a web-based proxy authentication port.	**show port web-auth** *mod/port*

This example shows how to display information about a web-based proxy authentication port:

```
Console> (enable) show port web-auth 3/48
Port IP-Address Vlan Web-Auth-State
----- --------------- ---- --------------
3/48 9.6.7.8 16 AUTHENTICATION_FAIL
Port IP-Address Session-Timeout Session-Timeleft Radius-Rcvd-Timeout
----- --------------- --------------- ---------------- -------------------
3/48 9.6.7.8 300 300 No
Port IP-Address Policy-Groups
----- --------------- -------------
3/48 9.6.7.8

Console> (enable)
```

Displaying Statistics

To display web-based proxy authentication statistics, perform this task in enable mode:

Task	Command
Display web-based proxy authentication statistics.	**show web-auth** *statistics*

This example shows how to display web-based proxy authentication statistics:

```
Console> (enable) show web-auth statistics
Total GET Requests received      : 0
Total POST Requests received     : 0
Total responses sent             : 0
Total web auth hosts             : 0
Total successful authentications : 0
Total failed authentications     : 0
Total critical active hosts      : 0
Total web auth Queue Entries     : 0
Total web auth Queue Drops       : 0
Console> (enable)
```

CHAPTER 43

Tracking Host Aging

This chapter describes how to configure IP device tracking with 802.1x, MAC authentication bypass, Web-proxy based authentication and EoU on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

Note For information on configuring IEEE 802.1X authentication, see Chapter 40, "Configuring 802.1X Authentication."

Note For information on configuring MAC Authentication Bypass, see Chapter 41, "Configuring MAC Authentication Bypass."

Note For information on configuring Web-Based Proxy Authentication, see Chapter 42, "Configuring Web-Based Proxy Authentication."

Note For information on configuring EoU, see Chapter 44, "Configuring Network Admission Control."

This chapter consists of the following sections:

Understanding How Host Aging is Tracked

Layer 2 authentication features, 802.1x, and MAC authentication bypass install entries into the CAM table to ensure packet switching in the hardware. The CAM entries are static and it cannot be ensured that they are current. The entries age with the hardware if they are not removed by the authentication feature at the end of the session. If a host leaves before the authentication session expires or if the authentication manager is not notified about removing the CAM entry, the stale entry remains in the hardware switching table. Even the Layer 3 protocols, LAN port IP and Web-based proxy authentication have no method to age out the CAM entry if the host leaves before the session expires.

The IP device-tracking feature, which is included in the authentication manager, tracks the existence of the host and removes aged entries in the CAM table. The device-tracking feature ensures that the hardware entries and authentication sessions get aged out. As a result of aging, the hosts are removed from the EARL.

Configuring IP Device Tracking Globally

When enabled, the IP device tracking feature sends out a probe to check if the host is still present. The probe can be sent out at regular intervals for a specified number of times. The default is enabled.

To enable or disable IP device tracking globally, perform this task in privileged mode:

Task	Command
Enable or disable IP device tracking globally.	**set ip device-tracking** {**disable** | **enable**}

This example shows how to enable IP device tracking globally:

```
Console> (enable) set ip device-tracking enable
Successfully enabled device tracking.
Console> (enable)
```

This example shows how to display the current global configuration of IP device tracking:

```
Console> (enable) show ip device-tracking
Device tracking mode    : Enabled
Device tracking count   : 3
Device tracking timeout : 30
Console> (enable)
```

The following sections describe how to set the probe interval and probe count values:

- Specifying the IP Device Tracking Interval, page 43-2
- Specifying the IP Device Tracking Count, page 43-3

Note The **probe interval** and **probe count** values can only be set globally and are common for all types of authentication methods.

Specifying the IP Device Tracking Interval

You can set IP device tracking to send a probe at regular intervals (in seconds). The range is from 5 to 65535 seconds. The default is 30 seconds.

To specify the probe interval, perform this task in privileged mode:

Task	Command
Specify the time period in seconds to send a probe.	**set ip device-tracking probe interval** *interval*

This example shows how to set the IP device tracking interval:

```
Console> (enable) set ip device-tracking probe interval 45
Device tracking probe interval set to 45 secs.
Console> (enable)
```

Specifying the IP Device Tracking Count

You can configure IP device tracking to send 1 to 10 probes after the host becomes idle. The default is 3 probes.

To set the probe count, perform this task in privileged mode:

Task	Command
Specify the number of times to check for the existence of a host.	**set ip device-tracking probe count** *count*

This example shows how to set the IP device tracking probe count:

```
Console> (enable) set ip device-tracking probe count 5
Device tracking probe count set to 5.
Console> (enable)
```

Configuring IP Device Tracking on a Port

The following topics describe how to configure IP device tracking on a port:

Enabling or Disabling IP Device Tracking on a Port with 802.1x Authentication

To enable or disable IP device tracking on a module or port with 802.1x authentication, perform this task in privileged mode:

Task	Command
Enable or disable IP device tracking on a module or port with 802.1x authentication. The defualt is disabled.	**set port dot1x** *mod/port* **ip-device-tracking** {**disable** | **enable**}

This example shows how to enable IP device tracking on a port with 802.1x authentication:

```
Console> (enable) set port dot1x 3/1 ip-device-tracking enable
Port 3/1 ip-device-tracking option is enabled.
Console> (enable)
```

This example shows how to view the current conifguration of IP device tracking on a port with 802.1x authentication:

```
Console> (enable) show port dot1x 3/13
Port  Auth-State          BEnd-State Port-Control        Port-Status
----- ------------------- ---------- ------------------- -------------
 3/13 authenticated       idle       auto                authorized

Port  Port-Mode     Re-authentication   Shutdown-timeout   Control-Mode
                                                           admin   oper
----- ------------- -----------------   ----------------   ---------------
 3/13 SingleAuth    enabled             disabled           Both    Both

Port  Posture-Token Critical-Status Termination action Session-timeout
----- ------------- --------------- ------------------ ---------------
 3/13 Healthy       no              Initialize         3600

Port  Session-Timeout-Override Url-Redirect
----- ----------------------- ----------------------------------------
 3/13 disabled                -

Port  Critical ReAuth-When IP-Device-Tracking
----- -------- ----------- ------------------
 3/13 disabled 105         enabled
Console> (enable)
```

Enabling or Disabling IP Device Tracking on a Port with MAC Authentication Bypass

To enable or disable IP device tracking on a module or port with MAC authentication bypass, perform this task in privileged mode:

Task	Command
Enable or disable IP device tracking on a module or port with MAC authentication bypass. The default is disabled.	**set port mac-auth-bypass** *mod/port* **ip-device-tracking** {**disable** | **enable**}

This example shows how to enable IP device tracking on a port with MAC authentication bypass:

```
Console> (enable) set port mac-auth-bypass 3/1 ip-device-tracking enable
Port 3/1 ip-device-tracking option is enabled.
Console> (enable)
```

This example shows how to view the current configuration of IP device tracking on a port with MAC authentication bypass:

```
Console> (enable) show port mac-auth-bypass 3/1
Port  Mac-Auth-Bypass State MAC Address       Auth-State         Vlan
----- -------------------- ----------------- -----------------  -----
 3/1 Enabled               00-00-00-00-00-00 waiting            1

Port  Termination action Session Timeout Shutdown/Time-Left
----- ------------------ --------------- ------------------
 3/1 initialize         300             NO       -

Port  PolicyGroups
----- ------------------------------------------------------------------------
 3/1 -

Port  Security ACL                     Sec ACL Type      QoS ACL Type
----- -------------------------------- ----------------- ----------------
 3/1 -                                -                 -

Port  QoS Ingress ACL                  QoS Egress ACL
----- -------------------------------- ----------------------------------
 3/1 -                                -

Port  Critical Critical-Status Ip-Device-Tracking
----- -------- --------------- ------------------
 3/1   Disabled -          Enabled

Port  Session-ID
----- --------------------------------
 3/1 -

Port  Posture Token URL-Redirect
----- ------------- --------------------------------
 3/1               -
```

Enabling or Disabling IP Device Tracking on a Port with Web-Based Proxy Authentication

To enable or disable IP device tracking on a port with web-based proxy authentication, perform this task in privileged mode:

Task	Command
Enable or disable IP device tracking on a module or port with web-based proxy authentication. The default is enabled.	**set port web-auth** *mod/port* **ip-device-tracking** {**disable** | **enable**}

This example shows how to enable IP device tracking on a port with web-based proxy authentication:

```
Console> (enable) set port web-auth 3/1 ip-device-tracking enable
```

```
Port 3/1 ip-device-tracking option is enabled.
Console> (enable)
```

This example shows how to view the current configuration of IP device tracking on a port with web-based proxy authentication:

```
Console> (enable) show port web-auth 3/1
Port  IP-Address      Vlan Enabled   Web-Auth-State        Critical-Status
----- --------------- ---- --------- -----------------     ---------------
 3/1  -               1    enabled   -                     -

Port  IP-Address      Session-Timeout Session-Timeleft Radius-Rcvd-Timeout
----- --------------- --------------- ---------------- --------------------
 3/1  -               -               -                No

Port  IP-Address      Policy-Groups
----- --------------- -------------
 3/1  -               -
Port  IP-Address      Ip-Device-Tracking
----- --------------- ------------------
 3/1  -               Enabled
```

Enabling or Disabling IP Device Tracking on a Port with EoU

To enable or disable IP device tracking on a port with EoU, perform this task in privileged mode:

Task	Command
Enable or disable IP device tracking on a module or port with EoU. The default is enabled.	**set port eou** *mod/port* **ip-device-tracking** {**disable** | **enable**}

This example shows how to enable IP device tracking on a port with EoU:

```
Console> (enable) set port eou 3/1 ip-device-tracking enable
Port 3/1 ip-device-tracking option is enabled.
Console> (enable)
```

This example shows how to view the current configuration of IP device tracking on a port with EoU:

```
Console> (enable) show port eou 3/1
Port     EOU-State IP Address      MAC Address       Critical-Status
-------- --------- --------------- ----------------- ---------------
 3/1    auto      -               -                 -
Port     FSM State     Auth Type   SQ-Timeout Session Timeout
-------- ------------- ----------- ---------- ---------------
 3/1     -              -          -          -
Port     Posture      URL Redirect
-------- ------------ --------------------
 3/1     -            -
Port     Termination action Session id
-------- ------------------ --------------------------------
 3/1     -                  -
Port     PolicyGroups
-------- ------------------------------------------------------
 3/1     -
Port  Critical Ip-Device-Tracking
----- -------- ------------------
 3/1 disabled enabled
```

CHAPTER 44

Configuring Network Admission Control

This chapter describes how to configure network admission control (NAC) on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

Note For information on configuring IEEE 802.1X authentication, see Chapter 40, "Configuring 802.1X Authentication."

Note For information on configuring MAC authentication bypass, see Chapter 41, "Configuring MAC Authentication Bypass."

Note For information on using port security to block input to an Ethernet, Fast Ethernet, or Gigabit Ethernet port when the MAC address of the station attempting to access the port is different from any of the MAC addresses that are specified for that port, see Chapter 38, "Configuring Port Security." That chapter also provides information on using port security to filter the traffic that is destined to or received from a specific host that is based on the host MAC address.

Note For information on configuring authentication, authorization, and accounting (AAA) to monitor and control access to the command-line interface (CLI) on the Catalyst 6500 series switches, see Chapter 39, "Configuring the Switch Access Using AAA."

This chapter consists of these sections:

Configuring Network Admission Control with LAN Port IP

These sections describe how to configure NAC with LAN port IP:

Understanding How Network Admission Control with LAN Port IP Works

These sections provide an understanding of LAN port IP:

Overview

NAC addresses the increased threat and impact of worms and viruses to networked businesses. This feature is part of the Cisco Self-Defending Network Initiative that helps customers identify, prevent, and adapt to security threats.

In its initial phase, NAC enables switches and routers to restrict access privileges from an end point that is attempting to connect to a network. The access can be based on information about the end-point device, such as its current antivirus state (version of antivirus software, virus definitions, and version of scan engine).

NAC systems allow noncompliant devices to be denied access, placed in a quarantined area, or given restricted access to computing resources, which keeps insecure nodes from infecting the network.

The key component of the Cisco NAC program is the Cisco Trust Agent (CTA), which resides on an end-point system and communicates with Cisco switches and routers on the network. The CTA collects security state information, such as the type of antivirus software that is used, and communicates this information to Cisco switches and routers. The information is then relayed to a Cisco Secure Access Control Server (ACS) where access control decisions are made. The ACS directs the Cisco switch or router to perform enforcement against the end point.

Virus Infections and Their Effect on Networks

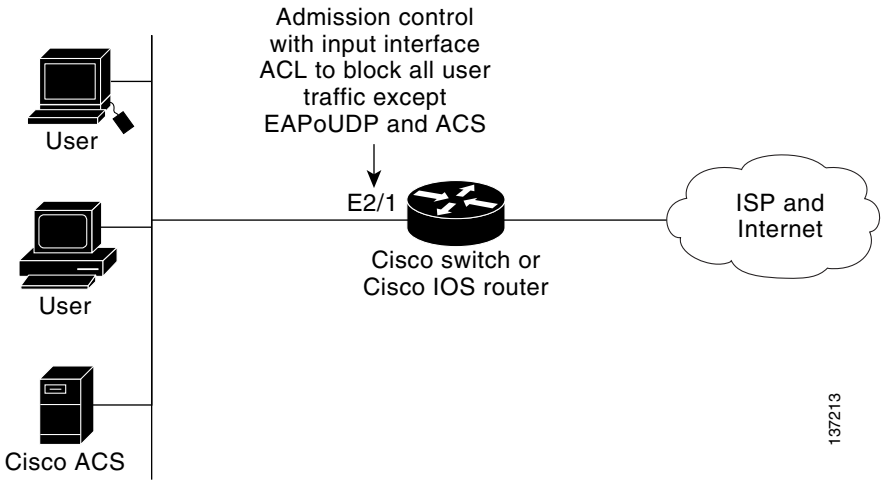
Virus infections are the single largest cause of serious security breaches for networks. Sources of virus infections are insecure end points (for example, PCs, laptops, and servers). Although the end points may have antivirus software installed, the software is often disabled. Even if the software is enabled, the end points may not have the latest virus definitions and scan engines. A larger security risk is from devices that do not have any antivirus software installed.

How Network Admission Control Works

End-point systems, or clients, are hosts on the network, such as PCs, laptops, workstations, and servers. The end-point systems are a potential source of virus infections, and their antivirus states need to be validated before they are granted network access. When an end point attempts an IP connection to a network through an upstream Cisco network access device (Cisco switch or router), the network access device challenges the end point for its antivirus state. The end-point systems run a client called Cisco Trust Agent, which collects antivirus state information from the end device and transports the information to the network access device. This information is then communicated to a Cisco Secure ACS where the antivirus state of the end point is validated and access control decisions are made and returned to network access devices. The network devices either permit, deny, or quarantine the end device. The Cisco Secure ACS may use back-end antivirus vendor-specific servers for evaluating the antivirus state of the end point.

Figure 44-1 shows how Cisco NAC works.

Figure 44-1 Cisco IOS Network Admission Control System



Network Access Device

A network access device (NAD) is a Cisco switch or router (a Layer 3 Extensible Authentication Protocol over UDP [EAPoUDP] access point) that provides connectivity to external networks, such as the Internet or remote enterprise networks.

Cisco Trust Agent

CTA is a specialized software that runs on end-point systems. CTA responds to challenges from the switch or router about the antivirus state of an end-point system. If an end-point system is not running the CTA, the network access device (switch or router) classifies the end-point system as "clientless."

Cisco Secure ACS

Cisco Secure ACS provides authentication, authorization, and accounting services for NAC using RADIUS authentication. Cisco Secure ACS returns access control decisions to the network access device on the basis of the antivirus credentials of the end-point system.

Using RADIUS cisco_av_pair vendor-specific attributes (VSAs), you can set the following attribute-value pairs (AV pairs) on the Cisco Secure ACS. These AV pairs are sent to the network access device with other access-control attributes:

- url-redirect—Enables the AAA client to intercept an HTTP request and redirect it to a new URL. This redirection is useful if the result of posture validation indicates that the network access control end point requires an update or patch that you have made available on a remediation web server. For example, a user can be redirected to a remediation web server to download and apply a new virus Directory Administration Tool (DAT) file or an operating system patch as follows:

  ```
  url-redirect=http://10.1.1.1
  ```

  URL-redirect for audit support—The audit function is for hosts that do not have Cisco CTA enabled. The audit can be triggered by the ACS by sending down a policy required for audit when there is a clientless authentication done by the network access device (NAD). The audit is accomplished by sending down the audit server's URL as the URL-redirect policy for the host. When HTTP traffic is seen from the host, it is given the URL of the audit server. The policy that is configured through policy-based ACLs (PBACLs) allows communication between the audit server and the host. The session timeout is typically small for the audit to complete and when this timeout expires, a revalidation occurs and the NAD sends the previously received state attribute to the ACS to bring down a new policy. If the audit is not finished during this session timeout, the ACS sends another short session timeout and this process continues until an audit posture token is received. If the process never completes or is taking too long, the audit server returns an "error" posture token to the ACS.

- posture-token—Enables Cisco Secure ACS to send a text version of a system posture token (SPT) that is derived by posture validation. The SPT is always sent in numeric format. Using the posture-token AV pair makes it easier to view the result of a posture validation request on the AAA client as follows:

  ```
  posture-token=Healthy
  ```

  Valid SPTs, in order from best to worst, are as follows:

  - Healthy
  - Checkup
  - Quarantine
  - Infected
  - Unknown

  Posture validation, or posture assessment, refers to the act of applying a set of rules to posture data to provide an assessment of the level of trust that you can place in an endpoint. The term posture is used to refer to the collection of attributes that play a role in the conduct and health of the endpoint

device that is seeking access to the network. Some of these attributes relate to the endpoint device-type and operating system; other attributes belong to various security applications that might be present on the endpoint, such as antivirus (AV) scanning software. The posture token is one of the conditions in the authorization rules for network access. Posture validation, together with traditional user authentication, provides a complete security assessment of the endpoint device and the user.

- status-query-timeout—Overrides the status-query default value of the AAA client with the value that you specify, in seconds, as follows:

```
status-query-timeout=150
```

For more information about AV pairs that are supported by Cisco software, see the documentation for the releases of software that are implemented on your AAA clients.

Redirection

NAC supports HTTP redirection that redirects any HTTP request from the end-point device to a specified redirect address. This support mechanism redirects all HTTP requests from a source to a specified web page (URL) to which the latest antivirus files can be downloaded. You must set the value of the url-redirect VSA on the ACS and, correspondingly, associate an access control entry in the downloadable ACL that permits the access of the end-point system to the redirect URL address.

LAN Port IP Posture Validation Summary

LAN port IP allows posture-validating end-user devices to access the network based on their posture. End-user devices are classified into one of five possible states after posture validation: healthy, checkup, quarantine, infected, or unknown. Network access is given depending on the device's posture.

LAN port IP enforcement mechanisms include URL redirection and auditing. PBACLs are used for enforcing network access.

The basic steps in posture validation are as follows:

1. The NAD learns the MAC and IP address bindings using ARP inspection and/or DHCP snooping.

   Note If you use DHCP triggering for posture validation, you must also enable ARP inspection. If ARP inspection is not enabled, the posture validation completes but the session is torn down within a few minutes because the ARP probe replies from the client are not seen by the EAP Over UDP (EOU) state machinery.

2. The NAD sends an EOU hello request to the host.
3. If the host is running CTA, it responds back with a hello response.
4. The NAD sends an EOU validate identity request.
5. The CTA responds back with an EOU validate response.
6. The NAD extracts the EAP packet from the EOU, embeds it in the RADIUS access request, and sends it to the authentication server (such as the ACS).
7. The ACS sends back an access challenge that is relayed back to the CTA in the form of an EOU validate packet.
8. Step 6 and Step 7 continue until the ACS sends a success or failure response for the posture validation session.

9. If it is a success, the ACS sends the posture token VSA and a policy associated with the posture that includes the PBACL groups, session timeout, status query timeout, and authenticated username.

If the host does not respond to the EOU hello requests that are sent by the NAD, the NAD (after a preconfigured number of attempts), declares the host as clientless (no CTA). The NAD does a pseudo authentication on behalf of the host and brings down a policy. Other posture validation mechanisms, such as an audit, may be triggered.

In the clientless mode, the NAD sends three EOU hello messages (by default) before declaring that the host does not have a CTA. This process could take 90 seconds for doing a clientless authentication and installing that policy. To avoid this delay on a port that you know does not have a CTA, you can set the port mode to bypass using the per-port CLI (enter the **set port eou** *mod/port* **bypass** command). When this action is done, the port immediately does a clientless authentication when it learns a new IP address.

Exceptions are hosts that should not attempt posture validation because they are not capable. When a host that has been specified as an exception is detected, a preconfigured policy is installed.

LAN Port IP Hardware and Software Requirements

Follow these hardware and software requirements when configuring LAN port IP:

- You must have a Catalyst 6500 series switch running software release 8.5(1) or later releases.
- You must have CTA installed on the end-point devices (for example, on PCs and laptops).
- You must have an ACS for AAA.

LAN Port IP Configuration Guidelines and Restrictions

Follow these configuration guidelines and restrictions when configuring LAN port IP:

- You must be familiar with configuring access control lists (ACLs) and policy-based ACLs (PBACLs).
- You should be familiar with configuring authentication, authorization, and accounting (AAA).
- LAN port IP works with other security features such as 802.1X, MAC authentication bypass, and web-based proxy authentication. The restrictions that apply to 802.1X ports also apply to LAN port IP ports as follows:
  - LAN port IP can be configured on access ports only; it cannot be configured on trunk ports.
  - LAN port IP ports cannot be part of an EtherChannel.
  - LAN port IP cannot be enabled with dynamic ports.
  - LAN port IP can be enabled on Ethernet ports only.
  - LAN port IP ports cannot be SPAN destination ports.
  - LAN port IP ports cannot be part of a private VLAN.

Note With software release 8.6(1) and later releases, LAN port IP ports can be part of a private VLAN. For more information, see the "Configuring LAN Port IP on Private VLAN Ports" section on page 44-34.

- LAN port IP, when enabled with any authentication feature such as 802.1X or MAC authentication bypass, is initialized only after the authentication is finished.

- 802.1X—802.1X authentication may apply a Layer 2 policy, such as a VLAN assignment, and can also bring Layer 3 policy attributes, such as policy-based ACLs (PBACLs), to a port. A LAN port IP policy consists only of the policy-group membership that is downloaded from the RADIUS server.
- Multihost and multiauthentication modes are not supported—802.1X with LAN port IP is supported only in single-host mode.
- Auxiliary VLANs—LAN port IP is supported on multi-VLAN access ports.
- Guest VLANs and the authentication failure VLAN—When LAN port IP is configured with these two features, the LAN port IP operation differs only in that the IP address that it gets for posture validation is from the guest VLAN or authentication failure VLAN.
- DHCP snooping and/or ARP inspection—IP learning is through ARP inspection or DHCP snooping. You must enable at least one of these features for LAN port IP to work. These features are required to trigger LAN port IP (you must map a PBACL containing the ACEs of these features to the VLAN that the LAN port IP port resides in). If you do not enable one of these features, a Layer 2 switch cannot learn new IP addresses that appear on a port.

Note If you use DHCP triggering for posture validation, you must also enable ARP inspection. If ARP inspection is not enabled, the posture validation completes but the session is torn down within a few minutes because the ARP probe replies from the client are not seen by the EOU state machinery.

Note Supervisor Engine 1 does not support ARP inspection. With a Supervisor Engine 1, you must enable DHCP snooping.

- Port security—LAN port IP works with port security. Only port security-validated MAC addresses are allowed to go through posture validation. If a port security violation occurs and results in a port shutdown, the LAN port IP state of the port is also cleared. When you configure an authentication feature, the authenticating feature gives the MAC address to port security to secure if it has been successfully authenticated and then LAN port IP is initialized.
- Security ACLs (VACLs)—Security ACLs are used as PBACLs and PBACLs are supported in VACL mode only with LAN port IP.
- MAC authentication bypass—LAN port IP is initialized only after a successful authentication using MAC authentication bypass, 802.1X, or web-based proxy authentication.
- Web-based proxy authentication—LAN port IP is initialized only after web-based proxy authentication completes verifying identity credentials. In the web-based proxy authentication state, a port waits indefinitely for authentication to complete. In this stage, only DHCP and DNS are allowed to go through. The ACL configured on the interface handles the redirecting of HTTP traffic. The PBACL configured on the interface should ensure that any other traffic is not allowed.

Configuring LAN Port IP

This section describes how to configure LAN port IP.

Note To display LAN port IP configuration information and to clear LAN port IP configuration elements, see the "LAN Port IP CLI Command Examples" section on page 44-9. To configure policy-based ACLs (PBACLs), see the "Configuring Policy-Based ACLs" section on page 44-21.

Note For assistance in following these configuration steps, see the "LAN Port IP Configuration Example" section on page 44-30.

To configure LAN port IP, perform these steps:

Step 1 Enable LAN port IP globally on the switch by entering the **set eou** {**enable** | **disable**} command (the default is disabled).

```
Console> (enable) set eou enable
EoU globally enabled.
Console> (enable)
```

Step 2 Enable LAN port IP on a per-port basis by entering the **set port eou** *mod/port* {**bypass** | **auto** | **disable** | **initialize** | **revalidate**} command.

```
Console> (enable) set port eou 7/1 auto
EoU enabled on 7/1
Console> (enable)
```

Step 3 Define the RADIUS server and RADIUS key by entering the following commands:

set radius server *ip_addr* [**auth-port** *port*] [**acct-port** *port*] [**primary**]

set radius key *key*

This example shows how to define the RADIUS server:

```
Console> (enable) set radius server 10.76.39.93 auth-port 1812 primary
10.76.39.93 with auth-port 1812 acct-port 1813 added to radius server table as primary
server.
Console> (enable)
```

This example shows how to define the RADIUS key:

```
Console> (enable) set radius key cisco
Radius key set to cisco
Console> (enable)
```

Step 4 Define a policy-based ACL (PBACL) and map it to a VLAN as follows:

a. Enable DHCP snooping and/or ARP inspection:

set security acl ip *acl-name* **permit dhcp-snooping**

set security acl ip *acl-name* **permit arp-inspection**

b. Enable EAPoUDP redirection:

set security acl ip *acl-name* **permit eapoudp**

c. Define other policy statements using policy groups that correspond to various LAN port IP states as follows:

set security acl ip NACACL permit ip group healthy_hosts any

set security acl ip NACACL deny ip group infected_hosts any

set security acl ip NACACL permit ip group exception_hosts any

set security acl ip NACACL permit ip group clientless_hosts host 10.76.39.100

d. For URL redirection, apply this ACE at an appropriate position:

set security acl ip NACACL permit url-redirect

Step 5 For clientless nonresponsive hosts (NRH hosts), enable the clientless functionality by entering the **set eou allow clientless enable** command.

Step 6 Define a policy for NRH hosts. The specified groups should also be present in the ACL that is defined in the previous steps:

set policy name *exception_policy* **group** *exception_hosts*

Step 7 Specify an exception host and assign the policy by entering the **set eou authorize ip 77.0.0.90 policy exception_policy** command.

Step 8 Configure the RADIUS server. For RADIUS server configuration details, refer to the *Implementing Network Admission Control Phase One Configuration and Deployment* publication at this URL:

http://www.cisco.com/application/pdf/en/us/guest/netsol/ns466/c654/cdccont_0900aecd80217e26.pdf

Ensure that the policy groups that are used in the ACLs are configured with the posture-token VSA, such as 26/9/1 sec:pg=healthy_hosts.

If you define a policy group in ACS but the VACL that is mapped to the VLAN does not refer to that group, posture validation will fail because the policy installation fails.

Step 9 Ensure that the sc0 interface is configured with a proper IP address by entering these commands:

set interface {**sc0** | **sl0** | **sc1**} {**up** | **down**}

set interface sc0 [**vlan**] [*ip_addr*/*netmask* [*broadcast*]]

Step 10 Ensure that there is a default router in the VLAN to which the host is connected. If there is no default router, you need a static ARP on the host for the sc0 IP address.

Step 11 If the host and the management interface (sc0) are in the same VLAN, and you have a VACL configured for that VLAN, you should configure an ACE to allow traffic to the RADIUS server from the switch IP address.

LAN Port IP CLI Command Examples

This section describes how to configure the LAN port IP CLI:

Enabling or Disabling LAN Port IP Globally

To globally enable or disable LAN port IP on the switch, perform this task in privileged mode (the default is disabled):

Task	Command
Globally enable or disable LAN port IP on the switch.	**set eou** {**enable** | **disable**}

This example shows how to globally enable LAN port IP on the switch:

```
Console> (enable) set eou enable
EoU globally enabled.
Console> (enable)
```

Enabling or Disabling the Bypassing of LAN Port IP Posture Validation for Clientless Hosts

To globally enable or disable the bypassing of the LAN port IP posture validation for clientless hosts, perform this task in privileged mode (the default is disable):

Task	Command
Enable or disable the bypassing of the LAN port IP posture validation for clientless hosts.	**set eou allow clientless** {**enable** | **disable**}

This example shows how to enable the bypassing of the LAN port IP posture validation for clientless hosts:

```
Console> (enable) set eou allow clientless enable
EoU Clientless hosts will be allowed
Console> (enable)
```

Statically Authorizing an IP Address as an Exception Host Device and Applying a Policy to the Device

This command allows a specific IP address to be treated as an exception host and when that host is detected, it will dynamically install the policy specified by the policy name.

Note If the policy template does not exist, entering these commands creates the policy template.

To statically authorize an IP device and apply an associated policy to the device, perform this task in privileged mode:

Task	Command
Statically authorize an IP device and apply an associated policy to the device.	**set eou authorize ip** *ip_addr* **policy** *policy_name* **set eou authorize ip** *ip_addr ip_mask* **policy** *policy_name*

This example shows how to statically authorize an IP device and apply an associated policy to the device:

```
Console> (enable) set eou authorize ip 172.20.52.19 255.255.255.224 policy poll
Mapped IP address 172.20.52.0 IP mask 255.255.255.224 to policy name poll
Console> (enable)
```

Statically Authorizing a MAC Address as an Exception Host Device and Applying a Policy to the Device

This command allows a specific MAC address to be treated as an exception host and when that host is detected, it will dynamically install the policy specified by the policy name.

Note If the policy template does not exist, entering these commands creates the template.

To statically authorize a device using the device MAC address and apply an associated policy to the device, perform this task in privileged mode:

Task	Command
Statically authorize a device using the device MAC address and apply an associated policy to the device.	**set eou authorize mac-address** *mac_address* **policy** *policy_name* **set eou authorize mac-address** *mac_address mac_mask* **policy** *policy_name*

This example shows how to statically authorize a device using the device MAC address and apply an associated policy to the device:

```
Console> (enable) set eou authorize mac-address 03-56-B7-45-65-56 policy poll
Mapped MAC 03-56-b7-45-65-56 to policy name poll.
Console> (enable)
```

Restarting a Host's State Machine

To restart a host's state machine, perform this task in privileged mode:

Task	Command
Restart a host's state machine.	**set eou initialize all** **set eou initialize authentication** {**clientless** | **eap** | **static**} **set eou initialize ip** *ip-address* **set eou initialize mac** *mac-address* **set eou initialize posture-token** *posture-token*

This example shows how to restart a host's state machine using the IP address:

```
Console> (enable) set eou initialize ip 172.20.52.19
Initializing Eou for ipAddress 172.20.52.19
Console> (enable)
```

Specifying the CTA Packet Retransmit Time and RADIUS Server Retransmit Time

To specify the number of times that a packet is retransmitted to the CTA before declaring the CTA as nonresponsive, and to specify the RADIUS server retransmit time, perform this task in privileged mode (the default is 3 and the range is 1 through 10):

Task	Command
Specify the number of times that a packet is retransmitted to the CTA before declaring the CTA as nonresponsive, and specify the RADIUS server retransmit time.	**set eou max-retry** *max-retry*

This example shows how to specify the number of times that a packet is retransmitted to the CTA before declaring the CTA as nonresponsive, and specify the RADIUS server retransmit time:

```
Console> (enable) set eou max-retry 6
eou max-retry set to 6.
Console> (enable)
```

Revalidating a Host

To revalidate a host, perform this task in privileged mode:

Task	Command
Revalidate a host.	**set eou revalidate all** **set eou revalidate authentication** {**clientless** | **eap** | **static**} **set eou revalidate ip** *ip-address* **set eou revalidate mac** *mac-address* **set eou revalidate posture-token** *posture-token*

This example shows how to revalidate all clientless hosts:

```
Console> (enable) set eou revalidate authentication clientless
Revalidate all clientless hosts
Console> (enable)
```

Enabling or Disabling EOU Logging for LAN Port IP Events

To enable or disable EOU logging for LAN port IP events, perform this task in privileged mode (the default is disable):

Task	Command
Enable or disable EOU logging for LAN port IP events.	**set eou logging** {**enable** | **disable**}

This example shows how to enable EOU logging for LAN port IP events:

```
Console> (enable) set eou logging enable
EoU Logging enabled
Console> (enable)
```

Setting EAPOUDP-Related Timers

To set EAPOUDP-related timers, perform this task in privileged mode:

Task	Command
Set EAPOUDP-related timers.	**set eou timeout aaa** *aaa-timeout*
	set eou timeout hold-period *hold-timeout*
	set eou timeout retransmit *retransmit-timeout*
	set eou timeout revalidation *revalidation-timeout*
	set eou timeout status-query *status-query-timeout*

The timer defaults and ranges are as follows:

- aaa—The default is 60 seconds; the range is 1 through 60 seconds.
- hold-period—The default is 180 seconds; the range is 60 through 86400 seconds.
- retransmit—The default is 3 seconds; the range is 1 through 60 seconds.
- revalidation—The default is 36000 seconds; the range is 5 through 86400 seconds.
- status-query—The default is 300 seconds; the range is 30 through 1800 seconds.

This example shows how to set the revalidation timer to 200 seconds:

```
Console> (enable) set eou timeout revalidation 200
Console> (enable)
```

Setting EOU Rate Limiting

To set EOU rate limiting (the default is 0 and the range is 10 through 200), perform this task in privileged mode:

Note The default rate limit value of 0 disables rate limiting. With rate limiting disabled, there is no limit on simultaneous LAN port IP authentication sessions.

Task	Command
Set EOU rate limiting.	**set eou rate-limit** *ratelimit*

This example shows how to set EOU rate limiting to 40:

```
Console> (enable) set eou rate-limit 40
eou ratelimit set to 40.
Console> (enable)
```

Enabling or Disabling EOU RADIUS Accounting

To enable or disable EOU RADIUS accounting, perform this task in privileged mode:

Task	Command
Enable or disable EOU RADIUS accounting.	**set eou radius-accounting** {**enable** | **disable**}

This example shows how to enable EOU RADIUS accounting:

```
Console> (enable) set eou radius-accounting enable
Radius Accounting for Eou Enabled.
Console> (enable)
```

Bypassing, Disabling, or Enabling LAN Port IP on a Per-Port Basis

You can bypass, disable, or enable LAN port IP on a per-port basis. Specifying **auto** mode enables LAN port IP automatically if a client is found.

To bypass, disable, specify auto mode, or set the aaa-fail policy for LAN port IP on a per-port basis, perform this task in privileged mode:

Task	Command
Bypass, disable, specify auto mode, or set the aaa-fail policy for LAN port IP on a per-port basis.	**set port eou** *mod*/*port* {**aaa-fail-policy** | **auto** | **bypass** | **disable** | **initialize** | **revalidate**}

This example shows how to enable an aaa-fail policy on a port:

```
Console> (enable) set port eou 1/2 aaa-fail-policy test_policy
Policy test_policy mapped as aaa-fail-policy on port 1/2
Console> (enable)
```

This example shows how to enable LAN port IP on port 5/1:

```
Console> (enable) set port eou 5/1 auto
EoU enabled on 5/1
Console> (enable)
```

This example shows how to set port 7/1 to bypass mode:

```
Console> (enable) set port eou 7/1 bypass

Eou Bypass enabled on 7/1
Console> (enable)
```

Initializing LAN Port IP on a Per-Port Basis

To initialize LAN port IP on a per-port basis, perform this task in privileged mode:

Task	Command
Initialize LAN port IP on a per-port basis.	**set port eou** *mod*/*port* **initialize**

This example shows how to initialize LAN port IP on port 7/1:

```
Console> (enable) set port eou 7/1 initialize
Initializing EoU for all hosts on port 7/1
Console> (enable)
```

Revalidating LAN Port IP on a Per-Port Basis

To revalidate LAN port IP on a per-port basis, perform this task in privileged mode:

Task	Command
Revalidate LAN port IP on a per-port basis.	**set port eou** *mod/port* **revalidate**

This example shows how to revalidate LAN port IP on port 7/1:

```
Console> (enable) set port eou 7/1 revalidate
Re-validating EoU for all hosts on port 7/1
Console> (enable)
```

Redirecting LAN Port IP Control Packets to the Supervisor Engine

To redirect all LAN port IP control packets to the supervisor engine (EAP over UDP packets), perform this task in privileged mode:

Task	Command
Redirect all LAN port IP control packets to the supervisor engine (EAP over UDP packets).	**set security acl ip** *acl_name* **permit eapoudp** *ip_mask* [**before** | **modify**] *ace_insert_position*

This example shows how to redirect all LAN port IP control packets to the supervisor engine (EAP over UDP packets):

```
Console> (enable) set security acl ip test permit eapoudp mask1 before pos1
Successfully configured EAPoUDP ACL test. Use 'commit' command to save changes
```

Displaying the Global EOU Configuration

To display the global EOU configuration, perform this task in normal mode:

Task	Command
Display the global EOU configuration.	**show eou config**

This example shows how to display the global EOU configuration:

```
Console> (enable) show eou config
Eou Protocol Version : 1
Eou Global Config
-----------------
Eou Global Enable         : Enabled
Eou Clientless            : Disabled
Eou Logging               : Enabled
```

```
Eou Radius Accounting     : Enabled
Eou MaxRetry              : 3
Eou AAA timeout           : 60
Eou Hold timeout          : 180
Eou Retransmit timeout    : 30
Eou Revalidation timeout  : 3600
Eou Status Query timeout  : 300
Eou Rate Limit            : 40
Eou Udp Port              : 21862

Ip Exception List and Policies
--------------------------------------
0.0.0.18         255.255.255.224  TEST

Console> (enable)
```

Displaying a Summary of the LAN Port IP State on All LAN Port IP-Enabled Ports

To display a summary of the LAN port IP state on all LAN port IP-enabled ports, perform this task in normal mode:

Task	Command
Display a summary of the LAN port IP state on all LAN port IP-enabled ports.	**show eou all**

This example shows how to display a summary of the LAN port IP state on all LAN port IP-enabled ports:

```
Console> (enable) show eou all
Eou Summary
-----------
Eou Global State = enabled

Currently Validating EOU Sessions = 0
mNo/pNo   Host Ip            Nac_Token   Host_Fsm_State   Username
-------   ----------------   ---------   --------------   --------
Console> (enable)
```

Displaying a Summary of the LAN Port IP State on a Per-Port Basis

To display a summary of the LAN port IP state on a per-port basis for LAN port IP-enabled ports, perform this task in normal mode:

Task	Command
Display a summary of the LAN port IP state on a per-port basis for LAN port IP-enabled ports.	**show port eou** *mod/port*

This example shows how to display a summary of the LAN port IP state on port 7/1:

```
Console> (enable) show port eou 7/1
Port     EOU-State IP Address      MAC Address
-------- --------- --------------- -----------------
 7/1     bypass    -               -

Port     FSM State     Auth Type   SQ-Timeout Session Timeout
```

```
-------- ------------- ----------- ---------- ---------------
 7/1      -             -           -          -

Port     Posture      URL Redirect
-------- ------------ --------------------
 7/1      -            -

Port     Termination action Session id
-------- ------------------ --------------------------------
 7/1      -                  -
Console> (enable)
```

Displaying Host-Specific Information

To display host-specific information, perform this task in normal mode:

Task	Command
Display host-specific information.	**show eou host** {**ip** | **mac**} *value* **show eou host mac_address** *mac_address*

This example shows how to display host-specific information:

```
Console> (enable) show eou host 9.6.2.15
HostIP          HostMac           Port   Posture-token
--------------- ----------------- ------ --------------------
9.6.2.15        00-11-85-8d-bf-ab 2/5    Healthy
IP Address      Eou State     AuthType SQTimeout SessTimeout
--------------- ------------- -------- --------- -----------
9.6.2.15        authenticated eap      301       3600
Console> (enable)
```

Displaying EOU Authentication-Related Information

To display the following authentication-related information, perform this task in normal mode:

- **clientless**—Display all clientless ports
- **eap**—Display all ports with EAP authentication
- **static**—Display all hosts in the exception list

Task	Command
Display authentication-related information.	**show eou authentication** {**clientless** | **eap** | **static**}

This example shows how to display authentication-related information:

```
Console> (enable) show eou authentication eap
Host IP         HostMac            Port    Posture-token
--------------- ------------------ ------- --------------------
9.6.2.15        00-11-85-8d-bf-ab  2/5     Healthy
IP Address      Eou State     AuthType  SQTimeout SessTimeout
--------------- ------------- --------- --------- -----------
9.6.2.15        authenticated eap       301       3600
```

```
Console> (enable)
```

Displaying the EOU Log

To display the EOU log, perform this task in normal mode:

Task	Command
Display the EOU log.	**show eou log**

This example shows how to display the EOU log:

```
Console> (enable) show eou log
LPIP-EVENT : New ip on port 3/12 9.9.150.21 from Arp-inspection
LPIP-ERROR : Failure to get host information for 9.9.143.20
LPIP-EVENT : Host 9.9.150.34 moved to EAPOUDP_TX_HELLO state
Console> (enable)
```

Displaying the EOU Results on a Posture-Token Basis

To display the EOU results on a posture-token basis, perform this task in normal mode:

Task	Command
Display the EOU results on a posture-token basis.	**show eou posture-token** *posture_token*

Clearing the LAN Port IP Configuration

To clear the LAN port IP configuration and return to default values, perform this task in privileged mode:

Task	Command
Clear the LAN port IP configuration and return to default values.	**clear eou config**

This example shows how to clear the LAN port IP configuration and return to default values:

```
Console> (enable) clear eou config
This command will disable EoU on all ports and take EoU parameter values back to defaults.
Do you want to continue (y/n) [n]? y
Console> (enable)
```

Clearing All the Host EOU Sessions

This command clears all the host EOU sessions learned on all the ports. It does not clear the EOU configuration.

To clear all the host EOU sessions learned on all the ports, perform this task in privileged mode:

Task	Command
Clear all the host EOU sessions learned on all the ports.	**clear eou all**

This example shows how to clear all the host EOU sessions learned on all the ports:

```
Console> (enable) clear eou all
Console> (enable)
```

Clearing the LAN Port IP Session for a Particular Host

To clear the LAN port IP session for a particular host by MAC address or IP address, perform this task in privileged mode:

Task	Command
Clear the LAN port IP session for a particular host by MAC address or IP address.	**clear eou host** {*ip-address* | *mac-address*}

This example shows how to clear an EOU session for a host with a specified IP address:

```
Console> (enable) clear eou host 9.9.10.10
EOU session of host with IP 9.9.10.10 cleared.
Console> (enable)
```

Clearing an IP Address from an Exception Group or Clearing an Exception Group

To clear an IP address from an exception group or clear an exception group, perform this task in privileged mode:

Task	Command
Clear an IP address from an exception group or clear an exception group.	**clear eou authorize ip** *ip-address* **policy** *policy_name* **clear eou authorize ip** *ip-address ip_mask* **policy** *policy_name* **clear eou authorize mac-address** *mac_address* **policy** *policy_name* **clear eou authorize mac-address** *mac_address mac_mask* **policy** *policy_name*

This example shows how to clear an IP address from an exception group:

```
Console> (enable) clear eou authorize ip 10.1.1.1 255.255.255.240 policy pol1
Cleared host 10.1.1.1 255.255.255.240 from exception group and removed its policy mapping.
Console> (enable)
```

Clearing EAPOUDP-Related Timers to Their Default Values

To clear EAPOUDP-related timers to their default values, perform this task in privileged mode:

Task	Command
Clear EAPOUDP-related timers to their default values.	**clear eou timeout** [**aaa** | **hold-period** | **restransmit** | **revalidation** | **status-query**]

This example shows how to clear the hold-period timers to their default values:

```
Console> (enable) clear eou timeout hold-period
Console> (enable)
```

Clearing the CTA Packet Retransmit Time

To clear the global CTA packet retransmit time, perform this task in privileged mode (this command sets the retransmit time back to the default value of 3):

Task	Command
Clear the global CTA packet retransmit time.	**clear eou max-retry**

This example shows how to clear the global CTA packet retransmit time:

```
Console> (enable) clear eou max-retry
Eou max-retry set to 3
Console> (enable)
```

Configuring Policy-Based ACLs

This section describes how to configure policy-based ACLs (PBACLs):

Adding IP Addresses to Existing Policy Groups

This command allows you to add an IP address to an existing policy group. The command fails if the group name is not already present in the group database.

To add an IP address to an existing policy group, perform this task in privileged mode:

Task	Command
Add an IP address to an existing policy group.	**set policy group** *group-name* **ip-address** *ip-address*

This example shows how to add an IP address to an existing policy group:

```
Console> (enable) set policy group grp1 ip-address 100.1.1.1 255.255.255.255
Added IP 100.1.1.1/255.255.255.255 to policy group grp1.
Console> (enable)
```

Adding a Policy Group to the Policy Template

You can add a policy group to the policy template. If a policy template does not exist, it is created. Similarly, if the policy group name does not exist, it is created.

To add a policy group to the policy template, perform this task in privileged mode:

Task	Command
Add a policy group to the policy template.	**set policy name** *policy-name* **group** *group-name*

This example shows how to add a policy group to the policy template:

```
Console> (enable) set policy name pol1 group grp1
Added group grp1 to policy template pol1.
Console> (enable)
```

Clearing an IP Address from a Policy Group

To clear an IP address from a policy group, perform this task in privileged mode:

Task	Command
Clear an IP address from a policy group.	**clear policy group** *group-name* **ip-address** *ip-address*

This example shows how to clear an IP address from a policy group:

```
Console> (enable) clear policy group grp1 ip-address 100.1.1.1
Cleared IP 100.1.1.1 from policy group grp1.
Console> (enable)
```

Clearing a Policy Group from a Policy Template

To clear a policy group from a policy template, perform this task in privileged mode:

Task	Command
Clear a policy group from a policy template.	**clear policy name** *policy-name* **group** *group-name*

This example shows how to clear a policy group from a policy template:

```
Console> (enable) clear policy name pol1 group grp1
Cleared group grp1 from policy template pol1.
Console> (enable)
```

Displaying Policy Group Information

To display policy group information, perform this task in normal mode:

Task	Command
Display policy group information.	**show policy group** {**all** | *group-name*}

This example shows how to display policy group information:

```
Console> (enable) show policy group all
Group Name            = grp1
Group Id                 = 1
No.of IP Addresses = 3
Src Type                 = ACL CLI
     List of Hosts in group.
     ----------------------
     Interface      = 0/0
     IpAddress    =  100.1.1.1
     Src type       = CONFIG

     Interface      = 0/0
     IpAddress    = 100.1.1.2
     Src type       = CONFIG

-------------------------------------------------
Group Name             = grp2
Group Id                  = 2
No.of IP Addresses  = 0
Src Type                 = ACL CLI
Console> (enable)
```

Displaying Policy Templates and Their Associated Policy Groups

To display policy templates and their associated policy groups, perform this task in normal mode:

Task	Command
Display policy templates and their associated policy groups.	**show policy name** {**all** | *policy-name*}

This example shows how to display policy templates and their associated policy groups:

```
Console> (enable) show policy name all
Policy Template pol1
Security Policy Groups :grp1 grp2
Console> (enable)
```

Configuring Inaccessible Authentication Bypass

When a switch cannot reach configured RADIUS servers and hosts cannot be authenticated, you can configure the switch to allow network access to the hosts connected to critical ports. A critical port is enabled by the inaccessible authentication bypass (IAB) feature.

When IAB is enabled, the switch checks the status of the configured RADIUS servers whenever the switch tries to authenticate a host connected to a critical port. If a server is available, the switch can authenticate the host. However, if all the RADIUS servers are unavailable, the switch grants network access to the host and puts the port in the critical-authentication state.

The operation function of the IAB feature depends on the authorization state of the port:

- If the port is unauthorized when a host connected to a critical port tries to authenticate and all servers are unavailable, the switch sends an EAP-success message to the host and puts the port in the critical-authentication state in the configured access VLAN.
- If the port is already authorized and reauthentication occurs, the switch puts the critical port in the critical-authentication state in the current VLAN, which might be the one previously assigned by the RADIUS server.
- If the RADIUS server becomes unavailable during an authentication exchange, the current exchanges times out, and the switch puts the critical port in the critical-authentication state during the next authentication attempt.

When the RADIUS server is available, all the ports in critical state are reinitialized if IAB initialization is enabled. Enable the IAB initialization feature by using the **set radius keepalive init [enable | disable]** command. The IAB initialization feature is disabled by default. If this feature is not enabled, the port waits until the reauthentication timer expires.

If IAB is enabled using the **set radius keepalive [enable | disable]** command, the switch sends periodic requests to the server. The interval between requests is configurable. Use the **set radius keepalivetimer** *time* command to set the timer. The server state can be in Init, CheckUp, Dead, or Alive state. During the initialization state, the first request is sent to all the RADIUS servers. The request waits for a response. If there is no response, the server state will be moved to Checkup. In the Checkup state, the switch sends two more requests to the server. If there is no response to the requests, the switch will be marked as “dead.” If there is a response to the request, the server will be marked as “alive.” To set the retry timer, use the **set radius timeout** *time* command to send a second request when there is no response to the first request.

The following sections describe how to configure IAB:

Enabling and Disabling Inaccessible Authentication Bypass

To enable or disable IAB, perform this task in enable mode:

Task	Command
Enable or disable IAB	**set port critical** *mod/port* **[disable | enable]**

This example shows how to enable IAB:

```
Console> (enable) set port critical 5/1 enable
Port, 5/1 Critical feature enabled.
Console> (enable)
```

This example shows how to enable IAB:

```
Console> (enable) set port critical 5/1 disable
Port, 5/1 Critical feature disabled.
Console> (enable)
```

Setting the AAA Fail Policy

To set the AAA fail policy, perform this task in enable mode:

Task	Command
Set the AAA fail policy.	**set port eou** *mod/port* **aaa-fail-policy** *policy-name*

This example shows how to set AAA fail policy for EOU:

```
Console> (enable) set port eou 12/1 aaa-fail-policy critical-eou-policy
Policy critical-eou-policy mapped as aaa-fail-policy on port 12/1
Console> (enable)
```

To set web-based proxy authentication on a port, perform this task in enable mode:

Task	Command
Set web-based proxy authentication on a port.	**set port webauth** *mod/port* **[aaa-fail-policy | disable | enable | initialize]** *policy-name*

This example shows how to enable webauth on a port:

```
Console> (enable) set port web-auth 5/1 enable
Port 5/1 Web-auth is enabled.
Console> (enable)
```

This example shows how to set AAA fail policy for webauth:

```
Console> (enable) set port web-auth 12/1 aaa-fail-policy critical-webauth-policy Policy
critical-webauth-policy set as web-auth aaa-fail-policy for port 12/1
Console> (enable)
```

Setting the RADIUS Keepalive Timer

To enable or disable the RADIUS keepalive timer, perform this task in enable mode:

Task	Command
Set the RADIUS keepalive timer.	**set radius keepalive [enable | disable]**

This example shows how to enable the RADIUS keepalive timer:

```
Console> (enable) set radius keepalive enable
Radius Keepalive enabled.
```

This example shows how to disable the RADIUS keepalive timer:

```
Console> (enable) set radius keepalive disable
Radius Keepalive disabled.
```

Setting the RADIUS Auto-Initialize Feature

To enable or disable the RADIUS auto-initialize feature, perform this task in enable mode:

Task	Command
Set the RADIUS auto-initialize feature.	**set radius auto-initialize** [**enable/disable**]

This example shows how to enable the RADIUS auto-initialize feature:

```
Console> (enable) set radius auto-initialize enable
Radius Auto-initialize enabled.
```

This example shows how to disable the RADIUS auto-initialize feature:

```
Console> (enable) set radius auto-initialize disable
Radius Auto-initialize disabled.
```

Displaying the Critical Status of Features on a Port

To display the critical status of features on a port, perform this task in enable mode:

Task	Command
Display the critical status of features on a port.	**show port critical** *mod/port*

This example shows how to display the critical status of features on a port:

```
Console> (enable) show port critical 5/1
Port  Critical State Features in Critical State
-----  -------------- --------------------------
 5/1  enabled          dot1x, eou
```

Displaying the AAA Fail Policy on a Port

To display the AAA fail policy for EOU on a port, perform this task in enable mode:

Task	Command
Display the AAA fail policy for EOU on a port.	**show port eou** *mod/port* **aaa-fail-policy**

This example shows how to display AAA fail policy on a port:

```
Console> (enable) show port eou 5/1 aaa-fail-policy
Port    AAA-Fail-Policy
-----   ------------------
5/1
```

To display the AAA fail policy for web-auth on a port, perform this task in enable mode:

Task	Command
Display the AAA fail policy for web-auth on a port.	**show port web-auth** *mod/port* **aaa-fail-policy**

This example shows how to display AAA fail policy on a port:

```
Console> (enable) show port web-auth 5/1 aaa-fail-policy
Port    AAA-Fail-Policy
-----   ------------------
5/1
```

Displaying RADIUS Server Information

To display RADIUS server information, perform this task in enable mode:

Task	Command
Display RADIUS server information.	**show radius**

This example shows how to display RADIUS server information:

```
Console> (enable) show radius
Active RADIUS Server          : 0.0.0.0
RADIUS Deadtime               : 0 minutes
RADIUS Key                    :
RADIUS Retransmit             : 2
RADIUS Timeout                : 5 seconds
Framed-Ip Address Transmit    : Disabled
RADIUS Framed MTU             : 1000 bytes
RADIUS Keepalive              : Enabled
RADIUS Keepalive Timer        : 0 minutes
RADIUS Autoinitialize Critical: Disabled

RADIUS-Server                   Status  Auth-port Acct-port Resolved IP Addrese
------------------------------- ------- --------- --------- -------------------
```

Displaying the MAC Authorization Bypass Settings on a Port

To display the MAC authorization bypass settings on a port, perform this task in enable mode:

Task	Command
Display the MAC authorization bypass settings on a port.	**show port mac-auth-bypass** *mod/port*

This example shows how to display the MAC authorization bypass settings on a port:

```
Console> (enable) show port mac-auth-bypass 5/1
Port  Mac-Auth-Bypass State MAC Address       Auth-State         Vlan
----- --------------------- ----------------- ------------------  -----
 5/1  Disabled              -                 -                  1

Port  Termination action Session Timeout Shutdown/Time-Left
----- ------------------ --------------- ------------------
 5/1  -                  3600            NO       -

Port  PolicyGroups
----- ---------------------------------------------------
 5/1   -

Port  Critical Critical-Status
----- -------- ---------------
 5/1  Disabled -
Console> (enable)
```

Displaying the Web Authorization Settings on a Port

To display web authorization settings on a port, perform this task in enable mode:

Task	Command
Display the web authorization settings on a port.	**show port web-auth** *mod/port*

This example shows how to display the web authorization settings on a port:

```
Console> (enable) show port web-auth 5/1
```

```
Port  IP-Address      Vlan Enabled   Web-Auth-State        Critical-Status
----- --------------- ---- --------- ------------------     ---------------
 5/1   -               1    enabled   -                      -

Port  IP-Address      Session-Timeout Session-Timeleft Radius-Rcvd-Timeout
----- --------------- --------------- ---------------- --------------------
 5/1   -               -               -                No

Port  IP-Address      Policy-Groups
----- --------------- -------------
 5/1   -               -
```

Displaying the EOU Settings on a Port

To display the EOU settings on a port, perform this task in enable mode:

Task	Command
Display the EOU settings on a port.	**show port eou** *mod/port*

This example shows how to display the EOU settings on a port:

```
Console> (enable) show port eou 5/1
Port     EOU-State IP Address      MAC Address       Critical-Status
-------- --------- --------------- ----------------- ---------------
 5/1     disabled  -               -                 -

Port     FSM State     Auth Type   SQ-Timeout Session Timeout
-------- ------------- ----------- ---------- ---------------
 5/1      -             -           -          -

Port     Posture      URL Redirect
-------- ------------ --------------------
 5/1      -            -

Port     Termination action Session id
-------- ------------------ --------------------------------
 5/1      -                  -

Port     PolicyGroups
-------- ----------------------------------------------------
 5/1      -

Port  Critical
----- --------
 5/1  enabled
```

Clearing Policy Mapping on a Port

To clear the policy mapping on a port, perform this task in enable mode:

Task	Command
Clear the EOU policy mapping on a port.	**clear port eou** *mod/port* **aaa-fail-policy**

This example shows how to clear the EOU policy mapping on a port:

```
Console> (enable) clear port eou 5/1 aaa-fail-policy
```

```
aaa-fail-policy cleared successfully on port 5/1
```

To clear the web-based proxy authentication mapping on a port, perform this task in enable mode:

Task	Command
Clear the webauth policy mapping on a port.	**clear port webauth** *mod/port* **aaa-fail-policy**

This example shows how to clear the webauth policy mapping on a port:

```
Console> (enable) clear port webauth 5/1 aaa-fail-policy
aaa-fail-policy cleared successfully on port 5/1
```

LAN Port IP Configuration Example

Use this configuration example when configuring LAN port IP:

- Port 8/14 connects to the RADIUS server
- Port 8/13 connects to the host with CTA
- Port 8/24 connects to the host without CTA

```
begin
!
# ***** NON-DEFAULT CONFIGURATION *****
!
!
#time: Fri Mar 4 2005, 17:11:20
!
#version 8.5(0.44)JAC
!
!
#Nac
set eou enable
set eou allow clientless enable
set policy name exception_policy group exception_hosts
set eou authorize ip 77.0.0.90 policy exception_policy
!
#radius
set radius server 10.76.39.93 auth-port 1812 primary
set radius key cisco
!
#vtp
set vtp mode transparent vlan
set vlan 12 name RADIUS_CONNECTIVIY type ethernet mtu 1500 said 100012 state active
set vlan 77 name ALL_HOSTS type ethernet mtu 1500 said 100077 state active
set vlan 1,3
!
#ip
set interface sc0 12 9.6.3.3/255.255.255.0 9.6.3.255
set interface sl0 down
set interface sc1 77 77.0.0.2/255.255.255.0 77.0.0.255
set ip route 10.0.0.0/255.0.0.0 9.6.3.1
!
!
#security ACLs
clear security acl all
#NACACL
set security acl ip NACACL permit arp
set security acl ip NACACL permit arp-inspection any any
```

```
set security acl ip NACACL permit dhcp-snooping
set security acl ip NACACL permit udp any eq 21862 host 9.6.3.3 eq 53000
set security acl ip NACACL permit ip group Healthy_hosts any
set security acl ip NACACL deny ip group infected_hosts any
set security acl ip NACACL permit ip group exception_hosts any
set security acl ip NACACL permit ip group clientless_hosts host 10.76.39.100
#
commit security acl all #
# map the ACL to VLAN 77
set security acl map NACACL 77
!
#module 8 : 48-port 10/100BaseTX Ethernet
set vlan 12 8/14
set vlan 77 8/13,8/24
set port name 8/13 HOSTS
set port name 8/14 RADIUS
set port name 8/24 HOSTS
set port eou 8/13 enable
set port eou 8/24 bypass
set port dhcp-snooping 8/14 trust enable
!
#module 9 empty
!
#module 15 : 1-port Multilayer Switch Feature Card
!
#module 16 empty
!
#switch port analyzer
set span permit-list disable
set span permit-list include
end
sup2> (enable)
```

The configuration on the MSFC (default router) is as follows:

```
Router# show run
Building configuration...
Current configuration : 509 bytes
!
version 12.1
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Router
!
!
ip subnet-zero
!
!
!
ip multicast-routing
ip dhcp-server 10.76.39.93
redundancy
 high-availability
 single-router-mode
!
!
!
interface Vlan12
 ip address 9.6.3.6 255.255.255.0
!
interface Vlan77
 ip address 77.0.0.76 255.255.255.0
```

```
 ip helper-address 10.76.39.93
!
ip classless
ip route 10.76.0.0 255.255.0.0 Vlan12
no ip http server
!
!
!
line con 0
line vty 0 4
 login
!
!
end
```

LAN Port IP Enhancements in Software Release 8.6(1) and Later Releases

These sections describe the enhancements for configuring NAC with LAN port IP in software release 8.6(1) and later releases:

- Configuring URL Redirect Support for LAN Port IP Exception Hosts, page 44-32
- Configuring LAN Port IP on Private VLAN Ports, page 44-34

Configuring URL Redirect Support for LAN Port IP Exception Hosts

Exception hosts (such as printers and IP phones) cannot validate posture. The IP/MAC addresses of the exception hosts are added to an exception list. When a host in the exception list is detected on an interface, a preconfigured policy is installed.

For normal, nonexception hosts, URL redirection is accomplished through information that is received from the RADIUS server after a successful posture validation. Because the RADIUS server is not contacted, exception hosts must find a way to access a server, or you must provide a URL through which the hosts can download software components (such as antivirus updates).

Configuration Guidelines and Restrictions

Follow these configuration guidelines and restrictions when configuring URL redirect for LAN port IP exception hosts:

- URL redirection is not supported on multiple-host and multiple-authentication ports.
- URL redirection works only if there is a VACL with ARP inspection and DHCP snooping mapped on the VLAN of the port.
- Because Supervisor Engine 1 does not support ARP inspection, URL redirection is not supported on Supervisor Engine 1.

Specifying the Policy Name and URL Redirect String

The **set policy name** *policy-name* **url-redirect** *url-redirect-string* command maps a URL redirect string to a policy name. URL strings of up to 255 characters are allowed. If the URL string exceeds 255 characters, the command fails.

To specify the policy name and URL redirect string, perform this task in privileged mode:

Task	Command
Specify the policy name and URL redirect string.	**set policy name** *policy-name* **url-redirect** *url-redirect-string*

This example shows how to specify the policy name and URL redirect string:

```
Console> (enable) set policy name exception_policy url-redirect http://cisco.com
Url Redirect http://cisco.com mapped to policy name exception_policy
Console> (enable)
```

Displaying the Policy Name and URL Redirect String Mapping

To display the policy name and URL redirect string mapping, perform this task in normal mode:

Task	Command
Display the policy name and URL redirect string mapping.	**show policy name** [**all** | *policy-name*]

This example shows how to display the policy name and URL redirect string mapping for the specified policy:

```
Console> (enable) show policy name exception_policy
Policy Name : exception_policy
-----------------------------------------------
URL-Redirect : http://cisco.com
Console> (enable)
```

This example shows how to display the policy names and URL redirect string mappings for all policies:

```
Console> (enable) show policy name all
Policy Name : TEST
-----------------------------------------------
Associated IP Address/Mask Information:
0.0.0.18/255.255.255.224
Policy Name : poll
-----------------------------------------------
Associated IP Address/Mask Information:
0.0.0.19/255.255.255.224
Policy Name : BLDG_F
-----------------------------------------------
Policy Name : exception_policy
-----------------------------------------------
URL-Redirect : http://cisco.com
Console> (enable)
```

Clearing the URL Redirect String Associated with the Policy Name

To clear the URL redirect string associated with the policy name, perform this task in privileged mode:

Task	Command
Clear the URL redirect string associated with the policy name.	**clear policy name** *policy-name* **url-redirect**

This example shows how to clear the URL redirect string associated with the policy name:

```
Console> (enable) clear policy name exception_policy url-redirect
Cleared url-redirect for the policy exception_policy
Console> (enable)
```

Configuring LAN Port IP on Private VLAN Ports

Note For detailed information on private VLANs, see the "Configuring Private VLANs on the Switch" section on page 11-19.

A private VLAN port is associated with two VLANs, the primary VLAN and the secondary VLAN. Traffic coming from the host (ingress traffic) is tagged with the secondary VLAN and traffic coming from the router port is tagged with the primary VLAN. To trigger EOU on a port, an ARP inspection or DHCP snooping ACL must be mapped to the port VLAN. To trigger EOU on a port in a private VLAN, you must map an ARP inspection or DHCP snooping ACL explicitly to the secondary VLAN as it is the VLAN that is associated with the ingress traffic.

Different PBACLs can be mapped to the primary and secondary VLANs. After a successful posture validation, if the PBACL that is mapped to the primary and secondary VLAN have groups where the host is a member, they are expanded to accommodate the IP address of the host.

Configuring Network Admission Control with LAN Port 802.1X

These sections describe how to configure NAC with LAN port 802.1X:

Understanding How Network Admission Control with LAN Port 802.1X Works

Note There are no LAN port 802.1X-specific CLI commands. Posture validation and authentication occur seamlessly inside a single EAP tunnel through standard 802.1X authentication. For information on configuring IEEE 802.1X authentication, see Chapter 40, "Configuring 802.1X Authentication."

Note The restrictions that apply to LAN port IP also apply to LAN port 802.1X. For LAN port IP restrictions, see the "LAN Port IP Configuration Guidelines and Restrictions" section on page 44-6.

LAN port 802.1X combined with standard 802.1X authentication provides a unified authentication and posture validation mechanism at the Layer 2 network edge. LAN port 802.1X acts at the same point in the network as LAN port IP but uses different mechanisms to initiate posture validation, to carry the communication between host and authentication server, and to enforce the resulting access limitations.

Posture validation in LAN port 802.1X is triggered by the standard 802.1X mechanisms (either the supplicant sends an EAPOL-Start message to the NAD, or the NAD probes the supplicant with an EAP-Request/Identity message); the posture information may be sent with the user identity credentials

for validation by the back-end server. The authentication exchange between the supplicant and the NAD is over EAPOL. Policy enforcement is done by assigning the authenticated port to a specified VLAN to provide segmentation and quarantine of poorly postured hosts at Layer 2.

Note LAN port 802.1X restricts non-IPv4 traffic from nonpostured hosts. LAN port 802.1X is preferred for deployments where such a restriction is a requirement.

The LAN port 802.1X policy enforcements include the following (which are already supported with standard 802.1X authentication):

- VLAN assignment—Normal native VLAN assignment (private VLAN assignment is not supported with LAN port 802.1X).
- Security ACL assignment—A PBACL name comes from the RADIUS server, it is assigned to the port interface, and it could be a PACL or VACL.
- Policy groups—PBACL policy groups can be sent down from the ACS server.

For LAN port 802.1X, the policy enforcement uses a VLAN/PBACL combination where LAN port IP uses only PBACLs.

Reauthentication works the same way as in standard 802.1X authentication which makes use of the RADIUS server-sent session timeout and termination action attributes or the local CLI-configured attributes. These attributes are not received as part of the Access-Accept message from the RADIUS server.

With LAN port 802.1X, hosts are classified into one of the following categories:

- Enhanced CTA—This CTA can send both authentication and posture TLVs in a single EAP tunnel and the policy enforcement that comes from the RADIUS server has both the VLAN assignment and the PBACL groups.
- Legacy supplicants and legacy CTA—These hosts do not have the enhanced CTA; they have the standard 802.1X supplicant that cannot connect to CTA and they also have the legacy CTA that can do posture validation using EAPoUDP. With these hosts, after LAN port 802.1X completes, the switch checks for posture validation results. If the posture results are not received, it is assumed that the host does not have enhanced CTA. If LAN port IP is configured on the port, it is triggered to do the posture validation. This category is a combination of LAN port IP and 802.1X authentication.
- Legacy supplicant and no CTA—These 802.1X-capable hosts do not have CTA. After 802.1X authentication completes, the switch realizes that posture validation has not occurred and if LAN port IP is enabled on the port, the switch directs LAN port IP to carry out the posture validation. When LAN port IP runs, it realizes that the host is not responding to its EoU packets and downloads the clientless posture policy for the host. In contrast, 802.1X authentication would have an enforced policy based on the authentication result.
- No supplicant and legacy CTA—When the host does not have an 802.1X-capable supplicant, 802.1X times out and moves the port into the guest VLAN or if MAC authentication bypass is configured, MAC authentication bypass is requested to authenticate the host's MAC address. After authorizing the port (through MAC authentication bypass or the guest VLAN), if LAN port IP is configured, LAN port IP does the posture validation and retrieves the posture policy.
- No supplicant and no CTA—When a dumb host is connected to a switch port that is not 802.1X-capable or does not have a CTA installed, the switch initially tries EAPOL exchanges. When it fails to get a response, the switch moves the port into the guest VLAN state or requests that MAC address bypass (if configured) authenticate the MAC address. Once the port is authorized by one of

these features, the switch requests that the LAN port IP (if configured) does the posture validation. LAN port IP realizes that its Hello messages are not getting any response and does a clientless authentication to retrieve the posture policy for nonresponsive hosts.

LAN Port 802.1X Enhancements in Software Release 8.6(1) and Later Releases

These sections describe the enhancements for configuring NAC with LAN port 802.1X in software release 8.6(1) and later releases:

- URL Redirection Support for LAN Port 802.1X, page 44-36
- Enabling and Disabling the Session Timeout Override for LAN Port 802.1X, page 44-37

URL Redirection Support for LAN Port 802.1X

After a successful LAN port 802.1X authentication, you can redirect HTTP traffic to the supervisor engine using URL redirection. URL redirection requires that you configure an ACL with an ACE that will redirect all ingress traffic with destination TCP port 80 to the supervisor engine. Enter the **set security acl ip** *acl-name* **permit url-redirect** command to create the ACE. Any ACL that is mapped to a port/VLAN with this ACE redirects all HTTP traffic to the supervisor engine.

URL redirection requires that the IP address of an authenticated host appears in a URL redirect list. The IP address of the host can be obtained in three ways:

- Framed IP address sent from the RADIUS server
- DHCP snooping
- ARP inspection

DHCP snooping is given the highest precedence, followed by ARP inspection, and then framed IP. If the IP address is received through a higher precedence mechanism than the current one and the previous IP address differs from the current one, the installed policies are removed and updated with the latest IP address. Also, the host IP address added to the URL redirect list is updated with the preferred IP address.

As a result of URL redirection, the NAD intercepts all HTTP traffic from the host that matches the URL redirect match ACL (configured locally or downloaded from the ACS). The intercepted HTTP TCP session is terminated at the NAD. The URL redirect feature then invokes the feature-specific handler that posts an HTTP 302 redirect status code to the host over the terminated TCP session in the following format:

```
HTTP/1.1 302 Page Moved
Location: <REDIRECT URL-ADDRESS>
Pragma: no-cache
Cache-Control: no-cache
```

The redirect URL address is sent from the RADIUS server. When the host browser receives the 302 status code, it initiates a new HTTP request to the provided redirected URL address and the redirection occurs.

The redirect URL that is sent from the RADIUS server needs to be configured on the RADIUS server. A typical URL redirect VSA would be as follows:

```
Url-redirect=<url-address>
```

To prevent all the HTTP packets from being redirected to software by the ACL on the interface, you must ensure that packets destined to the redirected URL are not redirected to the software for URL redirection. The ACL must have an ACE installed so that it occurs before the URL redirection ACE that permits traffic to the redirected host. Installing the ACE in this position ensures that the redirected request will encounter the prepositioned ACE and will not be intercepted by the supervisor engine.

A host can be added to URL redirection through the LAN port IP, web-based proxy authentication, and LAN port 802.1X. Web-based proxy authentication is given the highest precedence, followed by LAN port IP, and then LAN port 802.1X. The host port is opened only after a successful 802.1X authentication. When the host tries to access the web, it has to be authenticated through web-based proxy authentication, followed by posture validation by LAN port IP. The host is permitted to access the URL that is received from the RADIUS server after a successful 802.1X authentication.

For URL redirection to work with LAN port 802.1X, there must be an ACL mapped to the VLAN of the port that has DHCP snooping, ARP inspection, and the URL redirect ACE.

Enabling and Disabling the Session Timeout Override for LAN Port 802.1X

After a successful 802.1X authentication, and if reauthentication is enabled on a port, 802.1X authentication will reauthenticate the port when the reauthentication timer expires. The reauthentication timer value can be configured through the CLI or can be sent from the RADIUS server. The **set port dot1x** *mod/port* **re-authperiod server** {**disable** | **enable**} command allows you to specify whether the reauthentication timer value from the RADIUS server will be used or whether the CLI-configured value will be used. By default, the session timeout value that is received from the RADIUS server takes precedence over the CLI-configured timeout value. See Table 44-1 for suggested session timeout override mapping values.

Table 44-1 Session Timeout Override Mapping Values

Reauthorization Enabled	Reauthorization Period from Server Enabled	Session Timeout Received	Termination Action	NAS Action
No	Optional	n/a	n/a	No reauthorization
Yes	No	n/a	n/a	Reauthorization with local timer
Yes	Yes	No	n/a	No reauthorization
Yes	Yes	Yes	Default or no action	Termination with RADIUS timer
Yes	Yes	Yes	RADIUS request	Reauthorization with RADIUS timer

Note If you enable 802.1X IAB on a port that is already authenticated, if the RADIUS server is not reachable during reauthentication, then the port remains in the authenticated state.

To enable or disable the session timeout override for LAN port 802.1X, perform this task in privileged mode:

Task	Command
Enable or disable the session timeout override for LAN port 802.1X.	**set port dot1x** *mod/port* **re-authperiod server** {**disable** | **enable**}

This example shows how to enable the session timeout override for LAN port 802.1X:

```
Console> (enable) set port dot1x 5/8 re-authperiod server enable
Port 5/8 session-timeout-override option is enabled
Console> (enable)
```

This example shows how to display the session timeout override setting for LAN port 802.1X:

```
Console> (enable) show port dot1x 5/8
Port  Auth-State          BEnd-State Port-Control        Port-Status
----- ------------------- ---------- ------------------- -------------
 5/8  -                   -          force-authorized    -

Port  Port-Mode     Re-authentication   Shutdown-timeout   Control-Mode
                                                          admin   oper
----- ------------- -----------------   ----------------   ---------------
 5/8  SingleAuth    disabled            disabled           Both    -

Port  Posture-Token Critical-Status Termination action Session-timeout
----- ------------- --------------- ------------------ ---------------
 5/8  -             -               -                  -

Port  Session-Timeout-Override Url-Redirect
----- ----------------------- ----------------------------------
 5/8  enabled                 -

Port  Critical
----- --------
 5/8  disabled
Console> (enable)
```

CHAPTER 45

Configuring Unicast Flood Blocking

This chapter describes how to configure unicast flood blocking on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Understanding How Unicast Flood Blocking Works

You can enable unicast flood blocking on any Ethernet port on a per-port basis. Unicast flood blocking provides you the option to drop the unicast flood packets on an Ethernet port that has only one host that is connected to the port. All the Ethernet ports on a switch are configured to allow unicast flooding; unicast flood blocking allows you to drop the unicast flood packets before they reach the port.

Caution You must have a static CAM entry that is associated with the Ethernet port before you enable unicast flood blocking. If you do not have a static CAM entry that is associated with the port, you will lose network connectivity if you enable unicast flood blocking. You can verify that a static CAM entry exists by entering the **show cam static** command.

Note If you are configuring unicast flood blocking on a secure port, see Chapter 38, "Configuring Port Security."

Unicast Flood Blocking Configuration Guidelines

This section describes the guidelines for configuring unicast flood blocking:

- Only the Ethernet ports can block the unicast flood traffic.
- If the Ethernet port is part of an IPX network, you must manually enter a static CAM entry in the CAM table before you disable unicast flood blocking on the port.
- You cannot configure unicast flood blocking on the SPAN destination ports.
- You cannot configure a SPAN destination on a unicast flood blocking port.
- You cannot configure unicast flood blocking on a trunk port. If you attempt to configure unicast flood blocking on a trunk port, you will see an error message.
- You cannot configure unicast flood blocking on a port channel.
- You cannot configure a port channel on a unicast flood blocking port.
- Unicast flood blocking and GARP VLAN Registration Protocol (GVRP) are mutually exclusive. You cannot configure the port to block the unicast flood packets and exchange VLAN configuration information with the GVRP switches at the same time.

Configuring Unicast Flood Blocking on the Switch

These sections describe how to configure unicast flood blocking:

- Enabling Unicast Flood Blocking, page 45-2
- Disabling Unicast Flood Blocking, page 45-3
- Displaying Unicast Flood Blocking, page 45-3

Note When you are configuring unicast flood blocking, it is important to note that unicast flood blocking is given priority over other features (such as protocol filtering).

Enabling Unicast Flood Blocking

To configure the switch to drop the unicast flood packets on a port, you must disable unicast flood blocking.

Note The port disables unicast flooding once the MAC address limit is reached.

To enable unicast flood blocking, perform this task in privileged mode:

Task	Command
Enable unicast flood blocking on the desired Ethernet ports to disable unicast flooding.	**set port unicast-flood** *mod/port* **disable**

This example shows how to disable unicast flood blocking on a port:

```
Console> (enable) set port unicast-flood 4/1 disable
WARNING: Trunking & Channelling will be disabled on the port.
Unicast Flooding is successfully disabled on the port 4/1.
Console> (enable)
```

Disabling Unicast Flood Blocking

To configure the switch to receive the unicast flood packets on a port, you must enable unicast flood blocking.

To disable unicast flood blocking, perform this task in privileged mode:

Task	Command
Disable unicast flood blocking on the desired Ethernet ports to enable unicast flooding.	**set port unicast-flood** *mod/port* **enable**

This example shows how to disable unicast flood blocking on a port:

```
Console> (enable) set port unicast-flood 4/1 enable
Unicast Flooding is successfully enabled on the port 4/1.
Console> (enable)
```

Displaying Unicast Flood Blocking

To display unicast flood blocking information, perform this task in privileged mode:

Task	Command
Display unicast flood blocking information per port.	**show port unicast-flood** *mod/port*

This example shows how to display unicast flood blocking information for port 1 on module 4:

```
Console> (enable) show port unicast-flood 4/1
Port      Unicast Flooding
----      ----------------
4/1       Disabled
Console> (enable)
```

CHAPTER 46

Configuring the Switch Fabric Modules

This chapter describes the integrated 720-Gbps switch fabric that is supported on Supervisor Engine 720 and the external Switch Fabric Module (WS-C6500-SFM) and Switch Fabric Module 2 (WS-X6500-SFM 2) that is supported on the Supervisor Engine 2 on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter uses the following terminology:

- CEF720—Any module that has a part number that conforms to WS-X67*xx-xxx* (such as WS-X6724-SFP). These modules connect to the integrated 720-Gbps switch fabric on the Supervisor Engine 720.
- CEF256—Any module that has a part number that conforms to WS-X65*xx-xxx* (such as WS-X6548-GE-TX), the Optical Services Modules, the enhanced FlexWAN module, and most service modules, such as the Firewall Services Module (FWSM), the Secure Socket Layer Services Module (SSLM), the Virtual Private Network Services Module (VPNSM), the Network Analysis Module 1 (NAM-1), the NAM-2, the Intrusion Detection System Module (IDSM-2), the Content Services Gateway (CSG), and the Communications Media Module (CMM). These modules connect to either the integrated 720-Gbps switch fabric on the Supervisor Engine 720 or to the external 256-Gbps Switch Fabric Modules that are supported by the Supervisor Engine 2, and they connect to the 32-Gbps switching bus.
- Non-fabric-enabled—Any module that does not fall into the CEF720 or CEF256 categories. These modules have no fabric connections and connect only to the 32-Gbps switching bus.

This chapter consists of these sections:

Note The WS-C6500-SFM is supported in the Catalyst 6500 series 6-and 9-slot chassis only. The WS-X6500-SFM 2 is supported in the Catalyst 6500 series 6-slot, 9-slot, and 13-slot chassis and in the Catalyst 6509-NEB chassis.

Understanding How the Integrated 720-Gbps Switch Fabric Works

Note The integrated 720-Gbps switch fabric is supported only on Supervisor Engine 720.

Note With software release 8.3(1) and later releases, in redundant systems, the integrated 720-Gbps switch fabric supports a high-availability failover to the standby switch fabric. High availability must be enabled for the failover to work (enter the **set system highavailability enable** command).

The integrated 720-Gbps switch fabric, which is built into Supervisor Engine 720, creates a dedicated connection between the CEF256 and CEF720 modules and provides an uninterrupted transmission of frames between these modules. In addition to the direct connection between the CEF256 and CEF720 modules that is provided by the integrated 720-Gbps switch fabric, these modules have a direct connection to the 32-Gbps switching bus.

Understanding How the External Switch Fabric Module Works

Note The external Switch Fabric Modules are supported only with Supervisor Engine 2 in the Catalyst 6500 series switch.

The external Switch Fabric Modules create a dedicated connection between the CEF256 modules and provide an uninterrupted transmission of frames between these modules. The external Switch Fabric Modules do not support the CEF720 modules because these modules do not work with a Supervisor Engine 2. In addition to using the Switch Fabric Module, the CEF256 modules connect to the 32-Gbps switching bus. Table 46-1 lists the switch fabric connection speed for each module.

Table 46-1 Switch Fabric Connection Speed by Module Type

Switch Fabric Type	Module Type[1]	
	CEF256	CEF720
Switch Fabric Module (256-Gbps)	1 x 16-Gbps	Not supported
Switch Fabric Module 2 (256-Gbps)	1 x 16-Gbps	Not supported
Integrated switch fabric on Supervisor Engine 720 (720-Gbps)	1 x 16-Gbps	1 x 40-Gbps: WS-X6724-SFP 2 x 40-Gbps: WS-X6748-GE-TX WS-X6748-SFP WS-X6704-10GE

1. The speeds shown are bidirectional.

Enter the **set system crossbar-fallback bus-mode | none** command to specify how the packets are handled if the Switch Fabric Module is removed or fails. If you specify the **bus-mode** keyword, the switching is done in flow-through mode. If you specify the **none** keyword, the switch ports are disabled and switching stops. For more detailed information, see the "Configuring a Fallback Option" section on page 46-4.

The external Switch Fabric Module does not have a console. A two-line LCD display on the front panel shows fabric utilization, software revision, and basic system information.

Note The term *9-slot switch* refers to the WS-C6509-NEB and WS-C6509-NEB-A switches.

Install the WS-C6500-SFM in either slot 5 or 6 in the 6-slot and 9-slot switches. Install the WS-X6500-SFM 2 in either slot 7 or 8 in the 13-slot switches and slots 5 or 6 in the 6-slot and 9-slot switches. The Switch Fabric Module that is first installed functions as the primary module. For redundancy, you can install a standby Switch Fabric Module.

When you install two Switch Fabric Modules at the same time in a 6- or 9-slot chassis, insert the primary module into slot 5 and the backup module into slot 6. If you reset the module in slot 5, the module in slot 6 becomes active.

When you install two Switch Fabric Modules at the same time in a 13-slot chassis, insert the primary module into slot 7 and the backup module into slot 8. If you reset the module in slot 7, the module in slot 8 becomes active.

Forwarding Modes

The CEF256/CEF720 modules operate in one of three modes when using centralized forwarding:

- Compact mode—Operational mode when all modules in the system are CEF256 or CEF720 (no non-fabric-enabled modules can be present for this mode). In this mode, the CEF256/CEF720 modules send a "compact" 32-byte header for each frame to the supervisor engine over the switching bus. Once a forwarding decision is made, the CEF256/CEF720 modules send the entire frame through the switch fabric to the egress module.
- Truncated mode—Operational mode when at least one non-fabric-enabled module is present in the system. In this mode, the CEF256/CEF720 modules send the first 64 bytes of each frame to the supervisor engine over the switching bus. Once a forwarding decision is made, the CEF256/CEF720 modules send the entire frame through the switch fabric to the egress module.
- Flow-through mode—Operational mode for the CEF256 modules when there is no switch fabric present. In this mode, the CEF256 modules send the entire packet to the supervisor engine over the switching bus. This mode is not applicable for the CEF720 modules, which require the presence of the switch fabric.

Table 46-2 shows the switch modes that are used with the CEF256, CEF720, and non-fabric-enabled modules installed.

Table 46-2 Switching Modes with Switch Fabric Module Installed

Types of Modules Installed	Switching Modes
CEF256 or CEF720 modules (no non-fabric-enabled modules are installed)	Compact
CEF256 and/or CEF720 modules (non-fabric-enabled modules are installed)	Truncated
CEF256 and non-fabric-enabled modules	Flow-through
Non-fabric-enabled modules	Flow-through

Configuring and Monitoring the Integrated Switch Fabric and Switch Fabric Module on the Switch

The integrated 720-Gbps switch fabric and the Switch Fabric Modules do not require any user configuration but support a number of **show** commands for monitoring purposes. A fully automated startup sequence brings the module online and runs the connectivity diagnostics on the ports.

From the supervisor engine, you can reset the module by entering the **reset** *module* command, disable and enable the module by entering the **set module enable | disable** command, and power down the module by entering the **set module powerdown** *module* command.

These sections describe how to configure the integrated 720-Gbps switch fabric and Switch Fabric Modules:

Configuring a Fallback Option

The **set system crossbar-fallback** {**bus-mode | none**} command allows you to configure a fallback option if the Switch Fabric Module connection fails.

If a switch is in compact mode, setting the crossbar-fallback to **none** prohibits the switch from running in any other switching mode. Assuming there are no non-fabric-enabled modules in the switch, the switching mode will be compact. Then if the crossbar-fallback is set to **none** and a non-fabric- enabled module is inserted, the module will not be powered up because that would force the switch out of compact mode. Also, if crossbar-fallback is set to **none** and the last Switch Fabric Module is removed, ports on all modules will be disabled. If a port fails to become disabled, the module on which it resides will be powered down. With no Switch Fabric Module, the switch cannot run in compact mode so the switch is effectively disabled since crossbar-fallback is set to **none**.

To configure a fallback option for the Switch Fabric Module, perform this task in privileged mode:

Task	Command
Configure a fallback option for the Switch Fabric Module.	**set system crossbar-fallback** {**bus-mode** | **none**}

This example shows how to configure a fallback option to bus-mode:

```
Console> (enable) set system crossbar-fallback bus-mode
System crossbar-fallback set to bus-mode.
Console> (enable)
```

Configuring the Switching Mode

To improve performance, you can manually specify the switching mode that the system uses. If you have one or more non-fabric-enabled modules that are installed in the chassis, configure the switch to use flow-through mode. If you have only the CEF256 or CEF720 modules that are installed in the chassis, configure the switch to use compact mode.

Note Non-fabric-enabled modules do not support compact mode.

Note If there is a combination of a Supervisor Engine 720 with switch-fabric capability and CEF720 modules in the chassis, the **bus-only** operation is not permitted. The system stays in truncated mode.

To configure the switch to use flow-through mode if you have the non-fabric-enabled modules installed, perform this task:

Task	Command
Configure the switch to use flow-through mode.	**set system switchmode allow bus-only**

This example shows how to configure the switch to use flow-through mode:

```
Console> (enable) set system switchmode allow bus-only

Console> (enable)
```

To configure the switch to use truncated mode when CEF256 and/or CEF720 and non-fabric-enabled modules are installed, perform this task:

Task	Command
Configure the switch to use truncated mode.	**set system switchmode allow truncated**

This example shows how to configure the switch to use truncated mode:

```
Console> (enable) set system switchmode allow truncated

Console> (enable)
```

Redundancy

No configuration is required for the Switch Fabric Module redundancy. The module in slot 5 functions as the primary module, and a redundant Switch Fabric Module in slot 6 automatically takes over if the primary module fails. The Catalyst 6506 and 6509 switches support a mixed redundant configuration with a WS-C6500-SFM and a WS-X6500-SFM 2. The Catalyst 6513 switch supports a redundant configuration with the WS-C6500-SFM2s only.

No configuration is required for the integrated 720-Gbps switch fabric redundancy. The integrated switch fabric in the active Supervisor Engine 720 functions as the primary switch fabric. A supervisor engine switchover will also cause a switch fabric switchover.

Monitoring the Integrated Switch Fabric and Switch Fabric Module

This section describes how to monitor the integrated switch fabric and the Switch Fabric Module:

Note Enter all the **show** commands that are supported by the integrated 720-Gbps switch fabric and the Switch Fabric Modules from the supervisor engine.

Displaying the Module Information

To display the module information, perform this task:

Task	Command
Display the module information.	**show module** *mod*[1]

1. The **show module** command is not supported for the integrated switch fabric.

This example shows how to display the module information:

```
Console> show module
Mod Slot Ports Module-Type               Model               Sub Status
--- ---- ----- ------------------------- ------------------- --- --------
1   1    2     1000BaseX Supervisor      WS-X6K-SUP2-2GE     yes ok
4   4    24    100BaseFX MM Ethernet     WS-X6224-MM-MT      no  ok
5   5    0     Switch Fabric Module      WS-C6500-SFM        no  ok

Mod Module-Name         Serial-Num
--- ------------------- -----------
1                       Munish
4                       SAD02390156
5                       SAD042818BR

Mod MAC-Address(es)                         Hw     Fw          Sw
--- --------------------------------------- ------ ----------- ------------------
1   00-40-0b-ff-00-00 to 00-40-0b-ff-00-01 0.219  6.1(0.146) 6.2(0.33-Eng)KEY
    00-50-3e-7e-71-56 to 00-50-3e-7e-71-57
    00-01-64-f8-ca-00 to 00-01-64-f8-cd-ff
4   00-10-7b-c2-3a-c0 to 00-10-7b-c2-3a-d7 0.204  4.2(0.24)V 6.2(0.14)KEY
5   00-40-0b-ff-00-00                      0.204  6.1(0.133) 6.2(0.14)KEY

Mod Sub-Type                Sub-Model           Sub-Serial  Sub-Hw
--- ----------------------- ------------------- ----------- ------
1   L3 Switching Engine II  WS-F6K-PFC2         SAD04110B5S 0.305
Console> (enable)
```

Displaying the Fabric Channel Counters

To display the fabric channel counters, perform this task:

Task	Command
Display the fabric channel counters.	**show fabric channel counters** {*mod* | **all**} [**hex**]

This example shows how to display the fabric channel counters:

```
Console> show fabric channel counters 5
Channel 0 counters:
0  rxTotalPkts                      =                    0
1  txTotalPkts                      =                    0
2  rxGoodPkts                       =                    0
3  rxErrors                         =                    0
4  txErrors                         =                    0
5  txDropped                        =                    0
```

Displaying the Fabric Channel Switching Mode and Channel Status

To display the fabric channel switching mode and channel status, perform this task:

Task	Command
Display the fabric channel switching mode and channel status.	**show fabric channel switchmode** [*mod*]

This example shows how to display the fabric channel switching mode and channel status:

```
Console> show fabric channel switchmode
Global switching mode:truncated

Module Num Fab Chan Fab Chan Switch Mode  Channel Status
------ ------------ -------- ------------ --------------
     1             1   0, 0   flow through ok
     4             0 n/a      n/a          n/a
     5            18   0, 0   n/a          ok
     5            18   1, 1   n/a          unused
     5            18   2, 2   n/a          unused
     5            18   3, 3   n/a          unused
     5            18   4, 4   n/a          unused
     5            18   5, 5   n/a          unused
     5            18   6, 6   n/a          unused
     5            18   7, 7   n/a          unused
     5            18   8, 8   n/a          unused
     5            18   9, 9   n/a          unused
     5            18  10, 10  n/a          unused
     5            18  11, 11  n/a          unused
     5            18  12, 12  n/a          unused
     5            18  13, 13  n/a          unused
     5            18  14, 14  n/a          unused
     5            18  15, 15  n/a          unused
     5            18  16, 16  n/a          unused
     5            18  17, 17  n/a          unused
```

In the **show fabric channel switchmode** command output, the Switch Mode field displays one of the following modes:

- Flow-through mode
- Truncated mode
- Compact mode

Note For definitions for the different modes, see the "Understanding How the External Switch Fabric Module Works" section on page 46-2.

Displaying the Fabric Channel Utilization

To display the fabric channel utilization, perform this task:

Task	Command
Display the fabric channel utilization.	**show fabric channel utilization**

This example shows how to display the fabric channel utilization:

```
Console> show fabric channel utilization
Fab Chan Input Output
-------- ----- ------
Fab Chan Speed Input Output
-------- ----- ----- ------
       0   n/a    0%     0%
       1   n/a    0%     0%
       2   n/a    0%     0%
       3   n/a    0%     0%
       4   20G    0%     0%
       5   n/a    0%     0%
       6   n/a    0%     0%
       7   20G    0%     0%
       8    8G    0%     0%
       9   n/a    0%     0%
      10   n/a    0%     0%
      11   n/a    0%     0%
      12   n/a    0%     0%
      13   n/a    0%     0%
      14   n/a    0%     0%
      15   n/a    0%     0%
      16   20G    0%     0%
      17   n/a    0%     0%
```

Displaying the Fabric Errors

To display the fabric errors of one or all modules, perform this task:

Task	Command
Display the fabric errors.	**show fabric errors** {*mod* | **all**}

This example shows how to display the fabric errors:

```
Console> (enable) show fabric errors all
Module errors:
Slot  Channel    CRC        Hbeat      Sync       DDR sync
----- ---------- ---------- ---------- ---------- ----------
2     0          0          0          0          0

Fabric errors:
Slot  Channel    Sync       Buffer     Timeout
----- ---------- ---------- ---------- ----------
2     0          0          0          0

Console> (enable)
```

Displaying the Backplane Traffic and Fabric Channel Input and Output

To display the backplane traffic and fabric channel input and output, perform either of these tasks:

Task	Command
Display the system status including the backplane traffic and fabric channel input and output.	**show system**
Display the backplane traffic and fabric channel input and output.	**show traffic**

Note Supervisor Engine 720 with PFC3A does not support a hardware traffic meter. When you enter the **show system** and **show traffic** commands on this module, you do not receive backplane traffic information.

This example shows how to display the system status including backplane traffic and fabric channel input and output:

```
Console> (enable) show system
PS1-Status PS2-Status
---------- ----------
ok         none

Fan-Status Temp-Alarm Sys-Status Uptime d,h:m:s Logout
---------- ---------- ---------- -------------- ---------
ok         off        ok         13,19:01:16    20 min

PS1-Type             PS2-Type
-------------------- --------------------
WS-CAC-1300W         none

Modem   Baud  Backplane-Traffic Peak Peak-Time
------- ----- ----------------- ---- -------------------------
disable  9600   0%                0% Tue Oct 19 2004, 12:04:18

PS1 Capacity: 1153.32 Watts (27.46 Amps @42V)

System Name               System Location          System Contact           CC
------------------------ ------------------------ ------------------------ ---


Slot Channel Fab Chan Input Output
---- ------- -------- ----- ------
   2       0        1    0%     0%

Core Dump                Core File
------------------------ ------------------------------------------
disabled                 slot0:crashdump

Crash Info               Crash Info File
------------------------ ------------------------------------------
enabled                  bootflash:crashinfo

System Information Logging  Host              Interval
--------------------------  ----------------  --------
Disabled                    -                 1440

System Information Log File
------------------------------------------------------------------------
```

```
tftp:sysinfo

Index        System Information Logging Commands
-----        -----------------------------------------------------------

Syslog Dump              Syslog File
------------------------ -----------------------------------------
disabled                 bootflash:sysloginfo

No profile is configured for the system
Console> (enable)
```

This example shows how to display backplane traffic and fabric channel input and output:

```
Console> (enable) show traffic
Threshold: 100%

Backplane-Traffic Peak Peak-Time
----------------- ---- -------------------------
  0%                0% Tue Oct 19 2004, 12:04:18

Slot Channel Fab Chan Input Output
---- ------- -------- ----- ------
   2       0        1    0%     0%

Console> (enable)
```

Displaying the Switching Mode Configuration

To display the switching mode configuration, perform this task:

Task	Command
Display the switching mode configuration.	**show system switchmode**

This example shows how to display the switching mode configuration:

```
Console> show system switchmode
Switchmode allow:truncated
Switchmode threshold:2
Console> (enable)
```

Displaying the Integrated Switch Fabric Status

To display the integrated switch fabric status and forwarding speed, perform this task:

Task	Command
Display the integrated switch fabric status and speed.	**show fabric status**

This example shows how to display the integrated switch fabric status and speed:

```
Console> show fabric status
Mod Speed  Fabric
           status
--- ----- -------
  5   20G  active
Console> (enable)
```

Configuring the LCD Banner

You can modify the LCD banner from the supervisor engine by entering the **set banner lcd** command to include the following information:

- Chassis serial number
- Switch IP address
- System name
- Supervisor engine version
- Multilayer Switch Feature Card (MSFC) version on active and standby supervisor engine
- System contact

After the LCD banner content is modified, this information is sent to the Switch Fabric Modules that are installed in the chassis and displayed in the LCDs.

Note The **set banner lcd** command is not supported in the systems with an integrated switch fabric.

To modify the LCD banner content, perform this task in privileged mode:

	Task	Command
Step 1	Modify the LCD banner content.	**set banner lcd** *c* [*text*] *c*
Step 2	Verify the LCD banner change.	**show banner**

This example shows how to modify the LCD banner for the Switch Fabric Module:

```
Console> (enable) set banner lcd &HelloWorld!&
LCD banner set
Console> (enable) show banner
MOTD banner:

LCD config:
Hello
World!
```

CHAPTER 47

Configuring SNMP

This chapter describes how to configure the Simple Network Management Protocol (SNMP) on the Catalyst 6500 series switches.

This chapter consists of these sections:

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

SNMP Terminology

Table 47-1 lists the terms that are used in the SNMP technology.

Table 47-1 SNMP Terminology

Term	Definition
authentication	The process of ensuring message integrity and protection against message replays, including both data integrity and data origin authentication.
authoritative SNMP engine	One of the SNMP copies involved in network communication is designated the allowed SNMP engine to protect against message replay, delay, and redirection. The security keys that are used for authenticating and encrypting the SNMPv3 packets are generated as a function of the authoritative SNMP engine's ID and user passwords. When an SNMP message expects a response (for example, get exact, get next, set request), the *receiver* of these messages is authoritative. When an SNMP message does not expect a response, the *sender* is authoritative.
community string	A text string that is used to authenticate messages between a management station and an SNMPv1 or SNMPv2c engine.
data integrity	A condition or state of data in which a message packet has not been altered or destroyed in an unauthorized manner.
data origin authentication	The ability to verify the identity of a user on whose behalf that the message is supposedly sent. This ability protects the users against both message capture and replay by a different SNMP engine and against the packets that are received or sent to a particular user that uses an incorrect password or security level.
encryption	A method of hiding data from an unauthorized user by scrambling the contents of an SNMP packet.
group	A set of users belonging to a particular security model. A group defines the access rights for all the users belonging to it. Access rights define the SNMP objects that can be read, written to, or created. In addition, the group defines the notifications that a user is allowed to receive.
notification host	An SNMP entity to which notifications (traps and informs) are to be sent.
notify view	A view name (not to exceed 64 characters) for each group; the view name defines the list of notifications that can be sent to each user in the group.
privacy	An encrypted state of the contents of an SNMP packet; in this state, the contents are prevented from being disclosed on a network. Encryption is performed with an algorithm called CBC-DES (DES-56).
read view	A view name (not to exceed 64 characters) for each group; the view name defines the list of object identifiers (OIDs) that can be read by users belonging to the group.

Table 47-1 SNMP Terminology (continued)

Term	Definition
security level	A type of security algorithm that is performed on each SNMP packet. There are three levels: noauth, auth, and priv. The noauth level authenticates a packet by a string match of the username. The auth level authenticates a packet by using either the HMAC MD5 or SHA algorithms. The priv level authenticates a packet by using either the HMAC MD5 or SHA algorithms and encrypts the packet using the CBC-DES (DES-56) algorithm.
security model	The security strategy that is used by the SNMP agent. Currently, Cisco IOS software supports three security models: SNMPv1, SNMPv2c, and SNMPv3.
Simple Network Management Protocol (SNMP)	A network management protocol that provides a method to monitor and control network devices and to manage configurations, statistics collection, performance, and security.
Simple Network Management Protocol Version 2c (SNMPv2c)	Second version of SNMP. This protocol supports centralized and distributed network management strategies and includes improvements in the structure of management information (SMI), protocol operations, management architecture, and security.
SNMP engine	A copy of SNMP that can reside on the local or remote device.
SNMP entity	Unlike SNMPv1 and SNMPv2c, in SNMPv3 the terms SNMP Agents and SNMP Managers are no longer used. These concepts have been combined and are called an SNMP entity. An SNMP entity is made up of an SNMP engine and SNMP applications.
SNMP group	A collection of SNMP users that belong to a common SNMP list that defines an access policy, in which object identification numbers (OIDs) are both read-accessible and write-accessible. The users belonging to a particular SNMP group inherit all of these attributes that are defined by the group.
SNMP user	A person for which an SNMP management operation is performed. The user is the person on a remote SNMP engine who receives the inform messages.
SNMP view	A mapping between the SNMP objects and the access rights that are available for those objects. An object can have different access rights in each view. The access rights indicate whether the object is accessible by either a community string or a user.
write view	A view name (not to exceed 64 characters) for each group; the view name defines the list of object identifiers (OIDs) that can be created or modified by the users of the group.

Understanding How SNMP Works

SNMP is an application-layer protocol that facilitates the exchange of management information between the network devices. SNMP enables the network administrators to manage network performance, find and solve network problems, and plan for network growth.

There are three versions of SNMP:

- Version 1 (SNMPv1)—This version is the initial implementation of SNMP. Refer to RFC 1157 for a full description of functionality. See the "Understanding How SNMPv1 and SNMPv2c Work" section on page 47-5 for more information on SNMPv1.
- Version 2 (SNMPv2c)—The second release of SNMP, described in RFC 1902, has additions and enhancements to data types, counter size, and protocol operations. See the "Understanding How SNMPv1 and SNMPv2c Work" section on page 47-5 for more information on SNMPv2.
- Version 3 (SNMPv3)—This version is the most recent version of SNMP and is fully described in RFC 2571, RFC 2572, RFC 2573, RFC 2574, and RFC 2575. The SNMP functionality on the Catalyst enterprise LAN switches for SNMPv1 and SNMPv2c remain intact; however, SNMPv3 has significant enhancements to administration and security. See the "Understanding How SNMPv3 Works" section on page 47-7 for more information on SNMPv3.

Security Models and Levels

A security model is an authentication strategy that is set up for a user and the group in which the user resides. A security level is the permitted level of security within a security model. A combination of a security model and a security level determines which security mechanism is employed when handling an SNMP packet. Three security models are available: SNMPv1, SNMPv2c, and SNMPv3. Table 47-2 identifies the combinations of security models and defines the levels for SNMPv1, SNMPv2c, and SNMPv3.

Table 47-2 SNMP Security Levels

Model	Level	Authentication	Encryption	What Happens
v1	noAuthNoPriv	Community String	No	Uses a community string match for authentication.
v2c	noAuthNoPriv	Community String	No	Uses a community string match for authentication.
v3	noAuthNoPriv	Username	No	Uses a username match for authentication.
v3	authNoPriv	MD5 or SHA	No	Provides authentication that is based on the HMAC-MD5 or HMAC-SHA algorithms.
v3	authPriv	MD5 or SHA	DES	Provides authentication that is based on the HMAC-MD5 or HMAC-SHA algorithms. Provides DES 56-bit encryption in addition to authentication that is based on the CBC-DES (DES-56) standard.

Note the following about the SNMPv3 objects:

- Each user belongs to a group.
- A group defines the access policy for a set of users.
- SNMP objects access an access policy for reading, writing, and creating.
- A group determines the list of notifications that its users can receive.
- A group also defines the security model and security level for its users.

SNMP ifindex Persistence

The SNMP ifIndex persistence feature is always enabled. With ifIndex persistence, the ifIndex value of the port and VLAN is always retained and used after these occurrences:

- Switch reboot
- High-availability switchover
- Software upgrade
- Module reset
- Module removal and insertion of the same type of module

For Fast EtherChannel and Gigabit EtherChannel interfaces, the ifIndex value is only retained and used after a high-availability switchover.

Understanding How SNMPv1 and SNMPv2c Work

The components of SNMPv1 and SNMPv2c network management fall into three categories:

- Managed devices (such as a switch)
- SNMP agents and MIBs, including the Remote Monitoring (RMON) MIBs, which run on managed devices
- SNMP network management applications, such as CiscoWorks2000, which communicate with the agents to get the statistics and the alerts from the managed devices. See the "Using CiscoWorks2000" section on page 47-6 for more information on CiscoWorks2000.

Note An SNMP management application, together with the computer it runs on, is called a Network Management System (NMS).

Using Managed Devices

Catalyst 6500 series switches are managed devices that support the SNMP network management with these features:

- SNMP traps (see the "Configuring SNMPv1 and SNMPv2c from the CLI" section on page 47-11)
- RMON in the supervisor engine software (see Chapter 48, "Configuring RMON")
- RMON and RMON2 on an external SwitchProbe device

Using the SNMP Agents and MIBs

SNMP network management uses these SNMP agent functions:

- Accessing a MIB variable—This function is initiated by the SNMP agent in response to a request from the NMS. The agent retrieves the value of the requested MIB variable and responds to the NMS with that value.
- Setting a MIB variable—This function is also initiated by the SNMP agent in response to a message from the NMS. The SNMP agent changes the value of the MIB variable to the value that is requested by the NMS.

Note For more information about MIBs, refer to http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml.

- SNMP trap—This function is used to notify an NMS that a significant event has occurred at an agent. When a trap condition occurs, the SNMP agent sends an SNMP trap message to any NMSs that are specified as the trap receivers under the following conditions:
  - When a port or module goes up or down
  - When the temperature limitations are exceeded
  - When there are spanning-tree topology changes
  - When there are authentication failures
  - When power supply errors occur
- SNMP community strings—SNMP community strings authenticate access to the MIB objects and function as embedded passwords:
  - Read-only—Gives read access to all objects in the MIB except the community strings but does not allow write access
  - Read-write—Gives read and write access to all objects in the MIB but does not allow access to the community strings
  - Read-write-all—Gives read and write access to all objects in the MIB including the community strings

Note The community string definitions on your NMS must match at least one of the three community string definitions on the switch.

Using CiscoWorks2000

CiscoWorks2000 is a family of Web-based and management platform-independent products for managing Cisco enterprise networks and devices. CiscoWorks2000 includes Resource Manager Essentials and CWSI Campus, which allow you to deploy, configure, monitor, manage, and troubleshoot a switched internetwork. For more information, refer to the following publications:

- *Getting Started With Resource Manager Essentials*
- *Getting Started With CWSI Campus*

Understanding How SNMPv3 Works

SNMPv3 contains all the functionality of SNMPv1 and SNMPv2c, but SNMPv3 has significant enhancements to administration and security. SNMPv3 is an interoperable standards-based protocol that provides secure access to the devices by authenticating and encrypting the packets over the network. The security features that are provided in SNMPv3 are as follows:

- Message integrity—Collects data securely without being tampered with or corrupted
- Authentication—Determines that the message is from a valid source
- Encryption—Scrambles the contents of a packet to prevent it from being seen by an unauthorized source

SNMP Entity

Unlike SNMPv1 and SNMPv2c, in SNMPv3 the concept of *SNMP Agents* and *SNMP Managers* no longer apply. These concepts have been combined into an *SNMP entity*. An SNMP entity consists of an SNMP engine and SNMP applications. An SNMP engine consists of the following four components:

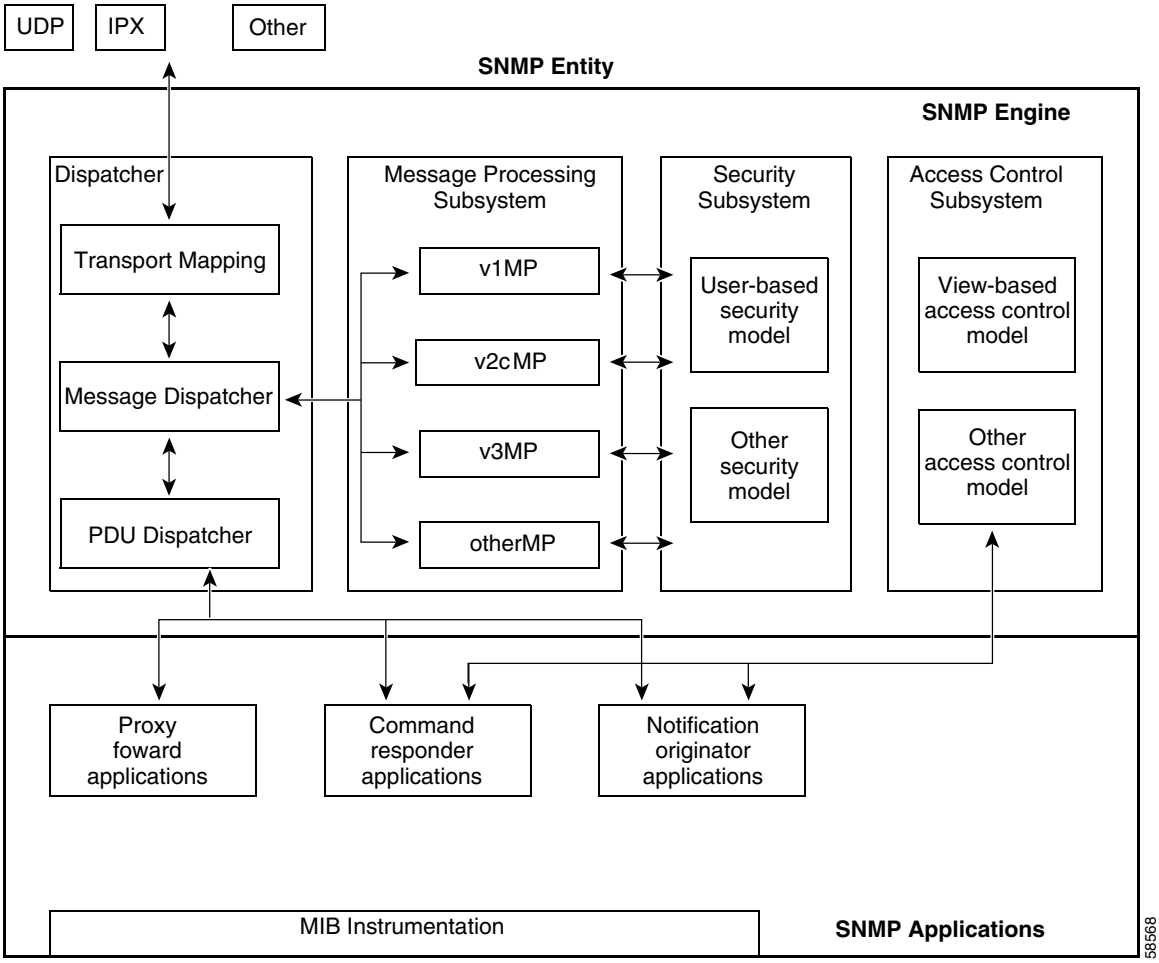
- Dispatcher
- Message processing subsystem
- Security subsystem
- Access control subsystem

Figure 47-1 shows an SNMP entity.

Dispatcher

The dispatcher is a traffic manager that sends and receives the messages. After receiving a message, the dispatcher tries to determine the version number of the message and then passes the message to the appropriate message processing model. The dispatcher is also responsible for dispatching the protocol data units (PDUs) to the applications and for selecting the appropriate transports for sending the messages.

Figure 47-1 SNMP Entity for Traditional SNMP Agents



Message Processing Subsystem

The message processing subsystem accepts the outgoing PDUs from the dispatcher and prepares them for transmission by wrapping them in a message header and returning them to the dispatcher. The message processing subsystem also accepts the incoming messages from the dispatcher, processes each message header, and returns the enclosed PDU to the dispatcher. An implementation of the message processing subsystem may support a single message format corresponding to a single version of SNMP (SNMPv1, SNMPv2c, SNMPv3), or it may contain a number of modules, each supporting a different version of SNMP.

Security Subsystem

The security subsystem authenticates and encrypts the messages. Each outgoing message is passed to the security subsystem from the message processing subsystem. Depending on the services required, the security subsystem may encrypt the enclosed PDU and some fields in the message header. In addition, the security subsystem may generate an authentication code and insert it into the message header. After encryption, the message is returned to the message processing subsystem.

Each incoming message is passed to the security subsystem from the message processing subsystem. If required, the security subsystem checks the authentication code and performs the decryption. The processed message is returned to the message processing subsystem. An implementation of the security subsystem may support one or more distinct security models. The only currently defined security model is the user-based security model (USM) for SNMPv3, which is specified in RFC 2274.

The USM protects the SNMPv3 messages from the following potential security threats:

- An authorized user sending a message that gets modified in transit by an unauthorized SNMP entity.
- An unauthorized user trying to masquerade as an authorized user.
- A user modifying the message stream.
- An unauthorized user listening to the message.

The USM currently defines the HMAC-MD5-96 and HMAC-SHA-96 as the authentication protocols and CBC-DES as the privacy protocol.

SNMPv1 and SNMPv2c security models provide only the community names for authentication and no privacy.

Access Control Subsystem

The access control subsystem determines whether access to a managed object should be allowed. With the view-based access control model (VACM), you can control which users and which operations can have access to which managed objects.

Applications

The SNMPv3 applications refer to the internal applications within an SNMP entity. These internal applications can do the following operations:

- Generate the SNMP messages
- Respond to the received SNMP messages
- Generate and receive the notifications
- Forward the messages between the SNMP entities

There are currently five types of applications:

- Command generators—Generate the SNMP commands to collect or set management data.
- Command responders—Provide access to the management data. For example, **processing get**, **get-next**, **get-bulk**, and **set pdus** are used in a command responder application.
- Notification originators—Initiate the Trap or Inform messages.
- Notification receivers—Receive and process the Trap or Inform messages.
- Proxy forwarders—Forward the messages between the SNMP entities.

Enabling and Disabling SNMP Processing

This section describes how to use the **set snmp enable | disable** command to enable or disable the processing of the SNMP requests to the switch and the SNMP traps from the switch.

If you set SNMP to enable mode, the SNMP requests to the switch are processed and the SNMP traps are sent out if there is no conflict with the other SNMP configurations on the switch.

If you set SNMP to disable mode, the SNMP requests are ignored and no SNMP traps are sent out independent of the other SNMP configurations on the switch.

In either SNMP mode (enabled or disabled), you can change the other SNMP configurations. The RMON-related processes are not affected in either mode.

To enable SNMP processing from the command-line interface (CLI), perform this task in privileged mode (enable mode is the default):

	Task	Command
Step 1	Enable SNMP processing.	**set snmp enable | disable**
Step 2	Verify that SNMP processing is enabled.	**show snmp**

This example shows how to enable SNMP processing:

```
Console> (enable) set snmp enable
SNMP enabled.
Console> (enable)
```

This example shows how to disable SNMP processing:

```
Console> (enable) set snmp disable
SNMP disabled.
Console> (enable)
```

This example shows how to verify the SNMP configuration:

```
Console> (enable) show snmp
SNMP:                       Disabled
RMON:                       Disabled
Extended RMON Netflow Enabled : None.
Memory usage limit for new RMON entries: 85 percent
Traps Enabled:
None
Port Traps Enabled: None

Community-Access     Community-String
----------------     --------------------
read-only            public
read-write           private
read-write-all       secret

Trap-Rec-Address Trap-Rec-Community Trap-Rec-Port Trap-Rec-Owner Trap-Rec-Index
---------------- ------------------ ------------- -------------- --------------
Console> (enable)
```

Configuring SNMPv1 and SNMPv2c on the Switch

This section provides the basic SNMPv1 and SNMPv2c configuration information. For detailed information on the SNMP commands that are supported by the Catalyst 6500 series switches, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

SNMPv1 and SNMPv2c Default Configuration

Refer to the *Catalyst 6500 Series Switch Command Reference* for the SNMP default configuration settings for each command that is listed in the configuration section.

Configuring SNMPv1 and SNMPv2c from an NMS

To configure SNMP from an NMS, refer to the NMS documentation (see the "Using CiscoWorks2000" section on page 47-6).

The switch supports up to 20 trap receivers through the RMON2 trap destination table. You configure the RMON2 trap destination table from the NMS.

Configuring SNMPv1 and SNMPv2c from the CLI

Note For the enhanced SNMP features in software release 7.5(1), see the "SNMPv1 and SNMPv2c Enhancements in Software Release 7.5(1)" section on page 47-12.

To configure SNMP from the CLI, perform this task in privileged mode:

	Task	Command
Step 1	Define the SNMP community strings for each access type.	**set snmp community read-only** *community_string* **set snmp community read-write** *community_string* **set snmp community read-write-all** *community_string*
Step 2	Assign a trap receiver and community. You can specify up to ten trap receivers.	**set snmp trap** *rcvr_address rcvr_community*
Step 3	Specify the SNMP traps to send to the trap receiver.	**set snmp trap enable** [**all** | **auth** | **bridge** | **chassis** | **config** | **entity** | **entityfru** | **envfan** | **envpower** | **envshutdown** | **envtemp** | **flashinsert** | **flashremove** | **ippermit** | **module** | **stpx** | **syslog** | **system** | **vlancreate** | **vlandelete** | **vmps** | **vtp**]
Step 4	Verify the SNMP configuration.	**show snmp**

This example shows how to define the community strings, assign a trap receiver, and specify which traps to send to the trap receiver:

```
Console> (enable) set snmp community read-only Everyone
SNMP read-only community string set to 'Everyone'.
Console> (enable) set snmp community read-write Administrators
SNMP read-write community string set to 'Administrators'.
Console> (enable) set snmp community read-write-all Root
SNMP read-write-all community string set to 'Root'.
Console> (enable) set snmp trap 172.16.10.10 read-write
SNMP trap receiver added.
Console> (enable) set snmp trap 172.16.10.20 read-write-all
SNMP trap receiver added.
Console> (enable) set snmp trap enable all
All SNMP traps enabled.
Console> (enable) show snmp
RMON:                          Disabled
Extended RMON:                 Extended RMON module is not present
Traps Enabled:
Port,Module,Chassis,Bridge,Repeater,Vtp,Auth,ippermit,Vmps,config,entity,stpx
Port Traps Enabled: 1/1-2,4/1-48,5/1
Community-Access     Community-String
----------------     --------------------
read-only            Everyone
read-write           Administrators
read-write-all       Root
Trap-Rec-Address                              Trap-Rec-Community
--------------------------------------------  --------------------
172.16.10.10                                  read-write
172.16.10.20                                  read-write-all
Console> (enable)
```

Note To disable access for an SNMP community, set the community string for that community to the null string (do not enter a value for the community string).

SNMPv1 and SNMPv2c Enhancements in Software Release 7.5(1)

These sections describe the enhancements that have been added to software release 7.5(1):

Setting Multiple SNMP Community Strings

You can set multiple SNMP community strings using the **community-ext** keyword. The community strings that are defined using the **community-ext** keyword cannot be duplicates of the existing community strings. When you add a new community string using the **community-ext** keyword, the appropriate entries are created in the vacmAccessTable (if a view is specified), snmpCommunityTable, and the vacmSecurityToGroup table.

To set multiple SNMP community strings from the CLI, perform this task in privileged mode:

	Task	Command
Step 1	Set multiple SNMP community strings.	**set snmp community-ext** *community_string* {**read-only** | **read-write** | **read-write-all**} [**view** *view_oid*] [**access** *access_number*]
Step 2	Verify the SNMP configuration.	**show snmp**

This example shows how to set an additional SNMP community string:

```
Console> (enable) set snmp community-ext public1 read-only

Community string public1 is created with access type as read-only
Console> (enable)
```

This example shows how to restrict the community string to an access number:

```
Console> (enable) set snmp community-ext private1 read-write access 2

Community string private1 is created with access type as read-write access number 2
Console> (enable)
```

This example shows how to change the access number to the community string:

```
Console> (enable) set snmp community-ext private1 read-write access 3

Community string private1 is updated with access type as read-write access
number 3
Console> (enable)
```

This example shows how to display the SNMP configuration:

```
Console> (enable) show snmp

SNMP:Enabled
RMON:Disabled
Extended RMON Netflow Enabled :None.
Memory usage limit for new RMON entries:85 percent
Traps Enabled:None
Port Traps Enabled:None

Community-Access Community-String
---------------- --------------------
read-only        public
read-write       private
read-write-all   secret
```

```
Additional-                          Access-
Community-String     Access-Type     Number  View
-------------------- --------------- ------- ----------------------------------
public1              read-only
public2              read-only       1
private1             read-write      2       1.3.6
secret1              read-write-all  500     1.3.6.1.4.1.9.9

Trap-Rec-Address Trap-Rec-Community Trap-Rec-Port Trap-Rec-Owner Trap-Rec-Index
--------------- ------------------ ------------- -------------- --------------
Console> (enable)
```

Clearing the SNMP Community Strings

You can clear the community strings using the **clear snmp community-ext** *community-string* command. When you use this command to clear a community string, the corresponding entries in the vacmAccessTable and vacmSecurityToGroup tables are also removed.

To clear an SNMP community string from the CLI, perform this task in privileged mode:

	Task	Command
Step 1	Clear an SNMP community string.	**clear snmp community-ext** *community-string*
Step 2	Verify the SNMP configuration.	**show snmp**

This example shows how to clear an SNMP community string:

```
Console> (enable) clear snmp community-ext public1
Community string public1 has been removed
Console> (enable)
```

Specifying the Access Numbers for Hosts

You can specify a list of access numbers that are associated with one or more hosts to limit which hosts can use a specific community string to access the system. You can specify more than one IP address that is associated with an access number by separating each IP address with a space. If an existing access number is used, the new IP addresses are appended to the list.

To specify an access number for a host from the CLI, perform this task in privileged mode:

	Task	Command
Step 1	Specify an access number for a host.	**set snmp access-list** *access_number IP_address* [**ipmask** *maskaddr*]
Step 2	Verify the SNMP configuration.	**show snmp access-list**

These examples show how to specify an access number for a host:

```
Console> (enable) set snmp access-list 1 172.20.60.100
Access number 1 has been created with new IP Address 172.20.60.100

Console> (enable) set snmp access-list 2 172.20.60.100 mask 255.0.0.0
Access number 2 has been created with new IP Address 172.20.60.100 mask 255.0.0.0
```

```
Console> (enable) set snmp access-list 2 172.20.60.7
Access number 2 has been updated with new IP Address 172.20.60.7

Console> (enable) set snmp access-list 2 172.20.60.7 mask 255.255.255.0
Access number 2 has been updated with existing IP Address 172.20.60.7 mask 255.255.255.0
Console> (enable)
```

This example shows how to display the SNMP configuration:

```
Console> (enable) show snmp access-list
Access-Number   IP-Addresses/IP-Mask
-------------   ------------------------
1               172.20.60.100/255.0.0.0
                1.1.1.1/-
2               172.20.60.7/-
                2.2.2.2/-
3               2.2.2.2/155.0.0.0
4               1.1.1.1/2.1.2.4
                2.2.2.2/-
                2.2.2.5/-
Console> (enable)
```

Clearing the IP Addresses Associated with Access Numbers

To clear the IP addresses that are associated with the access numbers from the CLI, perform this task in privileged mode:

	Task	Command
Step 1	Clear the IP addresses that are associated with the access numbers.	**clear snmp access-list** *access_number IP_address* [[*IP_address*] ...]
Step 2	Verify the SNMP configuration.	**show snmp access-list**

These examples show how to clear the IP addresses that are associated with the access numbers:

```
Console> (enable) clear snmp access-list 101
All IP addresses associated with access-number 101 have been cleared.
Console> (enable)

Console> (enable) clear snmp access-list 2 172.20.60.8
Access number 2 no longer associated with 172.20.60.8
Console> (enable)
```

Specifying, Displaying, and Clearing an Interface Alias

You can specify, display, and clear an interface alias. The length of the alias can be up to 64 characters.

Note The **set snmp ifalias** command cannot be used in binary configuration mode. You must use the text file configuration mode when entering this command or the interface alias is not saved in NVRAM.

To specify, display, and clear an interface alias, perform this task in privileged mode:

	Task	Command
Step 1	Specify an interface alias.	**set snmp ifalias** {*ifIndex*} [*ifAlias*]
Step 2	Display the interface alias.	**show snmp ifalias** [*ifIndex*]
Step 3	Clear the interface alias.	**clear snmp ifalias** {*ifIndex*} **all**

These examples show how to specify, display, and clear an interface alias:

```
Console> (enable) set snmp ifalias 1 Inband port

ifIndex 1 alias set
Console> (enable)

Console> (enable) show snmp ifalias 1
ifIndex    ifName               ifAlias
---------- -------------------- --------------------------------
1          sc0                  Inband port
Console> (enable)

Console> (enable) clear snmp ifalias all
Console> (enable)
```

Configuring SNMPv3 on the Switch

This section provides the basic SNMPv3 configuration information. For detailed information on the SNMP commands that are supported by the Catalyst 6500 series switches, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

SNMPv3 Default Configuration

Refer to the *Catalyst 6500 Series Switch Command Reference* publication for the SNMP default configuration settings for each command that is listed in the configuration section.

Configuring SNMPv3 from an NMS

To configure SNMP from an NMS, refer to the NMS documentation (see the "Using CiscoWorks2000" section on page 47-6).

The switch supports up to 20 trap receivers through the RMON2 trap destination table. You configure the RMON2 trap destination table from the NMS.

Configuring SNMPv3 from the CLI

To configure SNMPv3 from the CLI, perform this task in privileged mode:

	Task	Command
Step 1	Set the SNMP-Server EngineID name for the local SNMP engine.	**set snmp engineid** *engineid*
Step 2	Configure the MIB views.	**set snmp view** [**-hex**] {*viewname*} {*subtree*} [*mask*] [**included** | **excluded**] [**volatile** | **nonvolatile**]
Step 3	Set the access rights for a group with a certain security model in the different security levels.	**set snmp access** [**-hex**] {*groupname*} {**security-model v3**} {**noauthentication** | **authentication** | **privacy**} [**read** [**-hex**] {*readview*}] [**write** [**-hex**] {*writeview*}] [**notify** [**-hex**] {*notifyview*}] [**context** [**-hex**] {*contextname*} [**exact** | **prefix**]] [**volatile** | **nonvolatile**]
Step 4	Specify the target addresses for the notifications.	**set snmp notify** [**-hex**] {**notifyname**} **tag** [**-hex**] {*notifytag*} [**trap** | **inform**] [**volatile** | **nonvolatile**]
Step 5	Set the snmpTargetAddrEntry in the target address table.	**set snmp targetaddr** [**-hex**] {*addrname*} **param** [**-hex**] {*paramsname*} {*ipaddr*} [**udpport** {*port*}] [**timeout** {**value**}] [**retries** {**value**}] [**volatile** | **nonvolatile**] [**taglist** {[**-hex**] *tag*} [[**-hex**] *tag*]]
Step 6	Set the SNMP parameters that are used to generate a message to a target.	**set snmp targetparams** [**-hex**] {*paramsname*} **user** [**-hex**] {*username*} {**security-model v3**} {**message-processing v3**} {**noauthentication** | **authentication** | **privacy**} [**volatile** | **nonvolatile**]
Step 7	Configure a new user.	**set snmp user** [**-hex**] {*username*} [**remote** {*engineid*}] [{**authentication** [**md5** | *sha*] {*authpassword*}] [**privacy** {*privpassword*}] [**volatile** | **nonvolatile**]
Step 8	Relate a user to a group using a specified security model.	**set snmp group** [**-hex**] {*groupname*} **user** [**-hex**] {*username*} {**security-model v1** | **v2** | **v3**} [**volatile** | **nonvolatile**]
Step 9	Configure the community table for the system default part, which maps the community strings of the previous versions of SNMP to SNMPv3.	**set snmp community** {**read-only** | **read-write** | **read-write-all**} [*community_string*]
Step 10	Configure the community table for the mappings between the different community strings and the security models with full permissions.	**set snmp community index** {*index_name*} **name** [*community_string*] **security** {*security_name*} **context** {*context_name*} **transporttag** {*tag_value*} [**volatile** | **nonvolatile**]
Step 11	Verify the SNMP configuration.	**show snmp**

This example shows how to set a MIB view to interfacesMibView:

```
Console> (enable) set snmp view interfacesMibView 1.3.6.1.2.1.2 included
Snmp view name was set to interfacesMibView with subtree 1.3.6.1.2.1.2 included,
nonvolatile.
```

This example shows how to set the access rights for a group called guestgroup to SNMPv3 authentication-read mode:

```
Console> (enable) set snmp access guestgroup security-model v3 authentication read
interfacesMibView
Snmp access group was set to guestgroup version v3 level authentication,
readview interfacesMibView, context match:exact, nonvolatile.
```

This example shows how to specify the target addresses:

```
Console> (enable) set snmp notify notifytable1 tag routers trap
Snmp notify name was set to notifytable1 with tag routers notifyType trap, and storageType
nonvolatile.
```

These examples show how to set the snmpTargetAddrEntry in the target address table:

```
Console> (enable) set snmp targetaddr router_1 param p1 172.20.21.1
Snmp targetaddr name was set to router_1 with param p1
ipAddr 172.20.21.1, udpport 162, timeout 1500, retries 3, storageType nonvolatile.

Console> (enable) set snmp targetaddr router_2 param p2 172.20.30.1
Snmp targetaddr name was set to router_2 with param p2
ipAddr 172.20.30.1, udpport 162, timeout 1500, retries 3, storageType nonvolatile.
```

These examples show how to set the SNMP target parameters:

```
Console> (enable) set snmp targetparams p1 user guestuser1 security-model v3
message-processing v3 authentication
Snmp target params was set to p1 v3 authentication, message-processing v3,
user guestuser1 nonvolatile.

Console> (enable) set snmp targetparams p2 user guestuser2 security-model v3
message-processing v3 privacy
Snmp target params was set to p2 v3 privacy, message-processing v3,
user guestuser2 nonvolatile.
```

These examples show how to configure guestuser1 and guestuser2 as users:

```
Console> (enable) set snmp user guestuser1 authentication md5 guestuser1password privacy
privacypasswd1
Snmp user was set to guestuser1 authProt md5 authPasswd guestuser1password privProt des
privPasswd
privacypasswd1 with engineid 00:00:00:09:00:10:7b:f2:82:00:00:00 nonvolatile.

Console> (enable) set snmp user guestuser2 authentication sha guestuser2password
Snmp user was set to guestuser2 authProt sha authPasswd guestuser2password privProt
no-priv with engineid
00:00:00:09:00:10:7b:f2:82:00:00:00 nonvolatile.
```

These examples show how to set guestuser1 and guestuser2 as members of the groups guestgroup and mygroup:

```
Console> (enable) set snmp group guestgroup user guestuser1 security-model v3
Snmp group was set to guestgroup user guestuser1 and version v3, nonvolatile.

Console> (enable) set snmp group mygroup user guestuser1 security-model v3
Snmp group was set to mygroup user guestuser1 and version v3, nonvolatile.

Console> (enable) set snmp group mygroup user guestuser2 security-model v3
Snmp group was set to mygroup user guestuser2 and version v3, nonvolatile.
```

This example shows how to verify the SNMPv3 setup for guestuser1 from a workstation:

```
workstation% getnext -v3 10.6.4.201 guestuser1 ifDescr.0
Enter Authentication password :guestuser1password
Enter Privacy password        :privacypasswd1
ifDescr.1 = sc0
```

This example shows how to verify the SNMPv3 setup for guestgroup in the snmpEngineID MIB from a workstation:

```
workstation% getnext -v3 10.6.4.201 guestuser1 snmpEngineID
Enter Authentication password :guestuser1pasword
Enter Privacy password        :privacypasswd1
snmpEngineID = END_OF_MIB_VIEW_EXCEPTION
```

This example shows how to verify the SNMPv2c setup for public access from a workstation:

```
workstation% getnext -v2c 10.6.4.201 public snmpEngineID
snmpEngineID.0 =
00 00  00 09   00 10  7b f2    82 00  00 00
```

These examples show how to increase guestgroup's access right to read privileges for snmpEngineMibView:

```
Console> (enable) set snmp view snmpEngineMibView 1.3.6.1.6.3.10.2.1 included
Snmp view name was set to snmpEngineMibView with subtree 1.3.6.1.6.3.10.2.1 included,
nonvolatile

Console> (enable) set snmp access guestgroup security-model v3 authentication read
snmpEngineMibView
Snmp access group was set to guestgroup version v3 level authentication,
readview snmpEngineMibView, nonvolatile.
```

This example shows how to verify the SNMPv3 access for guestuser1 from a workstation:

```
workstation% getnext -v3 10.6.4.201 guestuser1 snmpEngineID
Enter Authentication password :guestuser1password
Enter Privacy password        :privacypasswd1
snmpEngineID.0 =
00 00  00 09   00 10  7b f2    82 00  00 00
```

This example shows how to remove the access for guestgroup:

```
Console> (enable) clear snmp acc guestgroup security-model v3 authentication
Cleared snmp access guestgroup version v3 level authentication.
```

This example shows how to verify that the access for guestuser1 has been removed from a workstation:

```
workstation% getnext -v3 10.6.4.201 guestuser1 ifDescr.1
Enter Authentication password :guestuser1password
Enter Privacy password        :privacypasswd1
Error code set in packet - AUTHORIZATION_ERROR:1.
```

This example shows how to verify the access for guestuser2 from a workstation:

```
workstation% getnext -v3 10.6.4.201 guestuser2 ifDescr.1
Enter Authentication password :guestuser2password
Enter Privacy password        :privacypasswd2
REPORT received, cannot recover:
usmStatsUnsupportedSecLevels.0 = 1
```

CHAPTER 48

Configuring RMON

This chapter describes how to configure Remote Monitoring (RMON) on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Understanding How RMON Works

RMON is an Internet Engineering Task Force (IETF) standard monitoring specification that allows the various network agents and console systems to exchange network monitoring data. The supervisor engine software provides embedded support for these components of the RMON specification (see the "Supported RMON and RMON2 MIB Objects" section on page 48-3 for details):

- The following RMON groups are defined in RFC 1757:
  - Statistics (RMON group 1) for Ethernet, Fast Ethernet, Fast EtherChannel, and Gigabit Ethernet switch ports (uses 140 bytes of supervisor engine RAM per port)
  - History (RMON group 2) for Ethernet, Fast Ethernet, Fast EtherChannel, and Gigabit Ethernet switch ports (uses 3 KB of supervisor engine RAM for the first 50 buckets; each additional bucket uses another 56 bytes)
  - Alarm (RMON group 3; each alarm configured uses 1.3 KB of supervisor engine RAM)
  - Event (RMON group 9; each event configured uses 1.3 KB of supervisor engine RAM)
- The following RMON2 groups are defined in RFC 2021:
  - UsrHistory (RMON2 group 18)
  - ProbeConfig (RMON2 group 19)

The embedded RMON agent allows the switch to monitor network traffic from all ports simultaneously at Layer 2 without requiring a dedicated monitoring probe or network analyzer. For more information on RMON, visit:

http://www.cisco.com/en/US/docs/internetworking/technology/handbook/RMON.html

Enabling RMON on the Switch

Note RMON is disabled by default.

To enable RMON, perform this task in privileged mode:

	Task	Command
Step 1	Enable RMON on the switch.	**set snmp rmon enable**
Step 2	Verify that RMON is enabled.	**show snmp**

This example shows how to enable RMON on the switch and how to verify that RMON is enabled:

```
Console> (enable) set snmp rmon enable
SNMP RMON support enabled.
Console> (enable) show snmp
RMON:                          Enabled
Extended RMON:                 Extended RMON module is not present
Traps Enabled:
Port,Module,Chassis,Bridge,Repeater,Vtp,Auth,ippermit,Vmps,config,entity,stpx
Port Traps Enabled: 1/1-2,4/1-48,5/1
Community-Access     Community-String
----------------     --------------------
read-only            Everyone
read-write           Administrators
read-write-all       Root
Trap-Rec-Address                            Trap-Rec-Community
------------------------------------------  --------------------
172.16.10.10                                read-write
172.16.10.20                                read-write-all
Console> (enable)
```

Viewing the RMON Data

Access to the RMON data is available only on a network management system (NMS) that supports RFC 1757 and RFC 2021 (see the "Using CiscoWorks2000" section on page 47-6). You cannot access the RMON data through the switch CLI; however, the CLI **show** commands provide similar information.

Supported RMON and RMON2 MIB Objects

Table 48-1 lists the RMON and RMON2 MIB objects that are supported by the supervisor engine software.

Table 48-1 Supervisor Engine RMON and RMON2 Support

Object Identifier (OID) and Description	Source
...mib-2(1).rmon(16).statistics(1).etherStatsTable(1)	RFC 1757 (RMON-MIB)
Counters for packets, octets, broadcasts, errors, etc.	
...mib-2(1).rmon(16).history(2).historyControlTable(1) ...mib-2(1).rmon(16).history(2).etherHistoryTable(2)	RFC 1757 (RMON-MIB) RFC 1757 (RMON-MIB)
Periodically samples and saves statistics group counters for later retrieval.	
...mib-2(1).rmon(16).alarm(3)	RFC 1757 (RMON-MIB)
A threshold that can be set on critical RMON variables for network management.	
...mib-2(1).rmon(16).event(9)	RFC 1757 (RMON-MIB)
Generates SNMP traps when an Alarms group threshold is exceeded and logs the events.	
...mib-2(1).rmon(16).usrHistory(18)	RFC 2021 (RMON2-MIB)
Extends history beyond RMON1 link-layer statistics to include any RMON, RMON2, MIB-I, or MIB-II statistic.	
...mib-2(1).rmon(16).probeConfig(19)	RFC 2021 (RMON2-MIB)
Displays a list of agent capabilities and configurations.	

CHAPTER 49

Configuring SPAN, RSPAN and the Mini Protocol Analyzer

This chapter describes how to configure Switched Port Analyzer (SPAN), Remote SPAN (RSPAN), and the Mini Protocol Analyzer on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Note To configure SPAN, RSPAN or the Mini Protocol Analyzer from a network management station (NMS), refer to the NMS documentation (see the "Using CiscoWorks2000" section on page 47-6).

Understanding How SPAN and RSPAN Work

These sections describe the concepts and terminology that are associated with SPAN and RSPAN configuration:

SPAN Session

A SPAN session is an association of destination ports with a set of source ports, configured with the parameters that specify the monitored network traffic. You can configure multiple SPAN sessions in a switched network. The SPAN sessions do not interfere with the normal operation of the switches. You can enable or disable the SPAN sessions with the command-line interface (CLI) or SNMP commands. When enabled, a SPAN session might become active or inactive based on various events or actions, and this would be indicated by a syslog message. The "Status" field in the **show span** and **show rspan** commands displays the operational status of a SPAN or RSPAN session.

A SPAN or RSPAN destination session remains inactive after system power up until the destination ports are operational. An RSPAN source session remains inactive until any of the source ports are operational or the RSPAN VLAN becomes active.

Destination Port

A destination port (also called a *monitor port*) is an access port where SPAN sends the packets for analysis. After a port becomes an active destination port, it does not forward any traffic except that required for the SPAN session. By default, an active destination port disables the incoming traffic (from the network to the switching bus), unless you specifically enable the port. If the incoming traffic is enabled for the destination port, it is switched in the native VLAN of the destination port. The destination port does not participate in spanning tree while the SPAN session is active. See the caution statement in the "Configuring SPAN from the CLI" section on page 49-8 for information on how to prevent loops in your network topology.

Multiple destination ports can be specified in each local SPAN session but a destination port cannot be a destination port for multiple SPAN sessions. An access port that is configured as a destination port cannot be configured as a source port. EtherChannel ports cannot be SPAN destination ports.

If the trunking mode of a SPAN destination port is "on" or "nonegotiate" during the SPAN session configuration, the SPAN packets that are forwarded by the destination port have the encapsulation as specified by the trunk type; however, the destination port stops trunking, and the **show trunk** command reflects the trunking status for the port prior to the SPAN session configuration.

Source Port

A source port is an access port that is monitored for network traffic analysis. The traffic through the source ports can be categorized as ingress, egress, or both. You can monitor one or more source ports in a single SPAN session with the user-specified traffic types (ingress, egress, or both) applicable for all the source ports.

You can configure the source ports in any VLAN. You can configure the VLANs as the source ports (*src_vlans*), which means that all the ports in the specified VLANs are the source ports for the SPAN session.

The source ports are administrative (*Admin Source*), operational (*Oper Source*), or both. The administrative source ports are the source ports or the source VLANs that are specified during the SPAN session configuration. The operational source ports are the source ports that are monitored by the destination port. For example, when the source VLANs are used as the administrative source, the operational source is all the ports in all the specified VLANs.

The operational sources are always the active ports. If a port is not in the spanning tree, it is not an operational source. All physical ports in an EtherChannel source are included in the operational sources if the logical port is included in the spanning tree.

The destination port, if it belongs to any of the administrative source VLANs, is excluded from the operational source.

You can configure a port as a source port in multiple active SPAN sessions, but you cannot configure an active source port as a destination port for any SPAN session.

If a SPAN session is inactive, the “oper source” field is not updated until the session becomes active.

The trunk ports can be configured as the source ports and can be mixed with the nontrunk source ports; however, the encapsulation of the packets that are forwarded by the destination port are determined by the trunk settings of the destination port during the SPAN session configuration.

Ingress SPAN

Ingress SPAN copies the network traffic that is received by the source ports for analysis at the destination ports.

Egress SPAN

Egress SPAN copies the network traffic that is transmitted from the source ports for analysis at the destination ports.

VSPAN

VLAN-based SPAN (VSPAN) is analysis of the network traffic in one or more VLANs. You can configure VSPAN as ingress SPAN, egress SPAN, or both. All the ports in the source VLANs become the operational source ports for the VSPAN session. The destination ports, if they belong to any of the administrative source VLANs, are excluded from the operational source. If you add or remove the ports from the administrative source VLANs, the operational sources are modified accordingly.

Use the following guidelines for VSPAN sessions:

- The trunk ports are included as the source ports for the VSPAN sessions, but only the VLANs that are in the Admin source list are monitored if these VLANs are active for the trunk.
- For the VSPAN sessions with both ingress and egress SPAN configured, the system operates as follows based upon the type of supervisor engine that you have:
  - WS-X6K-SUP1A-PFC, WS-X6K-SUP1A-MSFC, WS-X6K-S1A-MSFC2, WS-X6K-S2-PFC2, WS-X6K-S1A-MSFC2, WS-SUP720, WS-SUP32-GE-3B, —Two packets are forwarded by the SPAN destination port if the packets get switched on the same VLAN.
  - WS-X6K-SUP1-2GE, WS-X6K-SUP1A-2GE—Only one packet is forwarded by the SPAN destination port.
- An inband port is not included as Oper source for the VSPAN sessions.

- When a VLAN is cleared, it is removed from the source list for the VSPAN sessions.
- A VSPAN session is disabled if the Admin source VLANs list is empty.
- The inactive VLANs are not allowed for the VSPAN configuration.
- A VSPAN session is made inactive if any of the source VLANs become the RSPAN VLANs.

Trunk VLAN Filtering

Trunk VLAN filtering is analysis of network traffic on a selected set of VLANs on the trunk source ports. You can combine trunk VLAN filtering with the other source ports that belong to any of the selected VLANs, and you can also use trunk VLAN filtering for RSPAN. Based on the traffic type (ingress, egress, or both), SPAN sends a copy of the network traffic in the selected VLANs to the destination ports.

Use trunk VLAN filtering only with the trunk source ports. If you combine trunk VLAN filtering with the other source ports that belong to the VLANs that are not included in the selected list of filter VLANs, SPAN includes only the ports that belong to one or more of the selected VLANs in the operational sources.

When a VLAN is cleared, it is removed from the VLAN filter list. A SPAN session is disabled if the VLAN filter list becomes empty.

Trunk VLAN filtering is not applicable to the VSPAN sessions.

SPAN Traffic

All network traffic, including the multicast and bridge protocol data unit (BPDU) packets, can be monitored using SPAN (RSPAN does not support monitoring of BPDU packets or Layer 2 protocol packets such as CDP, DTP, and VTP). Multicast packet monitoring is enabled by default.

In some SPAN configurations, multiple copies of the same source packet are sent to the SPAN destination ports. For example, a bidirectional (both ingress and egress) SPAN session is configured for sources a1 and a2 to a destination port d1. If a packet enters the switch through a1 and gets switched to a2, both the incoming and outgoing packets are sent to destination port d1. Both packets would be the same (if a Layer 3 rewrite occurs, the packets are different). For the RSPAN sessions with the sources that are distributed in multiple switches, the destination ports might forward multiple copies of the same packet.

Understanding How the Mini Protocol Analyzer Works

The Mini Protocol Analyzer copies network traffic from a source port (see the "Source Port" section on page 49-2 for an explanation of a source port). A Mini Protocol Analyzer session differs from a SPAN session in that the copied source port traffic from a Mini Protocol Analyzer session travels over the switch backplane where it is written to an output file. By default, the output file is stored on the flash memory of the switch. No destination port is required for the Mini Protocol Analyzer.

Once the file is created, you open and view the file using the Ethereal Network Protocol Analyzer. The Ethereal Network Protocol Analyzer is open source software and is available from http://www.ethereal.com.

You specify a single port as the source port. The source port can be either an access port or a trunk port. You cannot specify a VLAN as a source port. The Mini Protocol Analyzer also captures double tagged frames on dot1qtunnel, PAgP and LACP channel ports.

The functional differences between the Mini Protocol Analyzer and SPAN are as follows:

- The Mini Protocol Analyzer does not use a SPAN destination port, which frees up an extra port for network traffic.
- The Mini Protocol Analyzer does not require an external traffic analyzer such as a remote monitor.
- You do not require physical access to the switch to attach a network analyzer. You can access and download the output file from the Flash memory.

Mini Protocol Analyzer Session

A Mini Protocol Analyzer session is an association of a source port with the output file to which the source port traffic is mirrored. You can filter the type of traffic that is monitored by the following criteria:

- Source IP address
- Destination IP address
- Source MAC address
- Destination MAC address

By default, all traffic is captured. If you specify any combination of source and destination filters, only the traffic that matches those source and destination filters will be captured. The source and destination filters are applied on a Boolean logical OR basis—if traffic meets any of the criteria specified in any of the filters, it will be captured.

If you specify a filter based on the packet size, the packets that are larger than the specified size are captured and truncated to the specified size. You can specify a maximum of 16 filters for a Mini Protocol Analyzer session. Enter the **set packet-capture snap-length** command to specify the length to which the packets are truncated. The packet length is not counted against the maximum number of filters.

You can specify the filtering criteria either before or after you begin the Mini Protocol Analyzer session. If you specify the filtering criteria before you start the Mini Protocol Analyzer session, only the traffic that meets the filtering criteria is captured and sent to the output file. You can also filter the captured traffic after the Mini Protocol Analyzer session completes by using the Filter function of the Ethereal Network Protocol Analyzer.

You enable or disable a Mini Protocol Analyzer session using CLI or SNMP commands.

A Mini Protocol Analyzer session becomes active when both of the following criteria are met:

- After the source port becomes operational.
- After you enter the **set packet-capture start** command.

SPAN, RSPAN and Mini Protocol Analyzer Session Limits

You can configure (and store in NVRAM) a maximum of 30 SPAN sessions or 29 SPAN sessions and (store in the Flash memory) one Mini Protocol Analyzer session in a Catalyst 6500 series switch.

See Table 49-1 for the supported combinations of SPAN, RSPAN, and Mini Protocol Analyzer sessions. You can configure multiple ports or VLANs as sources for each SPAN session, and you can configure a single source port for each Mini Protocol Analyzer session.

Table 49-1 SPAN RSPAN and Mini Protocol Analyzer Session Limits

SPAN, RSPAN, and Mini Protocol Analyzer Sessions	Catalyst 6500 Series Switches
rx or **both** SPAN sessions	2[1]
tx SPAN sessions	4
Mini Protocol Analyzer sessions	1
tx, rx, or both RSPAN source sessions	1[2]
RSPAN destinations	24
Total SPAN sessions	30[3]

1. Each RSPAN source session or Mini Protocol Analyzer session reduces the limit for **rx** or **both** SPAN sessions by one.
2. Supervisor Engine 720 supports two RSPAN source sessions.
3. 2 **rx** or **both** SPAN sessions + 4 **tx** SPAN sessions + 24 RSPAN destination sessions = 30 total SPAN sessions. A Mini Protocol Analyzer session counts as one **rx** or **both** SPAN session.

Configuring SPAN on the Switch

These sections describe how to configure SPAN:

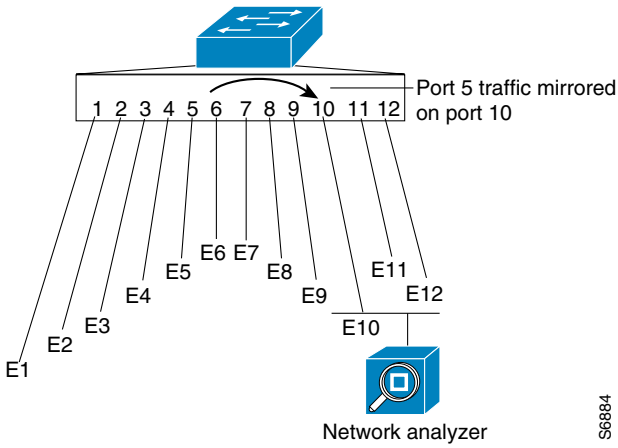
SPAN Hardware Requirements

All Catalyst 6500 series switch supervisor engines support SPAN.

Understanding How SPAN Works

SPAN selects the network traffic for analysis by a network analyzer such as a SwitchProbe device or other Remote Monitoring (RMON) probe. SPAN mirrors the traffic from one or more source ports on any VLAN, from one or more VLANs, or from the sc0 console interface to the destination ports for analysis (see Figure 49-1). In Figure 49-1, all traffic on Ethernet port 5 (the source port) is mirrored to Ethernet port 10. A network analyzer on Ethernet port 10 receives all network traffic from Ethernet port 5 without being physically attached to it.

Figure 49-1 SPAN Configuration



For SPAN configuration, the source ports and the destination ports must be on the same switch.

SPAN does not affect the switching of network traffic on the source ports; a copy of the packets that are received or transmitted by the source ports is sent to the destination ports.

SPAN Configuration Guidelines

This section describes the guidelines for configuring SPAN:

- Use a network analyzer to monitor ports.
- For the SPAN source ports, SPAN is not supported with the ATM ports; it works with the Ethernet 10/100/1000-Mbps ports and 10-Gbps ports.
- When enabled, SPAN uses any previously entered configuration. If you have not entered any configuration commands, SPAN uses the default parameters.
- If you specify multiple SPAN source ports, the ports can belong to the different VLANs.
- See the "SPAN, RSPAN and Mini Protocol Analyzer Session Limits" section on page 49-5.
- The RSPAN sessions can coexist with the SPAN sessions within the SPAN/RSPAN limits that are described in the "SPAN, RSPAN and Mini Protocol Analyzer Session Limits" section on page 49-5.
- The optional **inpkts** keyword is disabled by default. Use the **inpkts** keyword with the optional **enable** keyword to allow the SPAN destination ports to receive the normal incoming traffic. Enter the optional **disable** keyword to prevent the SPAN destination ports from receiving the normal incoming traffic.
- When you enable the optional **inpkts** keyword, a warning message notifies you that the destination port does not support the Spanning Tree Protocol (STP) and may cause loops if you enable this option.
- Learning is enabled by default. Use the **inpkts** keyword with the optional **learning** keyword to enable or disable learning for a specific port.
- You can specify a Multilayer Switch Module (MSM) port as the SPAN source port. However, you cannot specify an MSM port as the SPAN destination port.
- When you configure multiple SPAN sessions, the destination module number/port number must be known to index the particular SPAN session.

- If any of the VLANs on the SPAN source port(s) are blocked by spanning tree, you may see extra packets that are transmitted on the destination port(s) that were not actually transmitted out the source port(s). The extra packets are sent through the switch fabric to the source port and are blocked by spanning tree at the source port.

Caution In software releases before software release 8.4(1), if you used the **set span** command without the **create** keyword, and you had only one session configured, the session was overwritten. If two SPAN sessions were already configured, you received an error message. If a matching destination port existed, the particular session was overwritten (with or without specifying the **create** keyword). If you specified the **create** keyword and there was no matching destination port, the session was created.

In software release 8.4(1) and later releases, the **create** keyword has been removed from the **set span** command. When you enable a SPAN session without the **create** keyword, and another session is available, the first session is not overwritten.

Configuring SPAN from the CLI

To configure SPAN, you specify the source, the destination ports, the direction of the traffic through the source that you want to mirror to the destination ports, and if the destination port can receive the packets.

To configure a SPAN port, perform this task in privileged mode:

	Task	Command
Step 1	Configure the SPAN source and destination ports.	**set span** {*src_mod/src_ports* | *src_vlans* | **sc0**} {*dest_mod/dest_port*} [**rx** | **tx** | **both**] [**session** *session_number*] [**inpkts** {**enable** | **disable**}] [**learning** {**enable** | **disable**}] [**multicast** {**enable** | **disable**}] [**filter** *vlans...*]
Step 2	Verify the SPAN configuration.	**show span**

Caution If the SPAN destination port is connected to another device and you enable reception of the incoming packets (using the **inpkts enable** keywords), the SPAN destination port receives the traffic for whatever VLAN to which the SPAN destination ports belong. The SPAN destination port does *not* participate in spanning tree for that VLAN. Use caution when using the **inpkts** keyword to avoid creating network loops with the SPAN destination port or assigning the SPAN destination port to an unused VLAN.

This example shows how to configure SPAN so that both the transmit and receive traffic from port 1/1 (the SPAN source) is mirrored on port 2/1 (the SPAN destination):

```
Console> (enable) set span 1/1 2/1

Destination     : Port 2/1
Admin Source    : Port 1/1
Oper Source     : Port 1/1
Direction       : transmit/receive
Incoming Packets: disabled
Learning        : enabled
Multicast       : enabled
Filter          : -
```

This example shows how to set VLAN 522 as the SPAN source and port 2/1 as the SPAN destination:

```
Console> (enable) set span 522 2/1

Destination     : Port 2/1
Admin Source    : VLAN 522
Oper Source     : Port 3/1-2
Direction       : transmit/receive
Incoming Packets: disabled
Learning        : enabled
Multicast       : enabled
Filter          : -
Console> (enable)
```

This example shows how to set VLAN 522 as the SPAN source and port 2/12 as the SPAN destination. Only the transmit traffic is monitored. The normal incoming packets on the SPAN destination port are allowed.

```
Console> (enable) set span 522 2/12 tx inpkts enable

Destination     : Port 2/12
Admin Source    : VLAN 522
Oper Source     : Port 2/1-2
Direction       : transmit
Incoming Packets: enabled
Learning        : enabled
Multicast       : enabled
Filter          : -
Console> (enable)
```

This example shows how to set port 3/2 as the SPAN source and port 2/2 as the SPAN destination:

```
Console> (enable) set span 3/2 2/2 tx create

Destination     : Port 2/1
Admin Source    : port 3/1
Oper Source     : Port 3/1
Direction       : transmit/receive
Incoming Packets: disabled

Destination     : Port 2/2
Admin Source    : port 3/2
Oper Source     : Port 3/2
Direction       : transmit
Incoming Packets: disabled
Learning        : enabled
Multicast       : enabled
Filter          : -
Console> (enable)
```

To disable SPAN, perform this task in privileged mode:

Task	Command
Disable SPAN on the switch.	**set span disable** [*dest_mod*/*dest_port* | *all*]

This example shows how to disable SPAN on the switch:

```
Console> (enable) set span disable 2/1
This command will disable your span session.
Do you want to continue (y/n) [n]?y
Disabled port 2/1 to monitor transmit traffic of VLAN 522
Console> (enable)
```

Configuring RSPAN on the Switch

These sections describe how to configure RSPAN:

RSPAN Hardware Requirements

The RSPAN supervisor engine requirements are as follows:

- For source switches—The Catalyst 6500 series switch with any of the following:
  - Supervisor Engine 1A and Policy Feature Card (PFC): WS-X6K-SUP1A-PFC
  - Supervisor Engine 1A, PFC, and Multilayer Switch Feature Card (MSFC): WS-X6K-SUP1A-MSFC
  - Supervisor Engine 1A, PFC, and MSFC2: WS-X6K-S1A-MSFC2
  - Supervisor Engine 2 and PFC2: WS-X6K-S2-PFC2
  - Supervisor Engine 1A, PFC, and MSFC2: WS-X6K-S1A-MSFC2
  - Supervisor Engine 720 with the following onboard components: Policy Feature Card 3A (PFC3A/PFC3B/PFC3BXL), Multilayer Switch Feature Card 3 (MSFC3), and integrated 720-Gbps switch fabric: WS-SUP720
  - Supervisor Engine 32, PFC3B/PFC3BXL, and MSFC2A: WS-SUP32-GE-3B
- For destination or intermediate switches—Any Cisco switch supporting RSPAN VLAN

No third party or other Cisco switches can be placed in the end-to-end path for RSPAN traffic.

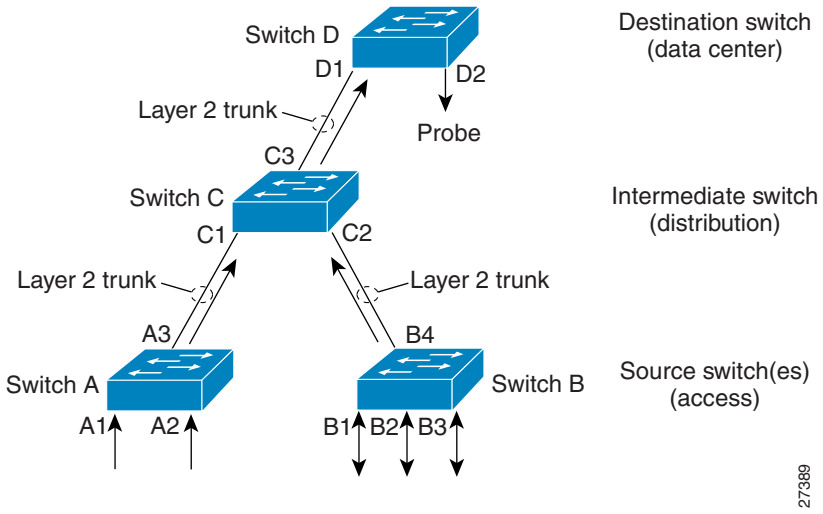
Understanding How RSPAN Works

Note See the "Understanding How SPAN and RSPAN Work" section on page 49-1 for the concepts and terminology that apply to both the SPAN and RSPAN configurations.

RSPAN has all the features of SPAN (see the "Understanding How SPAN Works" section on page 49-6), plus support for the source ports and the destination ports that are distributed across multiple switches, allowing remote monitoring of multiple switches across your network (see Figure 49-2).

The traffic for each RSPAN session is carried over a user-specified RSPAN VLAN that is dedicated for that RSPAN session in all participating switches. The SPAN traffic from the sources, which cannot be in the RSPAN VLAN, is switched to the RSPAN VLAN and is forwarded to the destination ports that are configured in the RSPAN VLAN. The traffic type for the sources (ingress, egress, or both) in an RSPAN session can be different in the different source switches but is the same for all the sources in each source switch for each RSPAN session. Do not configure any ports in an RSPAN VLAN except those that are selected to carry the RSPAN traffic. Learning is disabled on the RSPAN VLAN.

Figure 49-2 RSPAN Configuration



RSPAN Configuration Guidelines

This section describes the guidelines for configuring RSPAN:

Tip As RSPAN VLANs have special properties, we recommend that you reserve a few VLANs across your network for use as RSPAN VLANs. Do not assign an access port to these VLANs.

Tip You can apply an output access control list (ACL) to the RSPAN traffic to selectively filter the specific flows. Specify these ACLs on the RSPAN VLAN in the RSPAN source switches.

- All the items in the "SPAN Configuration Guidelines" section on page 49-7 apply to RSPAN.
- The RSPAN sessions can coexist with the SPAN sessions within the SPAN/RSPAN limits that are described in the "SPAN, RSPAN and Mini Protocol Analyzer Session Limits" section on page 49-5.
- For the RSPAN configuration, you can distribute the source ports and the destination ports across multiple switches.
- For RSPAN, trunking is required if you have a source switch with all the source ports in one VLAN (VLAN 2 for example) and it is connected to the destination switch through an uplink port that is also in VLAN 2. With RSPAN, the traffic is forwarded to the remote switches in the RSPAN VLAN. The RSPAN VLAN is configured only on trunk ports and not on access ports.
- The learning option applies to the RSPAN destination ports only.
- RSPAN does not support monitoring the BPDU packets or Layer 2 protocol packets such as Cisco Discovery Protocol (CDP), Dynamic Trunking Protocol (DTP), and VLAN Trunking Protocol (VTP).
- To optimize the bandwidth utilization in the connecting links, you can configure the quality of service (QoS) parameters for the RSPAN VLAN in each of the participating source, intermediate, or destination switches.

- Each Catalyst 6500 series switch can source a maximum of one RSPAN session (ingress, egress, or both). When you configure a remote ingress or bidirectional SPAN session in a source switch, the limit for the local ingress or bidirectional SPAN sessions is reduced to one. There are no limits on the number of RSPAN sessions that are carried across the network within the RSPAN session limits (see the "SPAN, RSPAN and Mini Protocol Analyzer Session Limits" section on page 49-5).
- The RSPAN VLANs cannot be included as the sources for the port-based RSPAN sessions when the source trunk ports have active RSPAN VLANs. Additionally, the RSPAN VLANs cannot be the sources in the VSPAN sessions.
- You can configure any VLAN as an RSPAN VLAN if these conditions are met:
  - The same RSPAN VLAN is used for an RSPAN session in all the switches.
  - All participating switches have the appropriate hardware and software.
  - No access port (including the sc0 interface) is configured in the RSPAN VLAN.
- If you enable VTP and VTP pruning, the RSPAN traffic is pruned in the trunks to prevent the unwanted flooding of the RSPAN traffic across the network.
- If you enable the GARP VLAN Registration Protocol (GVRP) and the GVRP requests conflict with the existing RSPAN VLANs, you might observe unwanted traffic in the RSPAN sessions.
- You can use the RSPAN VLANs in Inter-Switch Link (ISL) to dot1q mapping. However, ensure that the special properties of RSPAN VLANs are supported in all the switches to avoid the unwanted traffic in these VLANs.

Configuring RSPAN

The first step in configuring an RSPAN session is to select an RSPAN VLAN for the RSPAN session that *does not* exist in any of the switches that will participate in RSPAN. With VTP enabled in the network, you can create the RSPAN VLAN in one switch and VTP propagates it to the other switches in the VTP domain.

Use VTP pruning to get an efficient flow of RSPAN traffic, or manually delete the RSPAN VLAN from all trunks that do not need to carry the RSPAN traffic.

Once the RSPAN VLAN is created, you configure the source and destination switches by entering the **set rspan** command.

To configure the RSPAN VLANs, perform this task in privileged mode:

	Task	Command
Step 1	Configure the RSPAN VLANs.	**set vlan** *vlan* [**rspan**]
Step 2	Verify the RSPAN VLAN configuration.	**show vlan**

This example shows how to set VLAN 500 as an RSPAN VLAN and verify the configuration:

```
Console> (enable) set vlan 500 rspan
vlan 500 configuration successful
Console> (enable)
Console> (enable) show vlan
.
display truncated
.
VLAN DynCreated  RSPAN
---- ---------- --------
1    static     disabled
2    static     disabled
3    static     disabled
99   static     disabled
500  static     enabled
Console> (enable)
```

To configure the RSPAN source ports, perform this task in privileged mode:

	Task	Command
Step 1	Configure the RSPAN source ports. Use this command on each of the source switches that participate in RSPAN.	**set rspan source** {*src_mod/src_ports...* | *vlans...* | **sc0**} {*rspan_vlan*} [**rx** | **tx** | **both**] **session** *session_number* [**multicast** {**enable** | **disable**}] [**filter** *vlans...*] [**create**]
Step 2	Verify the RSPAN configuration.	**show rspan**

This example shows how to specify ports 4/1 and 4/2 as the ingress source ports for RSPAN VLAN 500:

```
Console> (enable) set rspan source 4/1-2 500 rx
Rspan Type      : Source
Destination     : -
Rspan Vlan      : 500
Admin Source    : Port 4/1-2
Oper Source     : None
Direction       : receive
Incoming Packets: -
Learning        : -
Multicast       : enabled
Filter          : -
Console> (enable)
```

To configure the RSPAN source VLANs, perform this task in privileged mode:

	Task	Command
Step 1	Configure the RSPAN source VLANs. All the ports in the source VLAN become the operational source ports.	**set rspan source** {*src_mod/src_ports...* | *vlans...* | **sc0**} {*rspan_vlan*} [**rx** | **tx** | **both**] **session** *session_number* [**multicast** {**enable** | **disable**}] [**filter** *vlans...*] [**create**]
Step 2	Verify the RSPAN configuration.	**show rspan**

This example shows how to specify VLAN 200 as a source VLAN for RSPAN VLAN 500 (selecting the optional **rx** keyword makes all the ports in the VLAN ingress ports):

```
Console> (enable) set rspan source 200 500 rx
Rspan Type      : Source
Destination     : -
Rspan Vlan      : 500
Admin Source    : VLAN 200
Oper Source     : None
Direction       : receive
Incoming Packets: -
Learning        : -
Multicast       : enabled
Filter          : -
Console> (enable)
```

To configure the RSPAN destination ports, perform this task in privileged mode:

	Task	Command
Step 1	Configure the RSPAN destination ports. Use this command on each of the destination switches that participate in RSPAN.	**set rspan destination** *mod/port* {*rspan_vlan*} **session** *session_number* [**inpkts** {**enable** | **disable**}] [**learning** {**enable** | **disable**}] [**create**]
Step 2	Verify the RSPAN configuration.	**show rspan**

```
Console> (enable) set rspan destination 3/1 500
Rspan Type      : Destination
Destination     : Port 3/1
Rspan Vlan      : 500
Admin Source    : -
Oper Source     : -
Direction       : -
Incoming Packets: disabled
Learning        : enabled
Multicast       : -
Filter          : -
Console> (enable)
```

To disable RSPAN, perform this task in privileged mode:

Task	Command
Disable RSPAN on the switch.	**set rspan disable source** [*rspan_vlan* | **all**] **set rspan disable destination** [*mod/port* | **all**]

This example shows how to disable all the enabled source sessions:

```
Console> (enable) set rspan disable source all
This command will disable all remote span source session(s).
Do you want to continue (y/n) [n]? y
Disabled monitoring of all source(s) on the switch for remote span.
Console> (enable)
```

This example shows how to disable one source session by *rspan_vlan* number:

```
Console> (enable) set rspan disable source 903
Disabled monitoring of all source(s) on the switch for rspan_vlan 903.
Console> (enable)
```

This example shows how to disable all the enabled destination sessions:

```
Console> (enable) set rspan disable destination all
This command will disable all remote span destination session(s).
Do you want to continue (y/n) [n]? y
Disabled monitoring of remote span traffic for all rspan destination ports.
Console> (enable)
```

This example shows how to disable one destination session by *mod*/*port*:

```
Console> (enable) set rspan disable destination 4/1
Disabled monitoring of remote span traffic on port 4/1.
Console> (enable)
```

RSPAN Configuration Examples

These sections describe how to configure RSPAN:

Configuring a Single RSPAN Session

This example shows how to configure a single RSPAN session. Figure 49-3 shows an RSPAN configuration; see Table 49-2 for the commands to configure this RSPAN session. Table 49-2 assumes that you have already set up RSPAN VLAN 901 for this session on all the switches using the **set vlan** *vlan* **rspan** command. With VTP enabled in the network, you can create the RSPAN VLAN in one switch and VTP propagates it to the other switches in the VTP domain. Note that in the configuration example shown in Table 49-2, the RSPAN session may be disabled in Switch A or B or both without modifying the configuration in Switch C or Switch D.

Figure 49-3 Single RSPAN Session

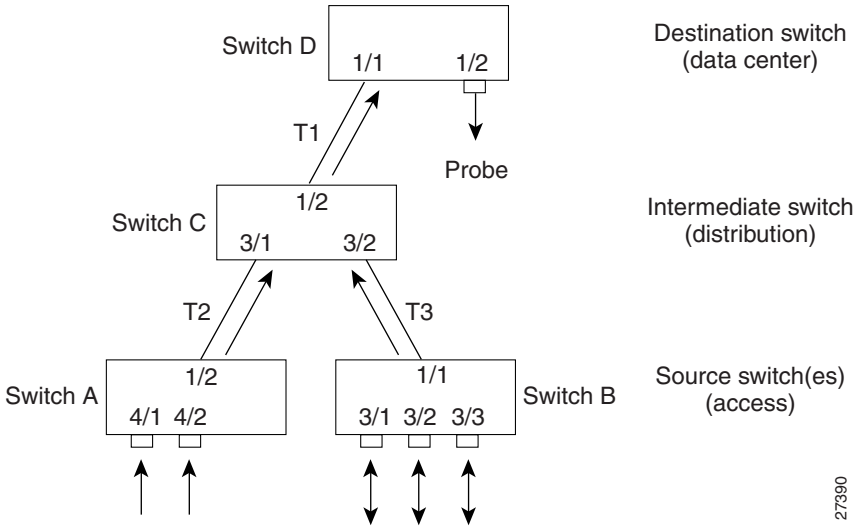


Table 49-2 Configuring a Single RSPAN Session

Switch	Ports	RSPAN VLAN	Direction	RSPAN CLI Commands
A (source)	4/1, 4/2	901	Ingress	**set rspan source 4/1-2 901 rx**
B (source)	3/1, 3/2, 3/3	901	Bidirectional	**set rspan source 3/1-3 901**
C (intermediate)	–	901	–	No RSPAN CLI command needed
D (destination)	1/2	901	–	**set rspan destination 1/2 901**

Modifying an Active RSPAN Session

This example shows how to modify an active RSPAN session. Use Figure 49-3 for reference; see Table 49-3 for the commands to disable an RSPAN session and to add or remove the source ports from an RSPAN session.

Table 49-3 Making Modifications to an Active RSPAN Session

Switch	Action	RSPAN CLI Commands
A (source)	Disable the RSPAN session.	**set rspan disable source 901**
B (source)	Remove source port 3/2 from the RSPAN session.	**set rspan source 3/1, 3/3 901**
B (source)	Add back source port 3/2 to the RSPAN session.	**set rspan source 3/1-3 901**

Adding the RSPAN Source Ports in Intermediate Switches

This example shows how to add the RSPAN source ports in the intermediate switches. Figure 49-4 shows an RSPAN configuration; see Table 49-4 for the commands to configure this RSPAN session. Ports 2/1-2 in Switch C can be configured for the same RSPAN session.

Figure 49-4 Adding the RSPAN Source Ports in Intermediate Switches

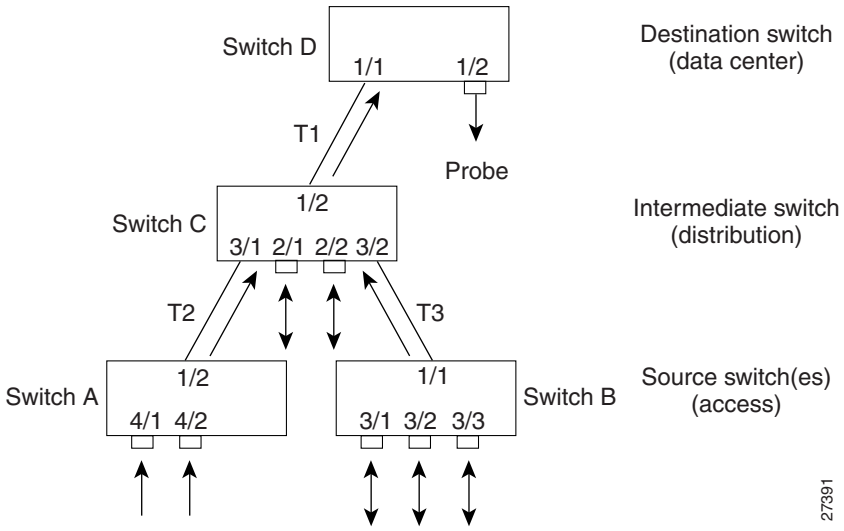


Table 49-4 Adding the RSPAN Source Ports in Intermediate Switches

Switch	Ports	RSPAN VLAN	Direction	RSPAN CLI Commands
A (source)	4/1, 4/2	901	Ingress	**set rspan source 4/1-2 901 rx**
B (source)	3/1, 3/2, 3/3	901	Bidirectional	**set rspan source 3/1-3 901**
C (intermediate)	–	901	–	No RSPAN CLI command needed
C (source)	2/1, 2/2	901	Bidirectional	**set rspan source 2/1-2 901**
D (destination)	1/2	901	–	**set rspan destination 1/2 901**

Configuring Multiple RSPAN Sessions

This example shows how to configure multiple RSPAN sessions. Figure 49-5 shows an RSPAN configuration; see Table 49-5 for the configuration commands to configure this RSPAN session. This example is a typical scenario where the monitoring probes would be placed in the data center and the source ports in the access switches (other ports in any of the switches can also be configured for RSPAN). If there is no change in the route for the SPAN traffic, the destination switch and the intermediate switches need to be configured only once.

In Figure 49-5, two RSPAN sessions are used with RSPAN VLANs 901 (for probe 1) and 902 (for probe 2). The direction of traffic over trunks T1 through T6 is shown only for understanding; the direction of the trunks depends on the STP states of the trunks for the RSPAN VLAN(s). You need to configure the RSPAN VLANs in each of the switches for the RSPAN sessions. With VTP enabled in the network, you can create the RSPAN VLAN in one switch and VTP propagates it to the other switches in that VTP domain. With VTP disabled, create the RSPAN VLANs in each switch.

Figure 49-5 Configuring Multiple RSPAN Sessions

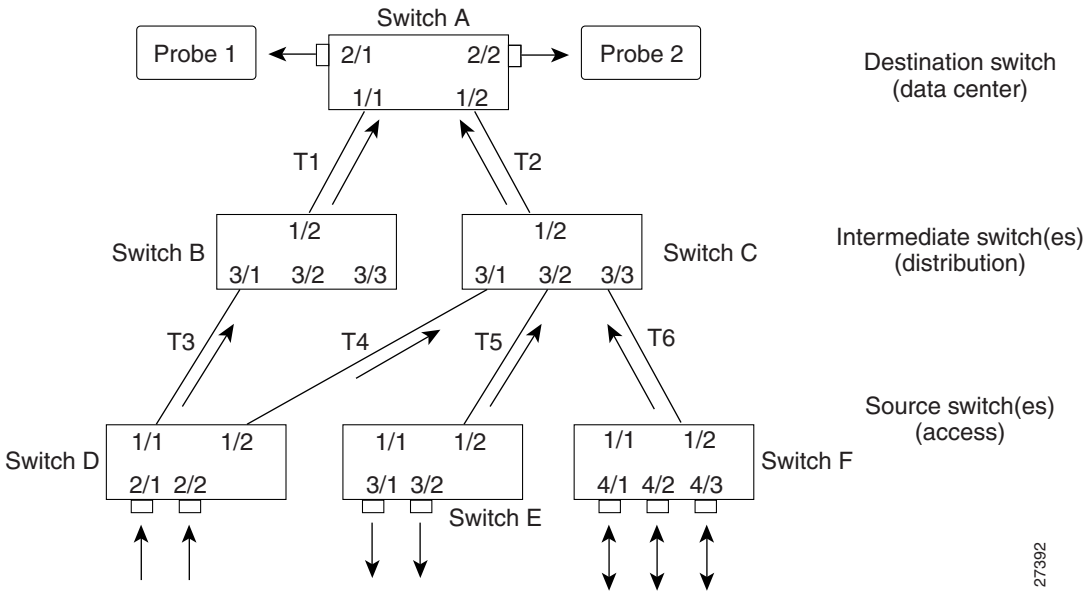


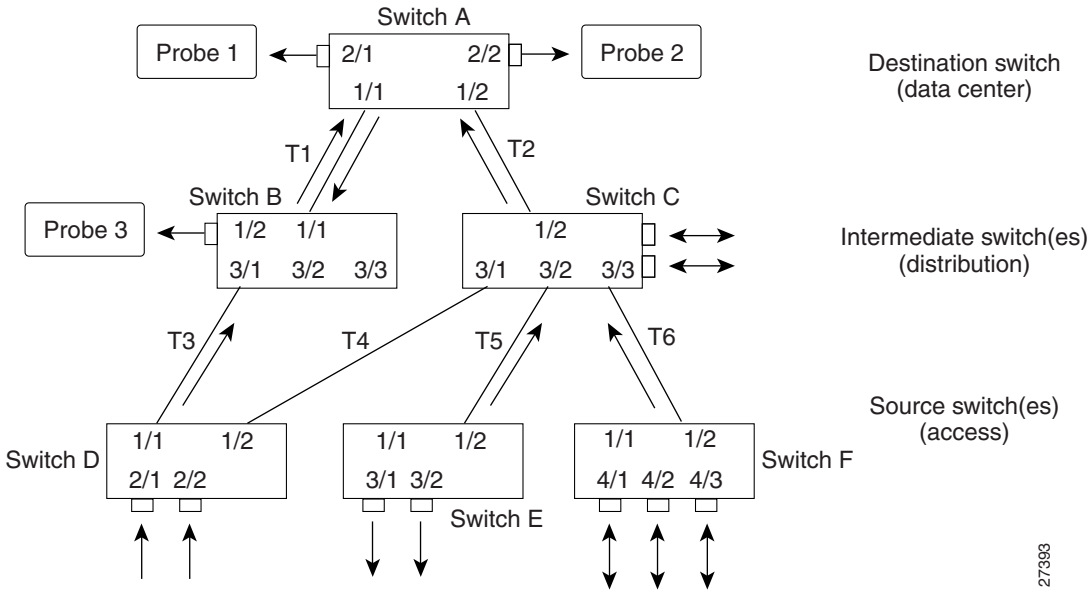
Table 49-5 Configuring Multiple RSPAN Sessions

Switch	Port	RSPAN VLAN(s)	Direction	RSPAN CLI Commands
A (destination)	2/1	901	–	**set rspan destination 2/1 901**
A (destination)	2/2	902	–	**set rspan destination 2/2 902**
B (intermediate)	–	901, 902	–	No RSPAN CLI command needed
C (intermediate)	–	901, 902	–	No RSPAN CLI command needed
D (source)	2/1-2	901	Ingress	**set rspan source 2/1-2 901 rx**
E (source)	3/1-2	901	Egress	**set rspan source 3/1-2 901 tx**
F (source)	4/1-3	902	Both	**set rspan source 4/1-3 902**

Adding Multiple Network Analyzers to an RSPAN Session

You can attach multiple network analyzers (probes) to the same RSPAN session. For example, in Figure 49-6, you can add probe 3 in Switch B to monitor RSPAN VLAN 901 using the **set rspan destination 1/2 901** command. Similarly, you could add the source ports to Switch C.

Figure 49-6 Adding Multiple Probes to an RSPAN Session



Configuring the Mini Protocol Analyzer on the Switch

These sections describe how to configure the Mini Protocol Analyzer on the switch:

- Mini Protocol Analyzer Hardware Requirements, page 49-19
- Understanding How the Mini Protocol Analyzer Works, page 49-19
- Mini Protocol Analyzer Configuration Guidelines, page 49-20
- Configuring the Mini Protocol Analyzer from the CLI, page 49-21

Mini Protocol Analyzer Hardware Requirements

Supervisor Engine 720 and Supervisor Engine 32 support the Mini Protocol Analyzer.

Understanding How the Mini Protocol Analyzer Works

A Mini Protocol Analyzer session mirrors the traffic from a single source port. The source port can be either an access port or a trunk port.

The Mini Protocol Analyzer does not affect the switching of network traffic on the source port. A copy of the packets that are received and transmitted by the source port is sent over the backplane where the copies are stored as a file in the Flash memory of the switch. The file can be stored in the following places:

- For the Supervisor Engine 720: bootflash (default), disk0, or disk1
- For the Supervisor Engine 32: bootdisk (default) or disk0

When a Mini Protocol Analyzer session starts (when you enter the **set packet-capture start** command), the session monitors the source port traffic and stores it on the flash memory until one of the following conditions occur:

- You enter the **set packet-capture stop** command to end the Mini Protocol Analyzer session.
- The number of packets as specified by the **set packet-capture limit** command is reached. A system message is displayed and the Mini Protocol Analyzer session ends.
- The flash device runs out of memory. A system message is displayed and the Mini Protocol Analyzer session ends.

Mini Protocol Analyzer Configuration Guidelines

This section describes the guidelines for configuring the Mini Protocol Analyzer:

- Ensure that your flash memory has enough space to store the output file from the Mini Protocol Analyzer session, or specify filters that limit the size of the output file.
- You use Ethernet 10/100/1000-Mbps ports and 10-Gbps ports as Mini Protocol Analyzer source ports. You cannot use ATM ports, MSFC ports, or service module ports as Mini Protocol Analyzer source ports.
- When enabled, the Mini Protocol Analyzer uses any previously entered configuration. If you have not entered any configuration commands, the Mini Protocol Analyzer uses the default parameters.
- Only one Mini Protocol Analyzer session is allowed on the switch at one time. See the "SPAN, RSPAN and Mini Protocol Analyzer Session Limits" section on page 49-5 for switch-wide limitations regarding SPAN, RSPAN, and the Mini Protocol Analyzer. One Mini Protocol Analyzer session counts as one SPAN session.
- A maximum of 16 filters can be run on the traffic that is being monitored in a Mini Protocol Analyzer session. The **set packet-capture snap-length** command, which specifies the length to which packets are truncated, is not counted against the maximum number of filters. Entering the **clear packet-capture filter** or **clear packet-capture all** command removes all filters.
- If you have saved a file on the flash memory from one Mini Protocol Analyzer session and you start another Mini Protocol Analyzer session with the same output filename, the existing information from the previous Mini Protocol Analyzer session is overwritten. Multiple output files can be stored on the flash memory if the output filenames are different.
- The file system in the flash memory is locked and cannot be modified or accessed while the Mini Protocol Analyzer session is running.
- Because the Mini Protocol Analyzer sends the copied traffic from the source port over the backplane, the performance of your switch might be affected if the source port is processing a large amount of traffic.

- If your switch performance is adversely affected by the amount of traffic that is being processed during a Mini Protocol Analyzer session, the CPU of the switch can become overloaded and cause the copied packets to drop or the control packets, such as the Bridge Protocol Data Units (BPDUs), to drop. If the switch becomes extremely overloaded, you cannot stop the Mini Protocol Analyzer session.
- High Availability (HA) is supported with the Mini Protocol Analyzer. However, if you perform a soft or hard reboot of the switch and a Mini Protocol Analyzer session is in progress, the Mini Protocol Analyzer session will not continue after the reboot until you enter the **set packet-capture start** command.
- You cannot view the copied traffic from a Mini Protocol Analyzer session on the system console. You can only save the copied traffic to a file on the switch's flash memory and view the file using the Ethernet Network Protocol Analyzer.
- You cannot capture Ethernet Out-of-Band Channel (EOBC) traffic with the Mini Protocol Analyzer.
- If any VLAN on the Mini Protocol Analyzer source port is blocked by spanning tree, you might see extra packets that are saved on the flash memory that were not actually transmitted out the source port. The extra packets are sent through the switch fabric to the flash memory and are blocked by spanning tree at the source port.

Configuring the Mini Protocol Analyzer from the CLI

To configure the Mini Protocol Analyzer, you specify the source port and, optionally, the name of the output file to which the copied packets will be written.

To monitor a source port using the Mini Protocol Analyzer, perform this task in privileged mode:

	Task	Command
Step 1	Configure the source port for the Mini Protocol Analyzer session.	**set packet-capture** *mod/port*
Step 2	(Optional) Specify the location and filename of the output file for the Mini Protocol Analyzer session. **Note** The default filename is **bootflash:eth_***yymmdd-hhmmss* where yymmdd-hhmmss is the year, month, day, hour, minute, and second when the Mini Protocol Analyzer session was started.	**set packet-capture dump-file** *device:filename*
Step 3	(Optional) Specify the filtering criteria for the source or destination IP or MAC addresses.	**set packet-capture filter** {**source** | **destination**} {**ip** | **mac**}
Step 4	(Optional) Specify the direction of the traffic to be captured as either receive (**rx**), transmit (**tx**), or **both**. **rx** is the default.	**set packet-capture direction {rx | tx | both}**
Step 5	(Optional) Specify the length to which the packets that are captured by the Mini Protocol Analyzer session are truncated. **Note** The range is 0 to 10258 bytes.	**set packet-capture snap-length** *packet-length*

	Task	Command
Step 6	(Optional) Specify the total number of packets that are captured by the Mini Protocol Analyzer session. **Note** The range is 0 to 32,000 packets. The default is 1,000 packets.	**set packet-capture limit** *packet-number*
Step 7	Verify the Mini Protocol Analyzer configuration.	**show packet-capture**
Step 8	Start the Mini Protocol Analyzer session.	**set packet-capture start**

This example shows how to configure the Mini Protocol Analyzer so that so that all traffic that is sent and received from port 5/1 is copied to a file on the bootflash memory called port_5_1_stats:

```
Console> (enable) set packet-capture 5/1
Capturing port set to 5/1.
Console> (enable) set packet-capture dump-file bootflash:port_5_1_stats
Packet capture dump file name set to  bootflash:port_5_1_stats.
```

The date and time when the Mini Protocol Analyzer session is started will be appended to the output filename. For example, if the Mini Protocol Analyzer session was started July 28, 2008 at 4:54:08 p.m. Greenwich Mean Time (GMT), the filename for the previous example would be port_5_1_stats_080728-165408.

This example shows how to specify that only traffic that has either a destination address of 10.1.1.2 or a destination address of 10.1.1.3 will be captured:

```
Console> (enable) set packet-capture filter destination ip 10.1.1.2
Successfully added the filter string.
Console> (enable) set packet-capture filter destination ip 10.1.1.3
Successfully added the filter string.
```

This example shows how to specify the direction of the traffic to be captured:

```
Console> (enable) set packet-capture direction tx
Packets from transmit (tx) direction will be captured.
```

This example shows how to specify that all packets will be captured but packets that have a length of 5,000 bytes or larger will be truncated to 5,000 bytes:

```
Console> (enable) set packet-capture snap-length 5000
Packets captured will be truncated to 5000 bytes.
```

This example shows how to specify that 500 packets will be captured during the Mini Protocol Analyzer session. After 500 packets have been captured, the Mini Protocol Analyzer session will end.

```
Console> (enable) set packet-capture limit 500
Packet capture number set to 500.
```

This example shows how to verify the configuration of the Mini Protocol Analyzer:

```
Console> (enable) show packet-capture
Packet-capture parameter          Value
--------------------------------  ---------------------------------------------
Operational Status                Not-running
Dump File Name                    bootflash:port_5_1_stats
Direction                         rx
Filter - Source IP                None
Filter - Destination IP           host 10.1.1.2/32,10.1.1.3/32
Filter - Source MAC address       None
Filter - Destination MAC address  None
Number of packets to capture      500
Packet Snap Length                5000
```

```
Source Port                     5/1
Bytes Captured                  7
```

This example shows how to start the Mini Protocol Analyzer session:

```
Console> (enable) set packet-capture start
Packet capturing can result in protocol packets(STP, UDLD, PAGP, etc.)
getting dropped resulting in network instability. Also, it can affect
system performance or inband connectivity as sc0/sc1 interface packets
can be dropped without warning
Do you want to continue(y/n) [n]? y
Console> (enable) 2006 Jul 28 16:54:08 %SYS-5-SPAN_CFGSTATECHG:local span sessio
n active for session Number 1
2006 Jul 28 16:54:08 %SYS-5-PKTCAP_START:Packet capture session active
2006 Jul 28 16:55:34 %SYS-5-PKTCAP_STOPPKT:Packet capture session ended after ca
pturing 300 packets
2006 Jul 28 16:55:34 %SYS-5-SPAN_CFGSTATECHG:local span session inactive for ses
sion Number 1
CCCCCCCCCCCCCCC
```

CHAPTER 50

Using Switch TopN Reports

This chapter describes how to use the Switch TopN Reports utility on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Understanding How the Switch TopN Reports Utility Works

These sections describe how the Switch TopN Reports utility works:

TopN Reports Overview

The Switch TopN Reports utility allows you to collect and analyze data for each physical port on a switch.

Note You cannot use the Switch TopN Reports utility to generate reports on the Multilayer Switch Module (MSM) or Multilayer Switch Feature Card (MSFC, MSFC2, MSFC2A, and MSFC3) ports.

Note When calculating the port utilization, the Switch TopN Reports utility bundles the Tx and Rx lines into the same counter and also looks at the full-duplex bandwidth when calculating the percentage of utilization. For example, a Gigabit Ethernet port would be 2000-Mbps full duplex.

The Switch TopN Reports utility collects the following data for each physical port:

- Port utilization (**util**)
- Number of in/out bytes (**bytes**)
- Number of in/out packets (**pkts**)
- Number of in/out broadcast packets (**bcst**)
- Number of in/out multicast packets (**mcst**)
- Number of in errors (**in-errors**)
- Number of buffer-overflow errors (**buf-ovflw**)

When the Switch TopN Reports utility starts, it gathers data from the appropriate hardware counters and then goes into sleep mode for a user-specified period. When the sleep time ends, the utility gathers the current data from the same hardware counters, compares the current data from the earlier data, and stores the difference. The data for each port is sorted using a user-specified metric that is chosen from the values that are shown in Table 50-1.

Table 50-1 Valid Switch TopN Reports Metric Values

Metric Value	Definition
util	Utilization
bytes	Input/output bytes
pkts	Input/output packets
bcst	Input/output broadcast packets
mcst	Input/output multicast packets
errors	Input errors
overflow	Buffer overflows

Running Switch TopN Reports without the Background Keyword

If you enter the **show top** command without specifying the **background** keyword, processing begins but the system prompt does not reappear on the screen and you cannot enter the other commands while the report is being generated.

You can terminate the Switch TopN process before it finishes by pressing **Ctrl-C** from the same console or Telnet session, or by opening a separate console or Telnet session and entering the **clear top** [*report_num*] command. After the Switch TopN Reports utility finishes processing the data, it displays the output on the screen immediately. The output is not saved.

Running Switch TopN Reports with the Background Keyword

If you enter the **show top** command and specify the **background** keyword, processing begins and the system prompt reappears immediately. When processing completes, the reports do not display immediately on the screen but are saved for later viewing.

The system notifies you when the reports are complete by sending a syslog message to the screen. Enter the **show top report** [*report_num*] command to view the completed reports. The system displays only those reports that are completed. For reports that are not completed, the system displays a short description of the Switch TopN process information.

You can terminate a Switch TopN process invoked with the **background** keyword only by entering the **clear top** [*report_num*] command. Pressing **Ctrl-C** does not terminate the process. The completed reports remain available for viewing until you remove them by entering the **clear top** {**all** | *report_num*} command.

Running and Viewing Switch TopN Reports

To start the Switch TopN Reports utility in the background and view the results, perform this task in privileged mode:

	Task	Command
Step 1	Run the Switch TopN Reports utility in the background.	**show top** [*N*] [*metric*] [**interval** *interval*] [*port_type*] **background**
Step 2	View the generated report when it is complete.	**show top report** [*report_num*]

Note You must run the Switch TopN Reports utility with the **background** keyword in order to use the **show top report** command to view the completed report contents. Otherwise, the report is displayed immediately upon completion of the process, and the results are not saved.

If you specify the *report_num* with the **show top report** command, the associated report is displayed. Each process is associated with a unique report number.

If you do not specify the *report_num* variable, all active Switch TopN processes and all available Switch TopN reports for the switch are displayed. All Switch TopN processes (both with and without the **background** keyword) are shown in the list.

This example shows how to run the Switch TopN Reports utility with the **background** keyword:

```
Console> (enable) show top 5 pkts background
Console> (enable) 06/16/1998,17:21:08:MGMT-5:TopN report 4 started by Console//.
Console> (enable) 06/16/1998,17:21:39:MGMT-5:TopN report 4 available.
Console> (enable) show top report 4
Start Time:     06/16/1998,17:21:08
End Time:       06/16/1998,17:21:39
PortType:       all
Metric:         pkts (Tx + Rx)
Port  Band- Uti Bytes               Pkts       Bcst       Mcst       Error Over
      width  %  (Tx + Rx)           (Tx + Rx)  (Tx + Rx)  (Tx + Rx)  (Rx)  flow
----- ----- --- -------------------- ---------- ---------- ---------- ----- ----
 1/1    100   0                7950         81          0         81     0    0
 2/1    100   0                2244         29          0         23     0    0
 1/2    100   0                1548         12          0         12     0    0
 2/10   100   0                   0          0          0          0     0    0
 2/9    100   0                   0          0          0          0     0    0
Console> (enable)
```

To run the Switch TopN Reports utility in the foreground and view the results immediately, perform this task in privileged mode:

Task	Command
Run the Switch TopN Reports utility in the foreground.	**show top** [*N*] [*metric*] [**interval** *interval*] [*port_type*]

This example shows how to run the Switch TopN Reports utility in the foreground:

```
Console> (enable) show top 5 pkts
Start Time:     06/16/1998,17:26:38
End Time:       06/16/1998,17:27:09
PortType:       all
Metric:         pkts (Tx + Rx)
Port  Band- Uti Bytes               Pkts       Bcst       Mcst       Error Over
      width  %  (Tx + Rx)           (Tx + Rx)  (Tx + Rx)  (Tx + Rx)  (Rx)  flow
----- ----- --- -------------------- ---------- ---------- ---------- ----- ----
 2/1    100   0                10838         94          2         26     0    0
 1/1    100   0                 7504         79          0         79     0    0
 1/2    100   0                 2622         21          0         21     0    0
 2/10   100   0                    0          0          0          0     0    0
 2/9    100   0                    0          0          0          0     0    0
Console> (enable)
```

To display the stored and pending reports, perform this task in privileged mode:

Task	Command
Display a report.	**show top report** [*report_num*]

Note To display all stored and pending reports, do not specify a *report_num*.

This example shows how to display a specific report and how to display all stored and pending reports:

```
Console> (enable) show top report 5
Start Time:     06/16/1998,17:29:40
End Time:       06/16/1998,17:30:11
PortType:       all
Metric:         overflow
Port  Band- Uti Bytes               Pkts       Bcst       Mcst       Error Over
      width  %  (Tx + Rx)           (Tx + Rx)  (Tx + Rx)  (Tx + Rx)  (Rx)  flow
----- ----- --- -------------------- ---------- ---------- ---------- ----- ----
 1/1    100   0                 7880         83          0         83     0    0
 2/12   100   0                    0          0          0          0     0    0
 2/11   100   0                    0          0          0          0     0    0
 2/10   100   0                    0          0          0          0     0    0
 2/9    100   0                    0          0          0          0     0    0
Console> (enable) show top report
Rpt  Start time          Int N   Metric     Status   Owner (type/machine/user)
---  ------------------- --- --- ---------- -------- -------------------------
  1  06/16/1998,17:05:00 30  20  Util       done     telnet/172.16.52.3/
  2  06/16/1998,17:05:59 30  5   Util       done     telnet/172.16.52.3/
  3  06/16/1998,17:08:06 30  5   Pkts       done     telnet/172.16.52.3/
  4  06/16/1998,17:21:08 30  5   Pkts       done     Console//
  5  06/16/1998,17:29:40 30  5   Overflow   pending  Console//
Console> (enable)
```

To remove the stored reports, perform this task in privileged mode:

Task	Command
Remove the reports.	**clear top** {**all** | *report_num*}

Note Use the **all** keyword to remove all the completed reports. The command **clear top all** command does not clear the pending reports; only the reports that have completed are cleared.

This example shows how to remove a specific report and how to remove all stored reports:

```
Console> (enable) clear top 4
Console> (enable) 06/16/1998,17:36:45:MGMT-5:TopN report 4 killed by Console//.
Console> (enable) clear top all
06/16/1998,17:36:52:MGMT-5:TopN report 1 killed by Console//.
06/16/1998,17:36:52:MGMT-5:TopN report 2 killed by Console//.
Console> (enable) 06/16/1998,17:36:52:MGMT-5:TopN report 3 killed by Console//.
06/16/1998,17:36:52:MGMT-5:TopN report 5 killed by Console//.
Console> (enable)
```

CHAPTER 51

Configuring Multicast Services

This chapter describes how to configure Internet Group Management Protocol (IGMP) snooping, GARP Multicast Registration Protocol (GMRP), Router-Port Group Management Protocol (RGMP), and bidirectional protocol independent multicast (PIM) on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Understanding How Multicasting Works

These sections describe how multicasting works on the Catalyst 6500 series switches:

Multicasting and Multicast Services Overview

IGMP snooping manages the multicast traffic in the switches by allowing the directed switching of the IP multicast traffic. GMRP is protocol independent and can manage both IP multicast traffic and any Layer 2 multicast traffic.

The switches can use IGMP snooping or GMRP to configure the switch ports dynamically so that the IP multicast traffic is forwarded only to those ports that are associated with the IP multicast hosts. The IGMP software components run on both the Cisco router and the switch.

Note For more information on IP multicast and IGMP, refer to RFC 1112. GMRP is described in IEEE 802.1p.

You can statically configure the multicast groups by entering the **set cam static** command. The multicast groups that are learned through IGMP snooping are dynamic. If you specify group membership for a multicast group address, your static setting supersedes any automatic manipulation by IGMP snooping or GMRP. The multicast group membership lists can consist of both user-defined settings and settings that are learned through IGMP snooping or GMRP.

Understanding How IGMP Snooping Works

Note You cannot enable IGMP snooping on a switch if GMRP is already enabled on the switch.

Note You can run IGMP snooping on any Catalyst 6500 series supervisor engine model (Supervisor Engine 1, Supervisor Engine 1A, Supervisor Engine 2, Supervisor Engine 720, and Supervisor Engine 32). A PFC is not required to enable IGMP snooping. Cisco Group Management Protocol (CGMP) is not supported on the Catalyst 6500 series switches, although the CGMP server is supported on the MSFC. To support the CGMP client devices, configure the MSFC as a CGMP server.

Note IGMP version 3 snooping is not supported on the systems with a Supervisor Engine 1 or Supervisor Engine 1A.

IGMP snooping manages the multicast traffic at Layer 2 on the Catalyst 6500 series switches by allowing the directed switching of the IP multicast traffic.

The switches can use IGMP snooping to configure the Layer 2 interfaces dynamically so that the IP multicast traffic is forwarded only to those interfaces that have expressed interest in particular IP multicast traffic streams through the IGMP join and report messages.

Catalyst 6500 series switches can distinguish the IGMP control traffic from the multicast data traffic. When you enable IGMP on the switch, the IGMP control traffic is redirected to the CPU for further processing. This process is performed in the hardware by the specialized ASICs. The ASICs allow the switch to snoop the IGMP control traffic with no performance penalty.

Periodically, the router sends out general queries to all VLANs. As the multicast receivers respond to the router's queries, the switch intercepts them. Only the first IGMP join (report) per VLAN and per IP multicast group is forwarded to the router. Subsequent reports for the same VLAN and group are suppressed. The switch processor creates one entry per VLAN in the Layer 2 forwarding table for each MAC group from which it receives an IGMP join request. All hosts that are interested in this multicast traffic send the join requests and are added to the port list of this forwarding table entry. If a port is disabled, it is removed from all multicast group entries.

IGMP version 3 snooping uses source-based filtering and is the industry-designated standard protocol for the hosts to signal channel subscriptions in Source Specific Multicast (SSM). The source-based filtering enables the hosts and routers to specify which source addresses should be allowed or blocked for a specific multicast group. When you enable IGMP version 3 snooping on a Catalyst 6500 series switch, the switch maintains the IGMP version 3 states based on the IGMP version 3 reports that it receives from a port on a per-group, per-VLAN basis and either allows or blocks the source traffic on that port based on the type of IGMP version 3 message that it receives. If the switch receives the IMGP version 2 snooping reports for SSM, the reports are forwarded to the MSFC2 and a system error message is generated.

Note For IGMP version 3 snooping, use Cisco IOS Release 12.1(11b)E1 or later releases on MSFC2.

IGMP Version 3 Snooping Restrictions

The following restrictions apply to IGMP version 3 snooping:

- With software release 8.3(1), it is mandatory that you run Cisco IOS Release 12.2(17d)SXB1 if you plan on using IGMP version 3 snooping with MMLS.
- IGMP version 3 snooping should be used with PIM-SSM only. For the IGMP version 3 reports that are received in non-SSM mode, IGMP version 2 snooping is performed.
- IGMP version 3 snooping is supported for INCLUDE mode only. IGMP version 3 snooping is not supported for EXCLUDE mode. IGMP version 3 reports pertaining to EXCLUDE mode are not processed but are just flooded on the VLAN.
- IGMP version 3 snooping will not discover the routers running Multicast OSPF (MOSPF) or Distance Vector Multicast Routing Protocol (DVMRP) in software release 8.3(1) and later releases.
- SPAN, RSPAN, private VLANs, and RGMP are not supported with IGMP version 3 snooping.
- IGMP version 3 snooping is supported for Single-Router Mode (SRM) only. Although Supervisor Engine 2 supports Dual-Router Mode (DRM), IGMP version 3 snooping does not support DRM.
- IGMP version 3 snooping is not supported on the systems with a Supervisor Engine 1 or Supervisor Engine 1A.
- *, G/m hardware switching for the SSM flows is supported for the ACEs that have only the permit action. The deny action should not be used for SSM. Configuring an ACE with the deny action when SSM is used may cause data loss for the IGMP version 2 snooping hosts, which operate under regular PIM sparse mode.
- A system that is configured for Layer 2 switching supports only approximately 700 ACLs.

Joining a Multicast Group

In IGMP version 2, when a host wants to join an IP multicast group, it either responds to a router query or sends an IGMP join (also known as a join message) specifying the IP multicast group to which it wants to join (for example, group 224.1.2.3). The switch hardware recognizes that the packet is an IGMP report and redirects it to the switch CPU. The switch installs a new group entry for 01-00-5e-01-02-03 and adds the host port and the router port to that entry. The switch then relays the join from the host to all multicast router ports. The designated multicast router for the segment adds the outgoing interface (OIF) to the outgoing interface list (OIL) for the group and begins forwarding the multicast traffic for 224.1.2.3 to this segment.

When a second host in this VLAN wants to join group 244.1.2.3, it sends out an IGMP join for this group. The switch hardware recognizes that this is an IGMP control packet and redirects it to the switch CPU. Because the switch already has a group entry for 01-00-5e-01-02-03 in this VLAN, it only adds the second host port to the entry. Because this is not the first host joining the group, the switch suppresses the report (the switch does not send it to the router).

The IGMP version 3 reports are sent by the hosts to the 224.0.0.22 address. The multicast router keeps a state record for each group on an interface, and the switch maintains a state record for each group on a per-VLAN basis. The state records contain the multicast IP address, the group timer, the source timer, and the filter mode as specified by the hosts. The hosts can specify one of the following filter modes:

- INCLUDE mode—In this mode, the host announces membership to a multicast group and provides a list of source IP addresses (the INCLUDE list) from which it wants to receive the traffic.
- EXCLUDE mode—In this mode, the host announces membership to a multicast group and provides a list of source IP addresses (the EXCLUDE list) from which it does not want to receive the traffic. This mode indicates that the host wants to receive the traffic only from those sources with IP addresses that are not listed in the EXCLUDE list. To receive the traffic from all sources, a host expresses EXCLUDE mode membership with an empty EXCLUDE list.

Note If IP MMLS is disabled, the IGMP compatibility mode changes to version 1 or version 2 as soon as version 1 or version 2 messages are received for a group on a VLAN (where the version 3 state previously existed for that particular group on that VLAN).

Constraining Multicast Traffic

When a host sends the multicast traffic to a group, the switch hardware does not recognize the stream as IGMP control packets. The packets are not redirected to the switch CPU. Instead, the multicast traffic is forwarded to the Media Access Control (MAC) group entry and the switch constrains the traffic to only those ports that have been added to that group entry.

The router sends the IGMP general queries. The switch floods these queries on all ports in the VLAN, and the hosts that are interested in a multicast group respond with an IGMP join for each group in which they are interested.

The switch intercepts these IGMP joins, and only the first join per VLAN and per IP multicast group is forwarded on the multicast router ports. The subsequent reports for the same VLAN and group are suppressed (not sent to the router). If you enable the switch for IGMP version 3 snooping, all joins are forwarded to the router ports.

Note If you have CGMP switches in your network, join and leave suppression does not occur. In a network that has both IGMP version 2 and CGMP switches, all join and leave messages are forwarded to the multicast routers so that the CGMP join and leave messages can be generated by the router.

Leaving a Multicast Group

In a network running IGMP version 1 or 2, the designated multicast router for a segment continues forwarding the multicast traffic to that VLAN as long as at least one host in the VLAN wishes to receive the multicast traffic. When the hosts want to leave a multicast group, they can either ignore the periodic general queries that are sent by the multicast router (IGMP version 1 host behavior), or they can send an IGMP leave (IGMP version 2 host behavior). In the systems with a Supervisor Engine 1 or 2, when the switch receives a leave message, it sends out a MAC-based general query on the port on which it received the leave message to determine if any of the devices that are connected to this port are interested in the traffic for the specific multicast group. If this port is the last port in the VLAN, the switch sends a MAC-based general query to all the ports in the VLAN. The MAC-based general queries are addressed to the Layer 2 Group Destination Address (GDA) MAC address for which the IGMP leave message was received. At Layer 3, the MAC-based general queries are addressed to 244.0.0.1 (all hosts), and in the IGMP header, the group address field is set to 0.0.0.0.

If no IGMP join is received for any of the IP multicast groups that map to the MAC multicast group address, the port is removed from the multicast forwarding entry. If the port is not the last nonmulticast-router port in the entry, the switch suppresses the IGMP leave (does not send it to the router). If the port is the last nonmulticast-router port in the entry, the IGMP leave is forwarded to the multicast router ports and the MAC group forwarding entry is removed.

When the router receives the IGMP leave, it sends several IGMP group-specific queries. If no join messages are received in response to the queries, and there are no downstream routers that are connected through that interface, the router removes the interface from the OIL for that IP multicast group entry in the multicast routing table.

IGMP Fast-Leave Processing

IGMP snooping fast-leave processing allows the switch processor to remove an interface from the port list of a forwarding-table entry without first sending out a MAC-based general query on the port. When an IGMP leave is received on a port, the port is immediately removed from the multicast forwarding entry (or the entire entry is removed).

IGMP Fast-Block Processing

IGMP version 3 supports fast-block processing. If you enable fast-block processing on the switch, the switch immediately stops forwarding the multicast packets to a port when it receives a block or exclude message from a host connected to that port.

Note Do not use the fast-leave processing feature if more than one host is connected to each port. If you enable fast-leave when more than one host is connected to a port, some hosts might be dropped inadvertently. Fast leave is supported with IGMP version 2 hosts only.

Understanding How GMRP Works

GMRP is a Generic Attribute Registration Protocol (GARP) application that provides a constrained multicast flooding facility similar to IGMP snooping. GMRP and GARP are industry-standard protocols that are defined by the IEEE. For detailed protocol operational information, refer to 802.1p.

The GMRP software components run on both the switch and on the host. (Cisco is not a source for GMRP host software.) On the host, in an IP multicast environment, you must use IGMP with GMRP; the host GMRP software spawns the Layer 2 GMRP versions of the host's Layer 3 IGMP control packets. The switch receives both the Layer 2 GMRP and the Layer 3 IGMP traffic from the host. The switch forwards the Layer 3 IGMP control packets to the router and uses the received GMRP traffic to constrain the multicasts at Layer 2 in the host's VLAN.

When a host wants to join an IP multicast group, it sends an IGMP join, which spawns a GMRP join. When the switch receives the GMRP join, it adds the port through which the join was received to the appropriate multicast group. The switch propagates the GMRP join to all the other hosts in the VLAN, one of which is typically the multicast source. When the source is multicasting to the group, the switch forwards the multicast only to the ports from which it received the join messages for the group.

The switch sends periodic GMRP queries. If a host wants to remain in a multicast group, it responds to the query and the switch does nothing. If a host does not want to remain in the multicast group, it can either send a leave message or not respond to the periodic queries from the switch. If the switch receives a leave message or receives no response from the host for the duration of the **leaveall** timer, the switch removes the host from the multicast group.

Note To use GMRP in a routed environment, enable the GMRP **forwardall** option on all ports where the routers are attached. (See the "Enabling the GMRP Forward-All Option on a Switch Port" section on page 51-22.)

Understanding How RGMP Works

Without RGMP, all multicast routers receive all multicast data traffic entering the switch. With RGMP, a multicast router can request not to receive the multicast traffic if that router has no downstream receivers for the multicast traffic. Catalyst 6500 series switches support RGMP, which enables a switch to reduce network congestion by forwarding the multicast data traffic only to those routers that are configured to receive it.

Note To use RGMP, you must enable IGMP snooping on the switch and Protocol Independent Multicast (PIM) on the routers. Only PIM sparse mode is currently supported.

All routers on the network must be RGMP capable. The RGMP-capable routers periodically send an RGMP hello message to the switch. The RGMP hello message tells the switch not to send the multicast data to the router unless an RGMP join has also been sent to the switch from that router. When an RGMP join is sent, the router is able to receive the multicast data. To learn how to set a router to receive the RGMP data, see the "RGMP-Related CLI Commands" section on page 51-34.

To stop receiving the multicast data, a router must send an RGMP leave message to the switch. To disable RGMP on a router, the router must send an RGMP bye message to the switch.

Table 51-1 provides a summary of the RGMP message types.

Table 51-1 RGMP Message Types

Description	Action
Hello	When RGMP is enabled on the router, no multicast data traffic is sent to the router by the switch unless an RGMP join is specifically sent for a group.
Bye	When RGMP is disabled on the router, all multicast data traffic is sent to the router by the switch.
Join	Multicast data traffic for a multicast MAC address from the Layer 3 group address G is sent to the router. These packets have group G in the Group Address field of the RGMP packet.
Leave	Multicast data traffic for the group G is not sent to the router. These packets have group G in the group address field of the RGMP packet.

Suppressing Multicast Traffic

On the Gigabit Ethernet ports, you can limit the amount of bandwidth to be used for the multicast traffic. Enter the **set port broadcast** command to specify a percentage of the total bandwidth to be used for the multicast traffic on the Gigabit Ethernet ports.

Rate-Limiting RPF Failure Traffic

In a redundant configuration where multiple routers connect to the same LAN segment, only one router forwards the multicast traffic from the source to the receivers on the outgoing interfaces. In this topology, only the Protocol Independent Multicast-designated forwarder (PIM-DF) forwards the data in the common VLAN, and the non-PIM-DF receives the forwarded multicast traffic. The redundant router (non-PIM-DF) must drop this traffic because it has arrived on the wrong interface and will fail the reverse path forwarding (RPF) check. The traffic that fails the RPF check is called the non-RPF traffic.

According to the multicast protocol specification, the router needs to receive the non-RPF packets for the PIM assert mechanism to function properly, so that all non-RPF packets cannot be dropped in the hardware.

PFC3A has enhanced hardware support for non-RPF packet rate limiting. On receiving a non-RPF packet, PFC3A creates a non-RPF entry (which contains source, group, and ingress interface information) in the NetFlow table, if there is no matching entry already present, and then bridges the non-RPF packet on the incoming VLAN and to MSFC3. The non-RPF packets that already have a matching NetFlow entry are only bridged on the incoming VLAN and are not sent to MSFC3.

The non-RPF entries in the NetFlow table are periodically aged out so that the non-RPF packets are leaked to MSFC3 for the PIM assert mechanism to function properly.

Rate limiting of RPF failures is enabled by default.

Enabling the Installation of Directly Connected Subnets

In PIM sparse mode, a first-hop router that is the designated router (DR) for the interface may need to encapsulate the source traffic in a PIM register message and unicast it to the rendezvous point (RP). To prevent the new sources for the group from being learned in the routing table, the (*,G) flows should remain completely hardware-switched flows. The (subnet/mask, 224/4) entries that are installed in the hardware FIB allow both (*,G) flows to remain completely hardware-switched flows, and new, directly connected sources to be learned correctly. The installation of directly connected subnets is enabled globally by default. One (subnet/mask, 224/4) is installed per PIM-enabled interface.

Enter the **show mls ip multicast connected** command to view these FIB entries.

To enable the installation of the directly connected subnets, perform this task:

Task	Command
Enable the installation of the directly connected subnets.	Router(config) # **mls ip multicast connected**

This example shows how to enable the installation of the directly connected subnets:

```
Router(config)# mls ip multicast connected
Router(config)#
```

Understanding IGMP Querier

IGMP querier enables IGMP snooping within a VLAN where PIM and IGMP are not configured because the multicast traffic does not need to be routed.

Note You must enable IGMP querier for IGMP snooping to work correctly in a VLAN in which no multicast routers are present.

When you configure IGMP querier for a VLAN, the switch sends out IGMP general query messages every 125 seconds and listens for the general query messages from the other switches. If the switch receives a general query, a querier election starts. A querier election across the switches is based either on an IP address or a MAC address. For an inbound query, if the source IP address is nonzero, the election is based on the IP address, and the switch with the lower source IP address becomes the querier. If the source IP address is zero for an inbound query, then the election is based on the source MAC address, and the switch with the lower MAC address wins the election and becomes the querier. The switch that becomes the nonquerier maintains an “other querier interval” timer. When this timer expires, the switch elects itself as the querier.

For information on enabling IGMP querier, see the “Enabling the IGMP Querier” section on page 51-16.

Redundancy for Multicast Traffic

Redundancy for multicast traffic requires the following conditions:

- Unicast routing protocol such as OSPF or EIGRP

  PIM uses RPF checks on the unicast routing table to determine the proper paths for multicast data to traverse. If a unicast routing path changes, PIM relies upon the unicast routing protocol (OSPF) to properly converge, so that the RPF checks used by PIM continue to work and show valid unicast paths to and from the source IP address of the server sourcing the multicast stream.

- PIM configured on all related Layer 3 interfaces

  The unicast routing table is used to do path selection for PIM. PIM uses RPF checks to ultimately determine the shortest path tree (SPT) between the client (receiver VLAN) and the source (multicast VLAN). Therefore, the objective of PIM is to find the shortest unicast path between the receiver subnet and the source subnet. You do not need to configure anything else for multicast when the unicast routing protocol is working as expected and PIM is configured on all the Layer 3 links associated with the unicast routing protocol.

Configuring IGMP Snooping on the Switch

IGMP snooping allows the switches to examine the IGMP packets and make the forwarding decisions based on their content.

Note Quality of service (QoS) does not support the IGMP traffic when IGMP snooping is enabled.

These sections describe how to configure IGMP snooping:

Default IGMP Snooping Configuration

Table 51-2 shows the default IGMP snooping configuration.

Note IGMP snooping is enabled by default in all supervisor engine software releases in the 7.*x* and 8.*x* release trains. It is enabled by default in software release 5.5(9) and later releases in the 5.*x* release train and in software release 6.3(1) and later releases in the 6.*x* train.

Table 51-2 IGMP Snooping Default Configuration

Feature	Default Value
IGMP snooping	Enabled
Multicast routers	None configured

IGMP Snooping Configuration Guidelines

This section describes the IGMP snooping configuration guidelines:

- There is no proxy reporting support with IGMP version 3 snooping. With IGMP version 2 snooping, only the first join and the last leave are forwarded to the router. For the group-specific (GS) queries that are initiated by the router, the switch responds with a report if at least one port is present for the group. With IGMP version 3 snooping, all reports are forwarded to the router, and the GS, group, and source-specific (GSS) queries are flooded onto the VLAN to refresh the memberships.
- At least one version 3 router must be present on the VLAN for IGMP version 3 snooping to work.
- Unlike IGMP version 2 snooping, for IGMP version 3 snooping, no permanent entries can be added that would be retained across reboots.
- IGMP version 2 snooping reports are captured and sent to the supervisor engine. The IGMP version 3 snooping reports are sent to the 224.0.0.22 address. Because snooping is not supported in this range, the reports are captured for the supervisor engine in addition to being flooded.
- With this release of IGMP version 3 snooping, the RGMP, SPAN, and RSPAN interaction is not enabled.
- IGMP querier interoperates only with IGMP version 2 snooping. Before you enable IGMP version 3 snooping, you must disable IGMP querier.

Enabling IGMP Snooping

Note You cannot enable IGMP snooping if GMRP is enabled.

To enable IGMP snooping, perform this task in privileged mode:

	Task	Command
Step 1	Enable IGMP snooping.	**set igmp enable**
Step 2	Verify that IGMP snooping is enabled.	**show igmp statistics** [*vlan*]

This example shows how to enable IGMP snooping and verify the configuration:

```
Console> (enable) set igmp enable
IGMP Snooping is enabled.
Console> (enable) show igmp statistics
IGMP enabled

IGMP statistics for vlan 1:
     Transmit:
                     General Queries: 0
               Group Specific Queries: 0
                              Reports: 0
                               Leaves: 0

     Receive:
                     General Queries: 1056
               Group Specific Queries: 0
   Group and Source Specific Queries: 2
                              Reports: 60379
                               Leaves: 0
```

```
                    Total Valid pkts: 63552
                  Total Invalid pkts: 0
                           Other pkts: 2115
            MAC-Based General Queries: 0
          Failures to add GDA to EARL: 0
               Topology Notifications: 0
                 IGMP packets dropped: 0
          IGMP Leave msgs in the list: 0
               IGMP V3 IS_IN messages: 13
               IGMP V3 IS_EX messages: 5
               IGMP V3 TO_IN messages: 0
               IGMP V3 TO_EX messages: 1
               IGMP V3 ALLOW messages: 0
               IGMP V3 BLOCK messages: 1
Console> (enable)
```

Enabling IGMP Flooding

When you disable IGMP flooding, the source traffic is never flooded in the VLAN and is sent only to the router ports. IGMP flooding is enabled by default.

To enable or disable IGMP flooding, perform this task in privileged mode:

	Task	Command
Step 1	Enable or disable IGMP flooding.	**set igmp flooding** {**enable** | **disable**}
Step 2	Display the IGMP flooding state.	**show igmp flooding**

These examples show how to enable and disable IGMP flooding:

```
Console> (enable) set igmp flooding enable
IGMP Flooding enabled (default)
Console> (enable) set igmp flooding disable
IGMP Flooding disabled
Console> (enable)
Console> (enable) show igmp flooding
Mcast flooding disabled
Console> (enable)
```

Specifying the IGMP Snooping Mode

IGMP snooping runs in either IGMP-only mode or IGMP-CGMP mode. The switch dynamically chooses either IGMP-only or IGMP-CGMP mode, depending on the traffic that is present on the network. IGMP-only mode is used in the networks with no CGMP devices. IGMP-CGMP mode is used in the networks with both IGMP and CGMP devices. Auto mode overrides the dynamic switching of the modes.

To specify the IGMP snooping mode, perform this task in privileged mode:

	Task	Command
Step 1	Specify the IGMP snooping mode.	**set igmp mode** {**igmp-only** | **igmp-cgmp** | **auto**}
Step 2	Display the IGMP snooping mode.	**show igmp mode**

This example shows how to specify the IGMP mode to IGMP-only and verify the configuration:

```
Console> (enable) set igmp mode igmp-only
IGMP mode set to igmp-only
Console> (enable) show igmp mode
IGMP Mode:                  igmp-only
IGMP Operational Mode:      igmp-only
IGMP Address Aliasing Mode: normal
Console> (enable)
```

Specifying the IGMP Leave-Query Type

You can specify the IGMP leave-query type to be used when a port receives a leave message from a host. When you specify a MAC-based general query, a leave query is sent for the exact GDA, and the version 1 or 2 hosts that have at least one membership for a group using that GDA will respond. When you specify a general query, the reports from all version 1 and 2 hosts for all groups are registered. You can also specify the auto mode. If you specify auto mode, a group-specific query is sent if there are no version 1 hosts in the network and a general query is sent if there are version 1 hosts in the network. A group-specific query provides faster network convergence.

By default, a MAC-based general query is sent when a port receives a leave message.

To specify the leave-query type, perform this task in privileged mode:

	Task	Command
Step 1	Specify the IGMP leave-query type.	**set igmp leave-query-type auto-mode | general-query | mac-gen-query**
Step 2	Display the IGMP leave-query type.	**show igmp leave-query-type**

This example shows how to set the IGMP leave-query type to a group-specific-query:

```
Console> (enable) set igmp leave-query-type auto-mode
IGMP Leave Query Type set to auto-mode
Console> (enable) show igmp leave-query-type
IGMP Leave Query Type : Group-Specific Query
Console> (enable)
```

Enabling IGMP Fast-Leave Processing

To enable IGMP fast-leave processing, perform this task in privileged mode:

	Task	Command
Step 1	Enable IGMP fast-leave processing on the switch.	**set igmp fastleave enable**
Step 2	Verify that IGMP fast-leave processing is enabled.	**show multicast protocols status**

This example shows how to enable IGMP fast-leave processing and verify the configuration:

```
Console> (enable) set igmp fastleave enable
IGMP fastleave set to enable.
Warning:Can cause disconnectivity if there are more than one host joining the
         same group per access port.
console> (enable) show multicast protocols status
IGMP disabled
IGMP fastleave enabled
IGMP V3 processing disabled
IGMP V3 fastblock feature disabled
RGMP enabled
GMRP disabled
Console> (enable)
```

Enabling IGMP Version 3 Snooping

To enable IGMP version 3 snooping, perform this task in privileged mode:

	Task	Command
Step 1	Enable IGMP version 3 snooping.	**set igmp v3-processing enable**
Step 2	Display IGMP version 3 snooping information.	**show multicast v3-group** **show multicast router**

This example shows how to enable IGMP snooping and verify the configuration:

```
Console> (enable) set igmp v3-processing enable
IGMP V3 processing enabled
Console> (enable) show multicast v3-group
Displaying V3 group information for all vlans
-------------------------------------------
(G,C): (227.1.1.1,2),  V3 state: INC
V1/V2 Compatibility mode: none (V3)
Include list: 2.2.2.6, Src timer 125 sec, Ports: 6/29 15/1
              2.2.2.5, Src timer 125 sec, Ports: 6/29 15/1
Exclude list: NULL

(G,C): (227.1.1.1,60),  V3 state: INC
V1/V2 Compatibility mode: none (V3)
Include list: 2.2.2.7, Src timer 115 sec, Ports: 13/30 15/1
              2.2.2.5, Src timer 115 sec, Ports: 13/30 15/1
              2.2.2.8, Src timer 115 sec, Ports: 13/30 15/1
Exclude list: NULL

Console> (enable) show multicast v3-group 2 227.1.1.1
 ----IGMP V3 information----
(G,C): (227.1.1.1,2),  V3 state: INC
V1/V2 Compatibility mode: none (V3)
Include list: 2.2.2.6, Src timer 125 sec, Ports: 6/29 15/1
              2.2.2.5, Src timer 125 sec, Ports: 6/29 15/1
Exclude list: NULL

Console> (enable) show multicast v3-group
Displaying V3 group information for all vlans
-------------------------------------------
(G,C): (227.1.1.1,2),  V3 state: EX
V1/V2 Compatibility mode: none (V3) Group timer: 125 sec
Include list: NULL
Exclude list: 2.2.2.6, Excluded Ports: 6/29
```

```
                2.2.2.5, Excluded Ports: 6/29

Console> (enable) show multicast router
Port           Vlan
-------------- ----------------
15/1           $  2,60

Total Number of Entries = 1
'*' - Configured
'+' - RGMP-capable
'#' - Channeled Port
'$' - IGMP-V3 Router
Console> (enable)
```

Enabling IGMP Version 3 Fast-Block Processing

To enable IGMP version 3 fast-block processing, perform this task in privileged mode:

	Task	Command
Step 1	Enable IGMP fast-block processing.	**set igmp fastblock enable**
Step 2	Verify that IGMP fast-block processing is enabled.	**show multicast protocols status**

This example shows how to enable IGMP fast-block processing and verify the configuration:

```
Console> (enable) set igmp fastblock enable
IGMP V3 fastblock enabled

Console> (enable) show multicast protocols status
IGMP enabled
IGMP fastleave disabled
IGMP V3 processing enabled
IGMP V3 fastblock feature enabled
RGMP disabled
GMRP disabled
Console> (enable)
```

Enabling IGMP Rate Limiting

Enter the **set multicast ratelimit** command to rate limit the multicast packets. The multicast packet rate limiting is disabled by default, and the default rate limit is 0 packets per second (pps).

To enable multicast rate limiting and specify a rate limit, perform this task in privileged mode:

	Task	Command
Step 1	Enable multicast rate limiting and specify a rate limit.	**set multicast ratelimit** {**disable** | **enable**} **set multicast ratelimit rate** *rate*
Step 2	Display multicast rate limiting information.	**show multicast ratelimit-info**

This example shows how to enable multicast rate limiting and specify a rate limit:

```
Console> (enable) set multicast ratelimit enable
Enabling Multicast Ratelimiting
Set a non-zero threshold rate to operationally enable multicast ratelimiting
Console> (enable) set multicast ratelimit rate 1000
Multicast ratelimit watermark rate is set to 1000 pps
Console> (enable) show multicast ratelimit-info
Multicast ratelimiting enabled
Ratelimit threshold rate: 1000 pps
VLAN  RateLimited-Since         Ratelimited-for(seconds)
----  ------------------        ----------------------
Console> (enable)
```

Enabling the IGMP Querier

Enter the IGMP querier to support IGMP snooping within a VLAN where PIM and IGMP are not configured because the multicast traffic does not need to be routed.

Note You can enable the IGMP querier on all the switches in the VLAN. One switch is elected as the querier.

To enable the IGMP querier in a VLAN, perform one of these tasks in privileged mode:

Task	Command
Enable IGMP querier on a VLAN or on all VLANs.	**set igmp querier** {**disable** | **enable**} *vlan*
Specify the time interval between the general queries sent by the switch. The default is 125 seconds.	**set igmp querier** *vlan* **qi** *val*
Specify the amount of time that the switch should wait before electing itself as the querier in the absence of general queries. The default is 300 seconds.	**set igmp querier** *vlan* **oqi** *val*
Specify an IP address for the IGMP querier. If you do not specify an IP address, the default IP address is 0.0.0.0.	**set igmp querier address** *ip_address vlan*
Display IGMP querier information.	**show igmp querier information**

This example shows how to enable the IGMP querier and display querier information:

```
Console> (enable) set igmp querier enable 4001
IGMP querier is enabled for VLAN(s) 4001
Console> (enable) set igmp querier 4001 qi 130
QI for VLAN(s) 4001 set to 130 second(s)
Console> (enable) set igmp querier address 40.1.1.1 4001
Querier Address for vlan 4001 set to 40.1.1.1
Console> (enable) show igmp querier information
VLAN Querier Address Querier State         Query Tx Count QI (sec) OQI (sec)
---- --------------- --------------------- -------------- -------- --------
4001 40.1.1.1        QUERIER               0              130      300
Console> (enable)
```

Displaying Multicast Router Information

When you enable IGMP snooping, the switch automatically learns to which ports a multicast router is connected.

To display the dynamically learned multicast router information, perform one of these tasks in privileged mode:

Task	Command
Display information on the dynamically learned and manually configured multicast router ports.	**show multicast router** [*mod/port*] [*vlan_id*]
Display information only on those multicast router ports that are learned dynamically using IGMP snooping.	**show multicast router igmp** [*mod/port*] [*vlan_id*]

This example shows how to display information on all multicast router ports (the asterisk [*] next to the multicast router on port 2/1 indicates that the entry was configured manually):

```
Console> (enable) show multicast router
Port           Vlan
-------------- ----------------
 2/1    *         @  99
 2/2              @  201
16/1     +        @  10,200-201

Total Number of Entries = 3
'*' - Configured
'+' - RGMP-capable
'#' - Channeled Port
'$' - IGMP-V3 Router
'@' - IGMP-Querier Router
Console> (enable)
```

This example shows how to display only those multicast router ports that were learned dynamically through IGMP:

```
Console> (enable) show multicast router igmp
IGMP enabled

Port       Vlan
---------  ----------------
 1/1       1
 2/1       2,99,255

Total Number of Entries = 2
'*' - Configured
'*' - Configured
'+' - RGMP-capable
'#' - Channeled Port
'$' - IGMP-V3 Router
'@' - IGMP-Querier Router
Console> (enable)
```

Displaying Multicast Group Information

To display information about the multicast groups, perform one of these tasks in privileged mode:

Task	Command
Display information about the multicast groups.	**show multicast group** [*mac_addr*] [*vlan_id*]
Display information only about the multicast groups that are learned dynamically through IGMP.	**show multicast group igmp** [*mac_addr*] [*vlan_id*]
Display the total number of multicast addresses (groups) in each VLAN.	**show multicast group count** [*vlan_id*]
Display the total number of multicast addresses (groups) in each VLAN that were learned dynamically through IGMP.	**show multicast group count igmp** [*vlan_id*]

This example shows how to display information about all multicast groups on the switch:

```
Console> (enable) show multicast group
IGMP enabled

VLAN  Dest MAC/Route Des  Destination Ports or VCs / [Protocol Type]
----  ------------------  -----------------------------------------------------
1     01-00-11-22-33-44*  2/6-12
1     01-11-22-33-44-55*  2/6-12
1     01-22-33-44-55-66*  2/6-12
1     01-33-44-55-66-77*  2/6-12

Total Number of Entries = 4
Console> (enable)
```

Displaying IGMP Snooping Statistics

To display the IGMP snooping statistics on the switch, perform this task:

Task	Command
Display the IGMP snooping statistics.	**show igmp statistics** [*vlan_id*]

This example shows how to display the IGMP snooping statistics:

```
Console> (enable) show igmp statistics
IGMP enabled

IGMP statistics for vlan 1:
     Transmit:
                     General Queries: 0
               Group Specific Queries: 0
                              Reports: 0
                               Leaves: 0

     Receive:
                     General Queries: 10
               Group Specific Queries: 0
```

```
   Group and Source Specific Queries: 0
                             Reports: 0
                              Leaves: 0
                    Total Valid pkts: 20
                  Total Invalid pkts: 0
                           Other pkts: 5
           MAC-Based General Queries: 0
         Failures to add GDA to EARL: 0
              Topology Notifications: 0
              IGMP packets dropped: 0
              IGMP Leave msgs in the list: 0
              IGMP V3 IS_IN messages: 0
              IGMP V3 IS_EX messages: 0
              IGMP V3 TO_IN messages: 0
              IGMP V3 TO_EX messages: 0
              IGMP V3 ALLOW messages: 0
              IGMP V3 BLOCK messages: 0
Console> (enable)
```

Disabling IGMP Fast-Leave Processing

To disable IGMP fast-leave processing, perform this task in privileged mode:

Task	Command
Disable IGMP fast-leave processing.	**set igmp fastleave disable**

This example shows how to disable IGMP fast-leave processing:

```
Console> (enable) set igmp fastleave disable
IGMP fastleave set to disable.
Console> (enable)
```

Disabling IGMP Snooping

To disable IGMP snooping, perform this task in privileged mode:

Task	Command
Disable IGMP snooping.	**set igmp disable**

This example shows how to disable IGMP snooping:

```
Console> (enable) set igmp disable
IGMP feature for IP multicast disabled
Console> (enable)
```

Configuring GMRP on the Switch

These sections describe how to configure the GARP Multicast Registration Protocol (GMRP):

Note For an overview of GMRP operation, see the "Understanding How GMRP Works" section on page 51-6.

GMRP Software Requirements

GMRP requires supervisor engine software release 5.2 or later releases.

Default GMRP Configuration

Table 51-3 shows the default GMRP configuration.

Table 51-3 GMRP Default Configuration

Feature	Default Value
GMRP enable state	Disabled
GMRP per-port enable state	Disabled
GMRP forward all	Disabled on all ports
GMRP registration	Normal on all ports
GARP/GMRP timers	• Join time: 200 ms • Leave time: 600 ms • Leaveall time: 10,000 ms

Enabling GMRP Globally

Note You cannot enable GMRP if IGMP snooping is enabled.

To enable GMRP globally, perform this task in privileged mode:

	Task	Command
Step 1	Enable GMRP globally.	**set gmrp enable**
Step 2	Verify the configuration.	**show gmrp configuration**

This example shows how to enable GMRP globally and verify the configuration:

```
Console> (enable) set gmrp enable
GMRP enabled.
Console> (enable) show gmrp configuration
Global GMRP Configuration:
GMRP Feature is currently enabled on this switch.
GMRP Timers (milliseconds):
Join = 200
Leave = 600
LeaveAll = 10000
Port based GMRP Configuration:
Port                                        GMRP Status Registration ForwardAll
------------------------------------------- ----------- ------------ ----------
1/1-2,3/1,6/1-48,7/1-24                     Enabled     Normal       Disabled
Console> (enable)
```

Enabling GMRP on Individual Switch Ports

Note You can change the per-port GMRP configuration regardless of whether GMRP is enabled globally. However, GMRP will not function on any ports until you enable it globally. For information on configuring GMRP globally, see the "Enabling GMRP Globally" section on page 51-21.

To enable GMRP on the individual switch ports, perform this task in privileged mode:

	Task	Command
Step 1	Enable GMRP on an individual switch port.	**set port gmrp enable** *mod/port*
Step 2	Verify the configuration.	**show gmrp configuration**

This example shows how to enable GMRP on port 6/12 and verify the configuration:

```
Console> (enable) set port gmrp enable 6/12
GMRP enabled on port 6/12.
Console> (enable) show gmrp configuration
Global GMRP Configuration:
GMRP Feature is currently enabled on this switch.
GMRP Timers (milliseconds):
Join = 200
Leave = 600
LeaveAll = 10000
```

```
Port based GMRP Configuration:
Port                                        GMRP Status Registration ForwardAll
------------------------------------------- ----------- ------------ ----------
1/1-2,3/1,6/1-9,6/12,6/15-48,7/1-24         Enabled     Normal       Disabled
6/10-11,6/13-14                             Disabled    Normal       Disabled
Console> (enable)
```

Disabling GMRP on Individual Switch Ports

Note You can change the per-port GMRP configuration regardless of whether GMRP is enabled globally. However, GMRP will not function on any ports until you enable it globally. For information on configuring GMRP globally, see the "Enabling GMRP Globally" section on page 51-21.

To disable GMRP on the individual switch ports, perform this task in privileged mode:

	Task	Command
Step 1	Disable GMRP on the individual switch ports.	**set port gmrp disable** *mod/port*
Step 2	Verify the configuration.	**show gmrp configuration**

This example shows how to disable GMRP on ports 6/10–14 and verify the configuration:

```
Console> (enable) set port gmrp disable 6/10-14
GMRP disabled on ports 6/10-14.
Console> (enable) show gmrp configuration
Global GMRP Configuration:
GMRP Feature is currently enabled on this switch.
GMRP Timers (milliseconds):
Join = 200
Leave = 600
LeaveAll = 10000
Port based GMRP Configuration:
Port                                        GMRP Status Registration ForwardAll
------------------------------------------- ----------- ------------ ----------
1/1-2,3/1,6/1-9,6/15-48,7/1-24              Enabled     Normal       Disabled
6/10-14                                     Disabled    Normal       Disabled
Console> (enable)
```

Enabling the GMRP Forward-All Option on a Switch Port

When you enable the GMRP forward-all option on a port, a copy of all multicast traffic that is registered on the switch is forwarded to that port. Enable the forward-all option on any port that is connected to a router that needs to receive any multicasts (routers do not support GMRP and cannot send GMRP join messages). The forward-all option can also be used to forward all registered multicast traffic to a port with a network analyzer or probe attached.

To enable the GMRP forward-all option on a switch port, perform this task in privileged mode:

Task	Command
Enable the GMRP forward-all option on a switch port.	**set gmrp fwdall enable** *mod/port*

This example shows how to enable the GMRP forward-all option on port 1/1:

```
Console> (enable) set gmrp fwdall enable 1/1
GMRP Forward All groups option enabled on port 1/1.
Console> (enable)
```

Disabling the GMRP Forward-All Option on a Switch Port

To disable the GMRP forward-all option on a switch port, perform this task in privileged mode:

Task	Command
Disable the GMRP forward-all option on a switch port.	**set gmrp fwdall disable** *mod/port*

This example shows how to disable the GMRP forward-all option on port 1/1:

```
Console> (enable) set gmrp fwdall disable 1/1
GMRP Forward All groups option disabled on port 1/1.
Console> (enable)
```

Configuring GMRP Registration

These sections describe how to configure the GMRP registration modes on the switch ports:

- Setting Normal Registration, page 51-23
- Setting Fixed Registration, page 51-24
- Setting Forbidden Registration, page 51-24

Setting Normal Registration

Configuring a switch port in **normal** registration mode allows dynamic GMRP multicast registration and deregistration on the port. Normal mode is the default on all switch ports.

To set the normal registration on a switch port, perform this task in privileged mode:

	Task	Command
Step 1	Set the normal registration on a switch port.	**set gmrp registration normal** *mod/port*
Step 2	Verify the configuration.	**show gmrp configuration**

This example shows how to set normal registration on port 2/10:

```
Console> (enable) set gmrp registration normal 2/10
GMRP Registration is set normal on port 2/10.
Console> (enable)
```

Setting Fixed Registration

When you configure a switch port in **fixed** registration mode, all the multicast groups that are currently registered on all ports are registered on the port, but the port ignores any subsequent registrations or deregistrations on other ports. A switch port in fixed registration mode continues to register the multicast groups that are specific to the port. You must return the port to **normal** registration mode to deregister the multicast groups on the port.

To set the fixed registration on a switch port, perform this task in privileged mode:

	Task	Command
Step 1	Set the fixed registration on a switch port.	**set gmrp registration fixed** *mod/port*
Step 2	Verify the configuration.	**show gmrp configuration**

This example shows how to set the fixed registration on port 2/10 and verify the configuration:

```
Console> (enable) set gmrp registration fixed 2/10
GMRP Registration is set fixed on port 2/10.
Console> (enable) show gmrp configuration
Global GMRP Configuration:
GMRP Feature is currently enabled on this switch.
GMRP Timers (milliseconds):
Join = 200
Leave = 600
LeaveAll = 10000
Port based GMRP Configuration:
GMRP-Status Registration ForwardAll Port(s)
----------- ------------ ---------- ------------------------------------------
Enabled     Normal       Disabled   1/1-4
                                    2/1-9,2/11-48
                                    3/1-24
                                    5/1
Enabled     Fixed        Disabled   2/10
Console> (enable)
```

Setting Forbidden Registration

Setting a switch port in **forbidden** registration mode deregisters all GMRP multicasts and prevents any further GMRP multicast registration on the port.

To set the forbidden registration on a switch port, perform this task in privileged mode:

	Task	Command
Step 1	Set the forbidden registration on a switch port.	**set gmrp registration forbidden** *mod/port*
Step 2	Verify the configuration.	**show gmrp configuration**

This example shows how to set the forbidden registration on port 2/10 and verify the configuration:

```
Console> (enable) set gmrp registration forbidden 2/10
GMRP Registration is set forbidden on port 2/10.
Console> (enable) show gmrp configuration
Global GMRP Configuration:
GMRP Feature is currently enabled on this switch.
GMRP Timers (milliseconds):
Join = 200
```

```
Leave = 600
LeaveAll = 10000
Port based GMRP Configuration:
GMRP-Status Registration ForwardAll Port(s)
----------- ------------ ---------- ------------------------------------------
Enabled     Normal       Disabled   1/1-4
                                    2/1-9,2/11-48
                                    3/1-24
                                    5/1
Enabled     Forbidden    Disabled   2/10
Console> (enable)
```

Setting the GARP Timers

Note The commands **set gmrp timer** and **show gmrp timer** are aliases for **set garp timer** and **show garp timer**. The aliases may be used if desired.

Note Modifying the GARP timer values affects the behavior of all GARP applications running on the switch, not just GMRP. (For example, GVRP uses the same timers.)

Note The only ports that send out the GMRP leaveall messages are the ports that have previously received the GMRP joins.

You can modify the default GARP timer values on the switch.

When setting the timer values, the value for **leave** must be equal to or greater than three times the **join** value (**leave >= join** * 3). The value for **leaveall** must be greater than the value for **leave** (**leaveall > leave**). The more registered attributes on the switch, the greater you should configure the difference between the **leave** value and the **join** value.

For better performance on the switches with many registered multicast groups, increase the timer values to the order of seconds.

If you attempt to set a timer value that does not adhere to these rules, an error is returned. For example, if you set the **leave** timer to 600 ms and you attempt to configure the **join** timer to 350 ms, an error is returned. Set the **leave** timer to at least 1050 ms, and then set the **join** timer to 350 ms.

Caution Set the same GARP timer values on all Layer 2-connected devices. If the GARP timers are set differently on the Layer 2-connected devices, the GARP applications (for example, GMRP and GVRP) do not operate successfully.

To set the GARP timer values, perform this task in privileged mode:

	Task	Command
Step 1	Set the GARP timer values.	**set garp timer** {**join** | **leave** | **leaveall**} *timer_value*
Step 2	Verify the configuration.	**show garp timer**

This example shows how to set the GARP timers and verify the configuration:

```
Console> (enable) set garp timer leaveall 12000
GMRP/GARP leaveAll timer value is set to 12000 milliseconds.
Console> (enable) set garp timer leave 650
GMRP/GARP leave timer value is set to 650 milliseconds.
Console> (enable) set garp timer join 300
GMRP/GARP join timer value is set to 300 milliseconds.
Console> (enable) show garp timer
Timer     Timer Value (milliseconds)
--------  --------------------------
Join      300
Leave     650
LeaveAll  12000
Console> (enable)
```

Displaying GMRP Statistics

To display the GMRP statistics on the switch, perform this task in privileged mode:

Task	Command
Display the GMRP statistics.	**show gmrp statistics** [*vlan_id*]

This example shows how to display the GMRP statistics for VLAN 23:

```
Console> show gmrp statistics 23
GMRP Statistics for vlan <23>:
Total valid GMRP Packets Received:500
Join Empties:200
Join INs:250
Leaves:10
Leave Alls:35
Empties:5
Fwd Alls:0
Fwd Unregistered:0
Total valid GMRP Packets Transmitted:600
Join Empties:200
Join INs:150
Leaves:45
Leave Alls:200
Empties:5
Fwd Alls:0
Fwd Unregistered:0
Total valid GMRP Packets Received:0
Total GMRP packets dropped:0
Total GMRP Registrations Failed:0
Console>
```

Clearing GMRP Statistics

To clear all GMRP statistics on the switch, perform this task in privileged mode:

Task	Command
Clear the GMRP statistics.	**clear gmrp statistics** {*vlan_id* | **all**}

This example shows how to clear the GMRP statistics for all VLANs:

```
Console> (enable) clear gmrp statistics all
Console> (enable)
```

Disabling GMRP Globally on the Switch

To disable GMRP globally on the switch, perform this task in privileged mode:

Task	Command
Disable GMRP globally on the switch.	**set gmrp disable**

This example shows how to disable GMRP globally on the switch:

```
Console> (enable) set gmrp disable
GMRP disabled.
Console> (enable)
```

Configuring Multicast Router Ports and Group Entries on the Switch

These sections describe how to specify the multicast router ports manually and configure the multicast group entries:

Specifying Multicast Router Ports

When you enable IGMP snooping, the switch automatically learns to which ports a multicast router is connected. However, you can manually specify the multicast router ports.

To specify the multicast router ports manually, perform this task in privileged mode:

	Task	Command
Step 1	Manually specify a multicast router port.	**set multicast router** *mod/port*
Step 2	Verify the configuration.	**show multicast router** [**igmp** | **rgmp**][*mod/port*] [*vlan_id*]

This example shows how to specify a multicast router port manually and verify the configuration (the asterisk [*] next to the multicast router on port 2/2 indicates that the entry was configured manually):

```
Console> (enable) set multicast router 2/2
Port 2/2 added to multicast router port list.
console> (enable) show multicast router
Port           Vlan
-------------- ----------------
 2/2    *            50
 8/48             @  10
16/1              @  200-201

Total Number of Entries = 3
'*' - Configured
'+' - RGMP-capable
'#' - Channeled Port
'$' - IGMP-V3 Router
'@' - IGMP-Querier Router
Console> (enable)
```

Configuring Multicast Groups

To configure a multicast group manually, perform this task in privileged mode:

Note With software release 7.1(1) and later releases, the maximum number of Layer 2 multicast entries is 15488.

	Task	Command
Step 1	Add one or more multicast MAC addresses to the CAM table.	**set cam** {**static** | **permanent**} *multicast_mac mod/port* [*vlan*]
Step 2	Verify the multicast group configuration.	**show multicast group** [*mac_addr*] [*vlan_id*]

This example shows how to configure the multicast groups manually and verify the configuration (the asterisks indicate that the entry was manually configured):

```
Console> (enable) set cam static 01-00-11-22-33-44 2/6-12
Static multicast entry added to CAM table.
Console> (enable) set cam static 01-11-22-33-44-55 2/6-12
Static multicast entry added to CAM table.
Console> (enable) set cam static 01-22-33-44-55-66 2/6-12
Static multicast entry added to CAM table.
Console> (enable) set cam static 01-33-44-55-66-77 2/6-12
Static multicast entry added to CAM table.
Console> (enable) show multicast group
IGMP disabled

VLAN  Dest MAC/Route Des  Destination Ports or VCs / [Protocol Type]
----  ------------------  -----------------------------------------------------
1     01-00-11-22-33-44*  2/6-12
1     01-11-22-33-44-55*  2/6-12
1     01-22-33-44-55-66*  2/6-12
1     01-33-44-55-66-77*  2/6-12

Total Number of Entries = 4
Console> (enable)
```

Clearing Multicast Router Ports

To clear the manually configured multicast router ports, perform one of these tasks in privileged mode:

Task	Command
Clear the specific, manually configured multicast router ports.	**clear multicast router** *mod/port*
Clear all manually configured multicast router ports.	**clear multicast router all**

This example shows how to clear a manually configured multicast router port:

```
Console> (enable) clear multicast router 2/12
Port 2/12 cleared from multicast router port list.
Console> (enable)
```

Clearing Multicast Group Entries

To clear the manually configured multicast group entries from the CAM table, perform this task in privileged mode:

Task	Command
Clear a multicast group entry from the CAM table.	**clear cam** *mac_addr* [*vlan*]

This example shows how to clear a multicast group entry from the CAM table:

```
Console> (enable) clear cam 01-11-22-33-44-55 1
CAM entry cleared.
Console> (enable)
```

Understanding How RGMP Works

RGMP constrains the multicast traffic that exits the switch through the ports to which only the disinterested multicast routers are connected. Catalyst 6500 series switches support RGMP, which enables a switch to reduce network congestion by forwarding the multicast data traffic to only those routers that are configured to receive it.

Note To use RGMP, you must enable IGMP snooping on the switch. IGMP snooping constrains the multicast traffic that exits through the switch ports to which the hosts are connected. IGMP snooping does not constrain the traffic that exits through the ports to which one or more multicast routers are connected.

Note You must enable PIM on all routers and switches for RGMP to work. Currently, only PIM sparse mode is supported.

All routers on the network must be RGMP capable. RGMP-capable routers send an RGMP hello message to the switch periodically. The RGMP hello message tells the switch not to send the multicast data to the router unless an RGMP join message has also been sent to the switch from that router. When an RGMP join message is sent, the router is able to receive the multicast data. To learn how to set a router to receive the RGMP data, see the "RGMP-Related CLI Commands" section on page 51-34.

To stop receiving the multicast data, a router must send an RGMP leave message to the switch. To disable RGMP on a router, the router must send an RGMP bye message to the switch.

Table 51-4 provides a summary of the RGMP packet types.

Table 51-4 RGMP Packet Types

Description	Action
Hello	When RGMP is enabled on the router, no multicast data traffic is sent to the router by the switch unless an RGMP join is specifically sent for a group.
Bye	When RGMP is disabled on the router, all multicast data traffic is sent to the router by the switch.
Join	Multicast data traffic for a multicast MAC address from the Layer 3 group address G is sent to the router. These packets have group G in the Group Address field of the RGMP packet.
Leave	Multicast data traffic for the group G is not sent to the router. These packets have group G in the group address field of the RGMP packet.

These restrictions apply to RGMP:

- Sparse mode only—RGMP supports PIM sparse mode only. RGMP does not support PIM dense mode. RGMP explicitly supports the two AutoRP groups in dense mode by not restricting the traffic to those groups but by flooding it to all router ports. For this reason, you should configure PIM sparse-dense mode. If you configure groups other than the AutoRP groups for dense mode, their traffic will not be correctly forwarded through the router ports that have been enabled for RGMP.
- To effectively constrain the multicast traffic with RGMP, connect the RGMP-enabled routers to separate the ports on the RGMP-enabled switches.
- RGMP constrains only the traffic that exits through the ports on which it detects an RGMP-enabled router. If a non-RGMP enabled router is detected on a port, that port receives all multicast traffic.
- RGMP does not support the directly connected sources in the network. A directly connected source will send the traffic into the network without signaling this through RGMP or PIM. This traffic will not be received by an RGMP-enabled router unless the router already requested receipt of that group through RGMP. This restriction applies to the hosts and to the functions in the routers that source the multicast traffic, such as the **ping** and **mtrace** commands, and the multicast applications that source the multicast traffic, such as UDPTN.
- RGMP supports the directly connected receivers in the network. The traffic to these receivers will be constrained by IGMP snooping, or if the receiver is a router, by PIM and RGMP. CGMP is not supported in the networks where RGMP is enabled on the routers. Enabling RGMP and CGMP on a router interface is mutually exclusive. If RGMP is enabled on an interface, CGMP is silently disabled or vice versa.

- The following properties of RGMP are the same as for IGMP snooping:
  - RGMP constrains the traffic that is based on the multicast group, not on the sender's IP address.
  - If spanning-tree topology changes occur in the network, the state is not flushed as it is with CGMP.
  - RGMP does not constrain the traffic for multicast groups 224.0.0.x (x = 0...255), which allow use of the PIMv2 bootstrap router (BSR) in an RGMP-controlled network.
  - RGMP in the Cisco switches operates on the MAC addresses, not on the IP multicast addresses. Because multiple IP multicast addresses can map to one MAC address (see RFC 1112), RGMP cannot differentiate between the IP multicast groups that might map to a MAC address.
  - The capability of the switch to constrain the traffic is limited by its content addressable memory (CAM) table capacity.

Configuring RGMP on the Switch

These sections describe the commands for configuring RGMP:

Configuring RGMP on the Supervisor Engine

These sections describe the commands for configuring RGMP:

Default RGMP Configuration

RGMP is disabled by default.

Enabling and Disabling RGMP

Note To enable RGMP, you must have IGMP snooping enabled.

To enable or disable RGMP, perform one of these tasks in privileged mode:

Task	Command
Enable RGMP.	**set rgmp enable**
Disable RGMP.	**set rgmp disable**

This example shows how to enable RGMP:

```
Console> (enable) set rgmp enable
RGMP enabled.
Console> (enable)
```

This example shows how to disable RGMP:

```
Console> (enable) set rgmp disable
RGMP disabled.
Console> (enable)
```

Displaying RGMP Group Information

Use these commands to display all multicast groups that were joined by one or more RGMP-capable routers and to display the count of multicast groups that were joined by one or more RGMP-capable routers.

To display RGMP group information, perform one of these tasks in privileged mode:

Task	Command
Display all multicast groups that were joined by one or more RGMP-capable routers.	**show rgmp group** [*mac_addr*] [*vlan_id*]
Display the count of multicast groups that were joined by one or more RGMP-capable routers.	**show rgmp group count** [*vlan_id*]

This example shows how to display RGMP group information:

```
Console> (enable) show rgmp group
VlanDest MAC/Route DesRGMP Joined Router Ports
--------------------------------------------------------------------------------
1   01-00-5e-00-01-285/1,5/15
1   01-00-5e-01-01-015/1
2   01-00-5e-27-23-70*3/1, 5/1
Total Number of Entries = 3
'*' - Configured
Console> (enable)

Console> (enable) show rgmp group count 1
Total Number of Entries = 2
```

Displaying RGMP VLAN Statistics

To display the RGMP statistics for a given VLAN, perform this task in privileged mode:

Task	Command
Display the RGMP statistics for a specified VLAN.	**show rgmp statistics** [*vlan*]

This example shows how to display the RGMP statistics for a specified VLAN:

```
Console> (enable) show rgmp statistics 23
RGMP enabled
RGMP Statistics for vlan <23>:
Receive:
Valid pkts:20
Hellos:10
Joins:5
Leaves:5
Byes:0
Discarded:0
Transmit:
Total Pkts:10
Failures:0
Hellos:10
Joins:0
Leaves:0
Byes:0
Console> (enable)
```

Displaying RGMP-Capable Router Ports

This command displays the detected RGMP-capable router ports. A "+" in front of the port indicates that it is an RGMP-capable router.

To display the RGMP-capable router ports, perform this task in privileged mode:

Task	Command
Display the RGMP-capable router ports.	**show multicast router** [**igmp** | **rgmp**] [*mod/port*] [*vlan_id*]

This example shows how to display the ports that are connected to the RGMP-capable routers:

```
Console> (enable) show multicast router
Port           Vlan
-------------- ----------------
 2/2     +        @  40
 8/48            @  10
16/1     +        @  200-201

Total Number of Entries = 3
'*' - Configured
'+' - RGMP-capable
'#' - Channeled Port
'$' - IGMP-V3 Router
'@' - IGMP-Querier Router
Console> (enable)
```

This example shows how to display only the RGMP-capable router ports:

```
Console> (enable) show multicast router rgmp
Port            Vlan
--------------- ----------------
 2/2     +        @  40
16/1     +        @  200

Total Number of Entries = 2
'*' - Configured
'+' - RGMP-capable
'#' - Channeled Port
'$' - IGMP-V3 Router
'@' - IGMP-Querier Router
Console> (enable)
```

Clearing RGMP Statistics

This command clears the stored RGMP statistics.

To clear the RGMP statistics, perform this task in privileged mode:

Task	Command
Clear the RGMP statistics.	**clear rgmp statistics**

This example shows how to clear the RGMP statistics:

```
Console> (enable) clear rgmp statistics
RGMP statistics cleared.
Console> (enable)
```

RGMP-Related CLI Commands

This command enables or disables the RGMP-related commands from the router.

To enable or disable RGMP, perform one of these tasks in configuration mode:

Task	Command
Enable RGMP.	Router(config)# **ip rgmp**
Disable RGMP.	Router(config)# **no ip rgmp**

This command enables or disables RGMP debugging.

To enable or disable RGMP debugging, perform one of these tasks in privileged mode:

Task	Command
Enable RGMP debugging.	Router# **debug ip rgmp** [*group-name* | *group-address*]
Disable RGMP debugging.	Router# **no debug ip rgmp** [*group-name* | *group-address*]

Configuring RGMP on the MSFC

To configure RGMP on a VLAN interface on the MSFC, perform this task:

	Task	Command
Step 1	Access VLAN interface configuration mode.	Router(config)# **interface vlan** *vlan_ID*
Step 2	Enable RGMP.	Router(config-if)# **ip rgmp**

You can use the **debug ip rgmp** command to monitor RGMP on the MSFC.

Displaying the Multicast Protocol Status

This command displays the status (enabled or disabled) of the Layer 2 multicast protocols on the switch.

To display the multicast protocol status, perform this task in privileged mode:

Task	Command
Display the multicast protocol status.	**show multicast protocols status**

This example shows how to display the multicast protocol status:

```
Console> (enable) show multicast protocols status
IGMP enabled
IGMP fastleave disabled
IGMP V3 processing disabled
IGMP V3 fastblock feature disabled
RGMP disabled
GMRP disabled
Console> (enable)
```

Understanding How Bidirectional PIM Works

Supervisor Engine 720 supports the hardware forwarding of the bidirectional Protocol Independent Multicast (PIM) groups. To support the bidirectional PIM groups, Supervisor Engine 720 implements a new mode called designated forwarder (DF) mode. The designated forwarder is the router that is elected to forward the packets to and from a segment for a bidirectional PIM group. In DF mode, the supervisor engine accepts the packets from the reverse path forwarding (RPF) interface and from the DF interface.

When the supervisor engine is forwarding the bidirectional PIM groups, the RPF interface is always included in the outgoing interface list of (*,G) entry, and the DF interfaces are included depending on IGMP/PIM joins.

If the route to the RP becomes unavailable, the group is changed to dense mode. If the RPF link to the RP becomes unavailable, the bidirectional flow is removed from the hardware FIB.

Configuring Bidirectional PIM on the Switch

These sections show how to configure bidirectional PIM and display the bidirectional PIM configuration information and statistics:

Configuring Bidirectional PIM

To configure bidirectional PIM, perform these steps:

Step 1 Enable bidirectional PIM globally.

Step 2 Configure the rendezvous point for the bidirectional group.

These steps are described in detail in the following sections.

Enabling or Disabling Bidirectional PIM Globally

To enable or disable bidirectional PIM, perform one of these tasks:

Task	Command
Enable bidirectional PIM globally on the switch.	Router(config)# **ip pim bidir-enable**
Disable bidirectional PIM globally on the switch.	Router(config)# [**no**] **ip pim bidir-enable**

This example shows how to enable bidirectional PIM on the switch:

```
Router(config)# ip pim bidir-enable
Router(config)#
```

This example shows how to disable bidirectional PIM on the switch:

```
Router(config)# no ip pim bidir-enable
Router(config)#
```

Configuring the Rendezvous Point for Bidirectional Groups

Note The traffic flow for the groups mapping to only four bidirectional rendezvous points (RPs) is hardware switched. The traffic to the rest of the groups is software forwarded.

To configure the rendezvous point for a bidirectional group statically, perform this task:

	Task	Command
Step 1	Statically configure the IP address of the rendezvous point for the group. When you specify the **override** keyword, the static rendezvous point is used.	Router(config)# **ip pim rp-adress** *ip_address access-list* [**override**]
Step 2	Configure an access list.	Router(config)# **access-list** *access-list* **permit** | **deny** *ip_address*
Step 3	Configure the system to use Auto-RP to configure groups for which the router will act as an RP.	Router(config)# **ip pim send-rp-announce** type number **scope** ttl-value [**group-list** access-list] [**interval** seconds] [**bidir**]
Step 4	Configure a standard IP access list.	Router(config)# **ip access-list standard** *access-list-name* **permit** | **deny** *ip_address*
Step 5	Enable MLS IP multicast.	Router(config)# **mls ip multicast**

This example shows how to configure a static rendezvous point for a bidirectional group:

```
Router(config)# ip pim rp-address 10.0.0.1 10 bidir override
Router(config)# access-list 10 permit 224.1.0.0 0.0.255.255
Router(config)# ip pim send-rp-announce Loopback0 scope 16 group-list c21-rp-list-0 bidir
Router(config)# ip access-list standard c21-rp-list-0 permit 230.31.31.1 0.0.255.255
```

Setting the Bidirectional PIM Scan Interval

You can specify the interval between the bidirectional PIM RP Reverse Path Forwarding (RPF) scans.

To set the bidirectional RP RPF scan interval, perform one of these tasks:

Task	Command
Set the bidirectional RP RPF scan interval; valid values are from 1 to 1000 seconds. The default is 10 seconds.	Router(config)# **mls ip multicast bidir gm-scan-interval** *interval*
Restore the default.	Router(config)# **no mls ip multicast bidir gm-scan-interval**

This example shows how to set the bidirectional RP RPF scan interval:

```
Router(config)# mls ip multicast bidir gm-scan-interval 30
Router(config)#
```

This example shows how to restore the default bidirectional RP RPF scan interval:

```
Router(config)# no mls ip multicast bidir gm-scan-interval
Router(config)#
```

Displaying Bidirectional PIM Information

To display the bidirectional PIM information, perform one of these tasks:

Task	Command
Display the mappings between the PIM groups and the rendezvous points and show the learned rendezvous points in use.	Router# **show ip pim rp mapping** [**in-use**]
Display the PIM group to the active rendezvous-point mappings.	Router# **show mls ip multicast rp-mapping** [*rp-address*]
Display information based on the group/mask ranges in the RP-mapping cache.	Router# **show mls ip multicast rp-mapping gm-cache**
Display information based on the DF list in the RP-mapping cache.	Router# **show mls ip multicast rp-mapping df-cache**
Display the bidirectional PIM information.	Router# **show mls ip multicast bidir**
Display information about the multicast routing table.	Router# **show ip mroute**

This example shows how to display information about the PIM group and rendezvous-point mappings:

```
Router# show ip pim rp mapping
PIM Group-to-RP Mappings
This system is an RP (Auto-RP)
This system is an RP-mapping agent
Group(s) 230.31.0.0/16
  RP 60.0.0.60 (?), v2v1, bidir
    Info source:60.0.0.60 (?), elected via Auto-RP
         Uptime:00:03:47, expires:00:02:11
  RP 50.0.0.50 (?), v2v1, bidir
    Info source:50.0.0.50 (?), via Auto-RP
         Uptime:00:03:04, expires:00:02:55
  RP 40.0.0.40 (?), v2v1, bidir
    Info source:40.0.0.40 (?), via Auto-RP
         Uptime:00:04:19, expires:00:02:38
```

This example shows how to display information in the IP multicast routing table that is related to bidirectional PIM:

```
Router# show ip mroute bidirectional
    (*, 225.1.3.0), 00:00:02/00:02:57, RP 3.3.3.3, flags:BC
      Bidir-Upstream:GigabitEthernet2/1, RPF nbr 10.53.1.7, RPF-MFD
      Outgoing interface list:
        GigabitEthernet2/1, Bidir-Upstream/Sparse-Dense, 00:00:02/00:00:00,H
        Vlan30, Forward/Sparse-Dense, 00:00:02/00:02:57, H

    (*, 225.1.2.0), 00:00:04/00:02:55, RP 3.3.3.3, flags:BC
      Bidir-Upstream:GigabitEthernet2/1, RPF nbr 10.53.1.7, RPF-MFD
      Outgoing interface list:
        GigabitEthernet2/1, Bidir-Upstream/Sparse-Dense, 00:00:04/00:00:00,H
        Vlan30, Forward/Sparse-Dense, 00:00:04/00:02:55, H

    (*, 225.1.4.1), 00:00:00/00:02:59, RP 3.3.3.3, flags:BC
      Bidir-Upstream:GigabitEthernet2/1, RPF nbr 10.53.1.7, RPF-MFD
      Outgoing interface list:
        GigabitEthernet2/1, Bidir-Upstream/Sparse-Dense, 00:00:00/00:00:00,H
        Vlan30, Forward/Sparse-Dense, 00:00:00/00:02:59, H
```

This example show how to display information that is related to a specific multicast route:

```
Router# show ip mroute 239.1.1.2 4.4.4.4
IP Multicast Routing Table
Flags:D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report, Z - Multicast Tunnel
       Y - Joined MDT-data group, y - Sending to MDT-data group
Outgoing interface flags:H - Hardware switched
 Timers:Uptime/Expires
 Interface state:Interface, Next-Hop or VCD, State/Mode

(4.4.4.4, 239.1.1.2), 1d02h/00:03:20, flags:FTZ
 Incoming interface:Loopback0, RPF nbr 0.0.0.0, Partial-SC
  Outgoing interface list:
    Vlan10, Forward/Sparse-Dense, 1d02h/00:02:39 (ttl-threshold 5)
```

This example shows how to display the entries for a specific multicast group address:

```
Router# show mls ip multicast group 230.31.31.1
Multicast hardware switched flows:
(*, 230.31.31.1) Incoming interface:Vlan611, Packets switched:1778
Hardware switched outgoing interfaces:Vlan131 Vlan151 Vlan415 Gi4/16 Vlan611
RPF-MFD installed
```

This example shows how to display the PIM group to the active rendezvous-point mappings:

```
Router# show mls ip multicast rp-mapping
State:H - Hardware Switched, I - Install Pending, D - Delete Pending, Z - Zombie

RP Address      State       RPF            DF-count    GM-count
60.0.0.60       H           Vl611                 4           1
```

This example shows how to display information that is based on the group/mask ranges in the RP-mapping cache:

```
Router# show mls ip multicast rp-mapping gm-cache
State:H - Hardware Switched, I - Install Pending, D - Delete Pending,
       Z - Zombie

RP Address     State       Group           Mask            State         Packet/Byte-count
60.0.0.60      H           230.31.0.0      255.255.0.0     H             100/6400
```

This example shows how to display information about the specific MLS IP multicasting groups:

```
Router# show mls ip multicast rp-mapping df-cache
State:H - Hardware Switched, I - Install Pending, D - Delete Pending, Z - Zombie

RP Address      State        DF             State
60.0.0.60       H           Vl131           H
60.0.0.60       H           Vl151           H
60.0.0.60       H           Vl415           H
60.0.0.60       H           Gi4/16          H
```

CHAPTER 52

Configuring QoS

This chapter describes how to configure quality of service (QoS) on the Catalyst 6500 series switches and includes the configuration information that is required to support Common Open Policy Service (COPS) and Resource ReSerVation Protocol (RSVP).

Note

- For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.
- For information on using automatic QoS, see Chapter 53, "Using Automatic QoS."

You can configure QoS using one of the following:

- SNMP
- COPS protocol
- RSVP null service template and receiver proxy functionality
- Command-line interface (CLI)

This chapter consists of these sections:

Understanding How QoS Works

Note Throughout this publication and all Catalyst 6500 series publications, the term *QoS* refers to the QoS feature as implemented on the Catalyst 6500 series switch.

Typically, networks operate on a *best-effort* delivery basis, which means that all traffic has equal priority and an equal chance of being delivered in a timely manner. When congestion occurs, all traffic has an equal chance of being dropped.

QoS on the Catalyst 6500 series switches selects network traffic, prioritizes it according to its relative importance, and provides priority-indexed treatment through congestion avoidance techniques. QoS makes network performance more predictable and bandwidth utilization more effective.

QoS sets the Layer 2 and Layer 3 values in the network traffic to a configured value or to a value that is based on the received Layer 2 or Layer 3 values. The IP traffic retains the Layer 3 value when it leaves the switch.

With PFC3, you can configure QoS for both the ingress and egress traffic. You can configure QoS per-port or per-VLAN for the ingress traffic. You can configure QoS only per VLAN for the egress traffic.

With other hardware, you can configure QoS per port or per VLAN for the ingress traffic.

These sections describe QoS:

QoS Terminology

This section defines some QoS terminology:

- *Packets* carry traffic at Layer 3.
- *Frames* carry traffic at Layer 2. The Layer 2 frames carry the Layer 3 packets.
- *Labels* are prioritization values that are carried in the packets and the frames:
  - Layer 2 class of service (CoS) values range between zero for low priority and seven for high priority:

    The Layer 2 Inter-Switch Link (ISL) frame headers have a 1-byte User field that carries an IEEE 802.1p CoS value in the three least significant bits.

    The Layer 2 802.1Q frame headers have a 2-byte Tag Control Information field that carries the CoS value in the three most significant bits, which are called the User Priority bits.

    The other frame types cannot carry the CoS values.

    Note On the ports that are configured as ISL trunks, all traffic is in ISL frames. On the ports that are configured as 802.1Q trunks, all traffic is in the 802.1Q frames except for the traffic in the native VLAN.

  - Layer 3 IP precedence values—The IP version 4 specification defines the three most significant bits of the 1-byte Type of Service (ToS) field as the IP precedence. The IP precedence values range between zero for low priority and seven for high priority.
  - Layer 3 differentiated services code point (DSCP) values—The Internet Engineering Task Force (IETF) defines the six most significant bits of the 1-byte ToS field as the DSCP. The priority that is represented by a particular DSCP value is configurable. The DSCP values range between 0 and 63 (for more information, see the "Configuring the DSCP Value Maps" section on page 52-73).

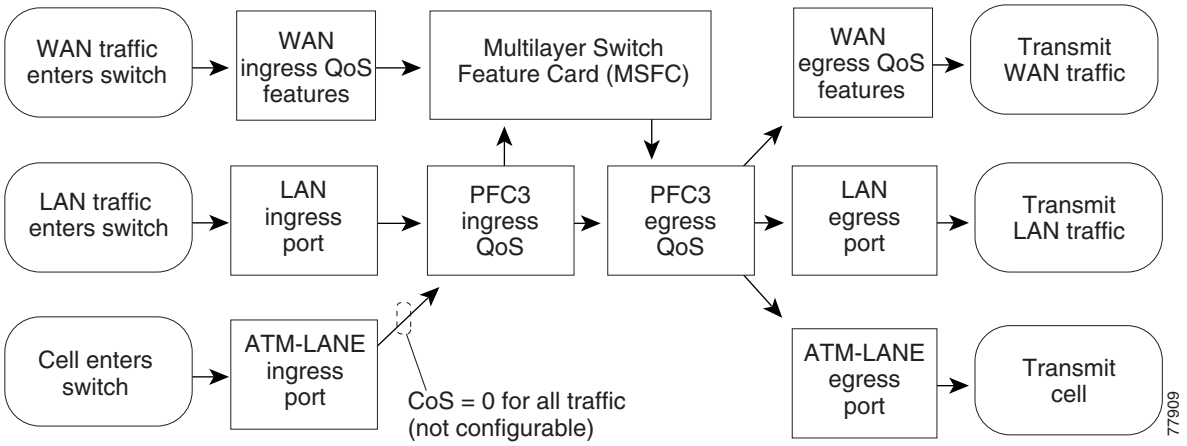
Note The Layer 3 IP packets can carry either an IP precedence value or a DSCP value. QoS supports the use of either value, because DSCP values can be set equal to the IP precedence values.

- *Classification* is the selection of traffic.
- *Marking*, according to RFC 2475, is the process of setting a Layer 3 DSCP value in a packet; in this publication, the definition of marking is extended to include setting the Layer 2 CoS values.
- *Scheduling* is the assignment of traffic to a queue. QoS assigns the traffic that is based on the CoS values.
- *Congestion avoidance* is the process by which QoS reserves the ingress and egress port capacity for the traffic with the high-priority CoS values. QoS implements congestion avoidance with the CoS value-based drop thresholds. A drop threshold is the percentage of buffer utilization at which the traffic with a specified CoS value is dropped, leaving the buffer available for the traffic with the higher-priority CoS values.
- *Policing* is the process by which the switch limits the bandwidth that is consumed by a flow of traffic. Policing can mark or drop traffic.
- Except where specifically differentiated, the *Layer 3 switching engine* refers to either of the following:
  - Supervisor Engine 2 with Layer 3 Switching Engine II (Policy Feature Card 2 or PFC2)
  - Supervisor Engine 1 with Layer 3 Switching Engine WS-F6K-PFC (Policy Feature Card or PFC)
- Random early detection (RED) is a drop threshold algorithm.
- Shaped round robin (SRR) is a dequeuing algorithm.
- Weighted random early detection (WRED) is a drop threshold algorithm.
- Weighted round robin (WRR) is a dequeuing algorithm.
- Deficit weighted round robin (DWRR) is a dequeuing algorithm.

Flowcharts

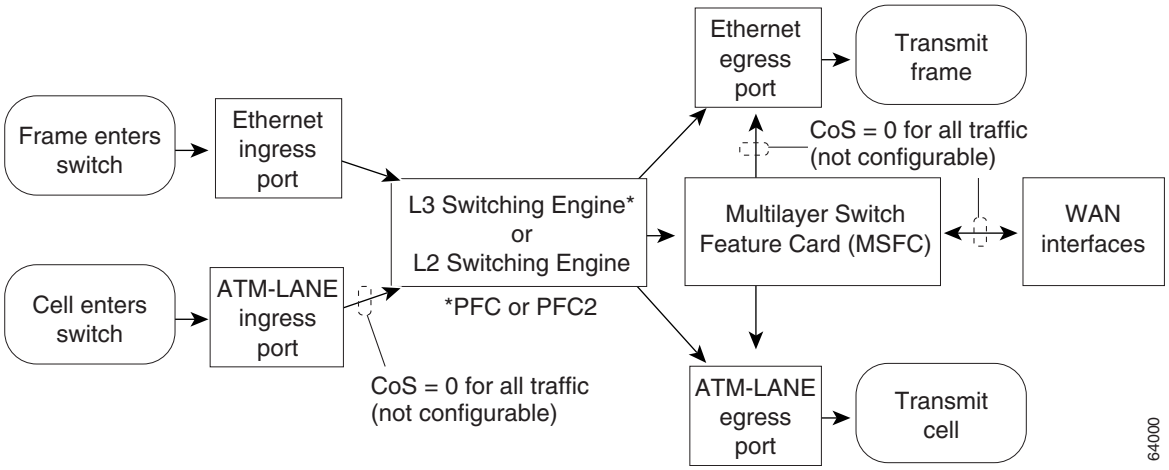
Figure 52-1 shows how traffic flows through the QoS features; Figure 52-2 through Figure 52-9 show more details of the traffic flow through QoS features.

Figure 52-1 Traffic Flow Through QoS Features with PFC3



Note PFC3 can provide Layer 3 switching for the WAN traffic. PFC3 can provide QoS for the ingress WAN traffic that is forwarded in the software by MSFC3. PFC3 can provide QoS for the egress WAN traffic that entered the switch through a LAN port or that was forwarded in the software by MSFC3.

Figure 52-2 Traffic Flow Through QoS Features with PFC and PFC2



Note

- PFC or PFC2 can provide Layer 3 switching for the ingress WAN traffic.
- PFC or PFC2 does not provide QoS for the WAN traffic. With PFC or PFC2, PFC QoS does not change the ToS byte in the WAN traffic.
- Ingress LAN traffic that is Layer 3 switched does not go through the Multilayer Switch Feature Card (MSFC or MSFC2) and retains the CoS value that is assigned by the Layer 3 switching engine.
- Enter the **show port capabilities** command to see the queue structure of a port (for more information, see the "Receive Queues" section on page 52-13 and the "Transmit Queues" section on page 52-28).

Figure 52-3 Ethernet ingress Port Classification, Marking, Scheduling, and Congestion Avoidance

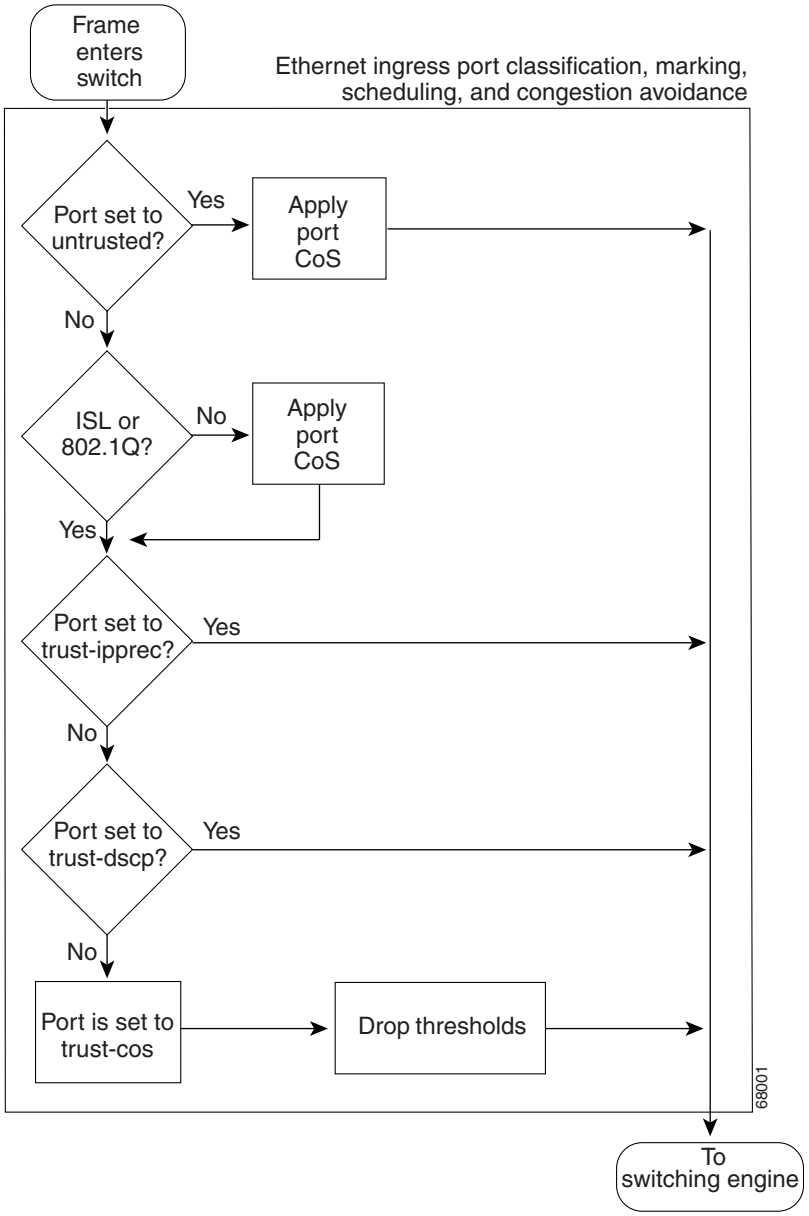


Figure 52-4 PFC3 Classification, Marking, and Policing

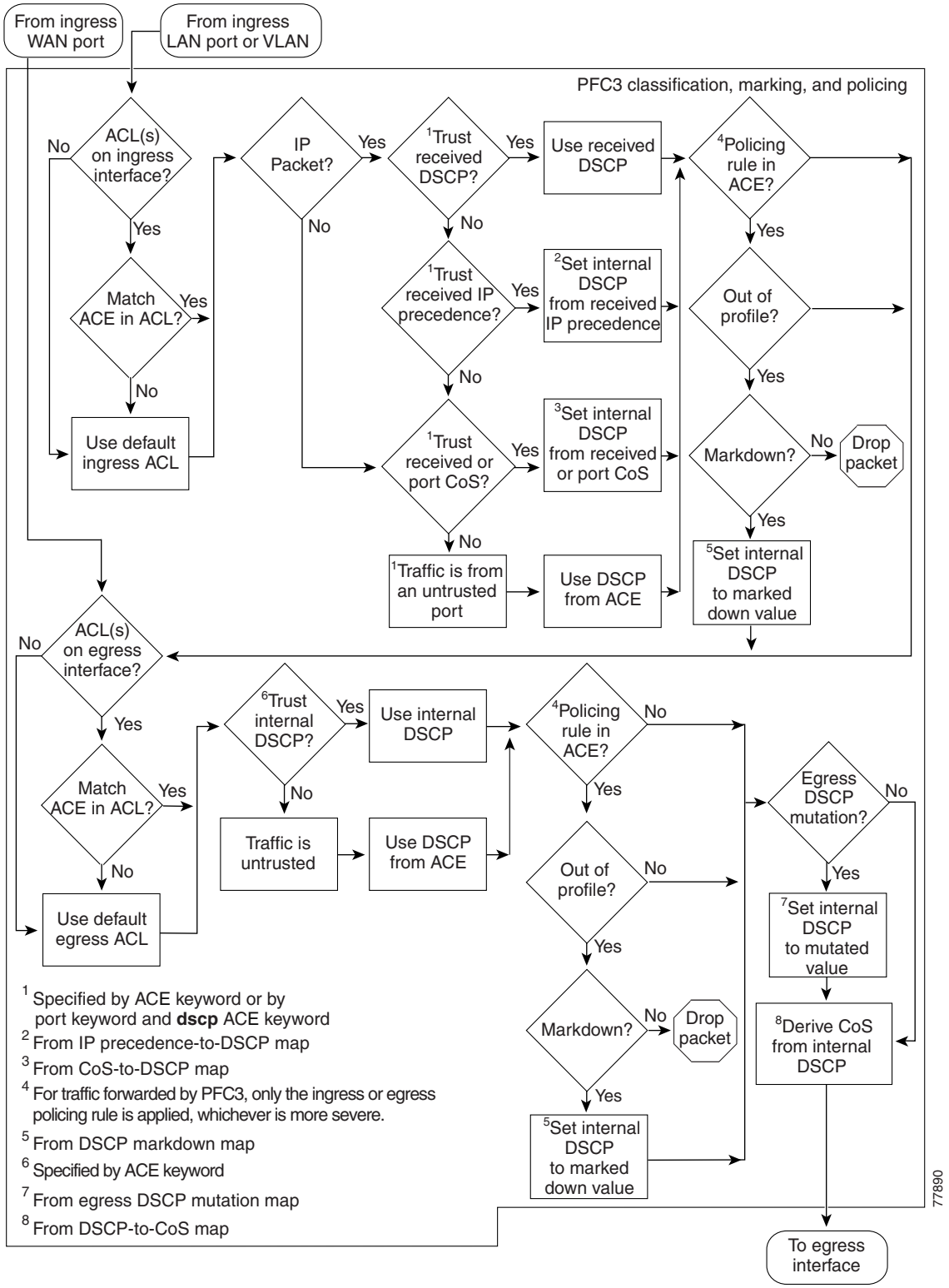
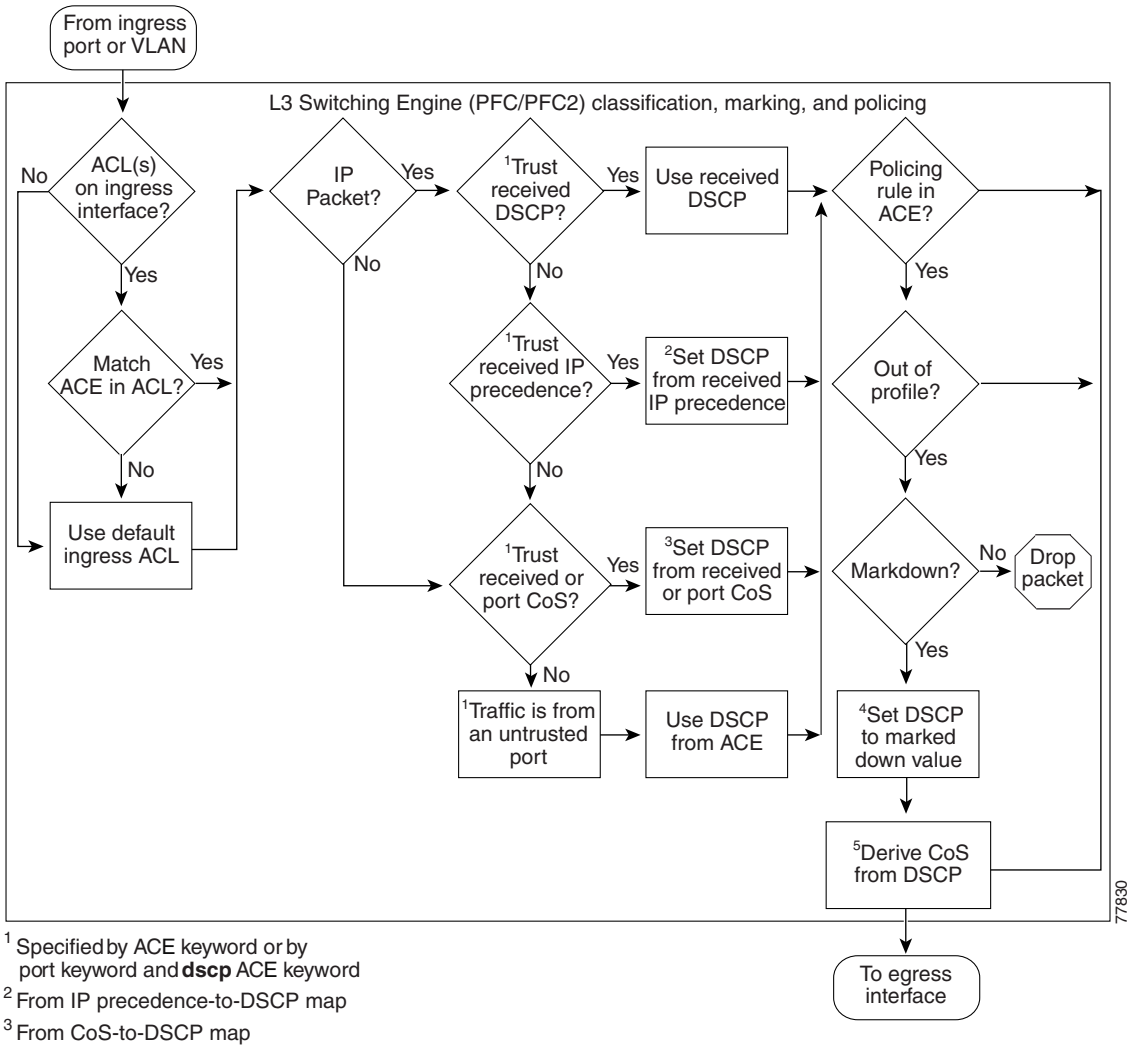


Figure 52-5 PFC and PFC2 Classification, Marking, and Policing



[1] Specified by ACE keyword or by port keyword and **dscp** ACE keyword
[2] From IP precedence-to-DSCP map
[3] From CoS-to-DSCP map
[4] From DSCP markdown map
[5] From DSCP-to-CoS map

Figure 52-6 Layer 2 Switching Engine Classification and Marking

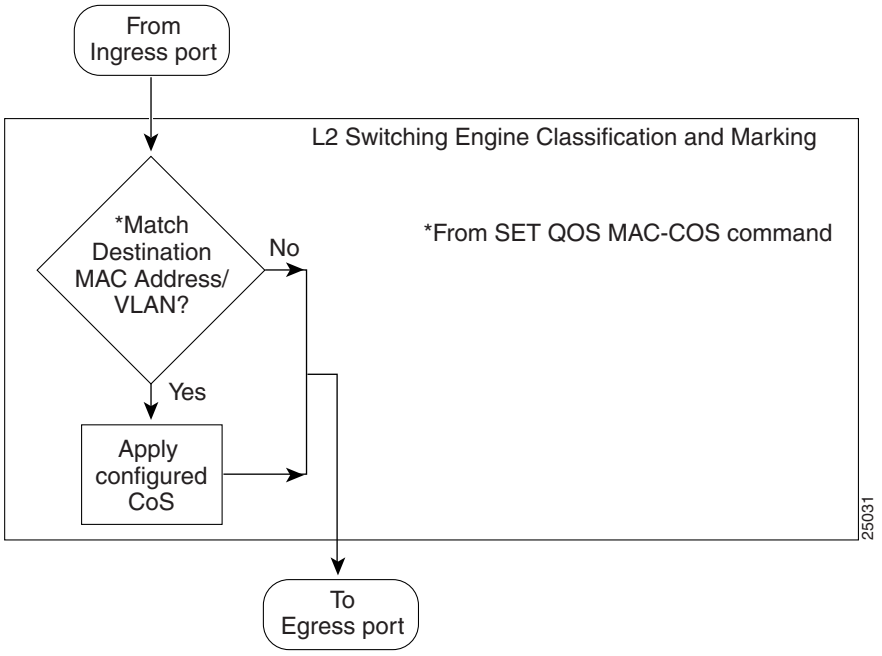


Figure 52-7 Multilayer Switch Feature Card Marking (MSFC and MSFC2)

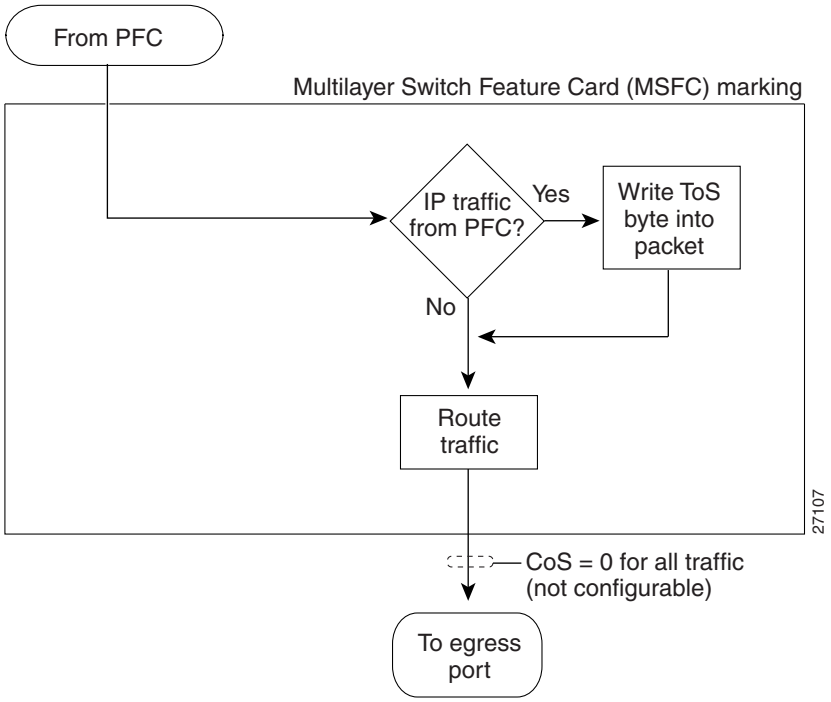


Figure 52-8 Ethernet Egress Port Scheduling, Congestion Avoidance, and Marking

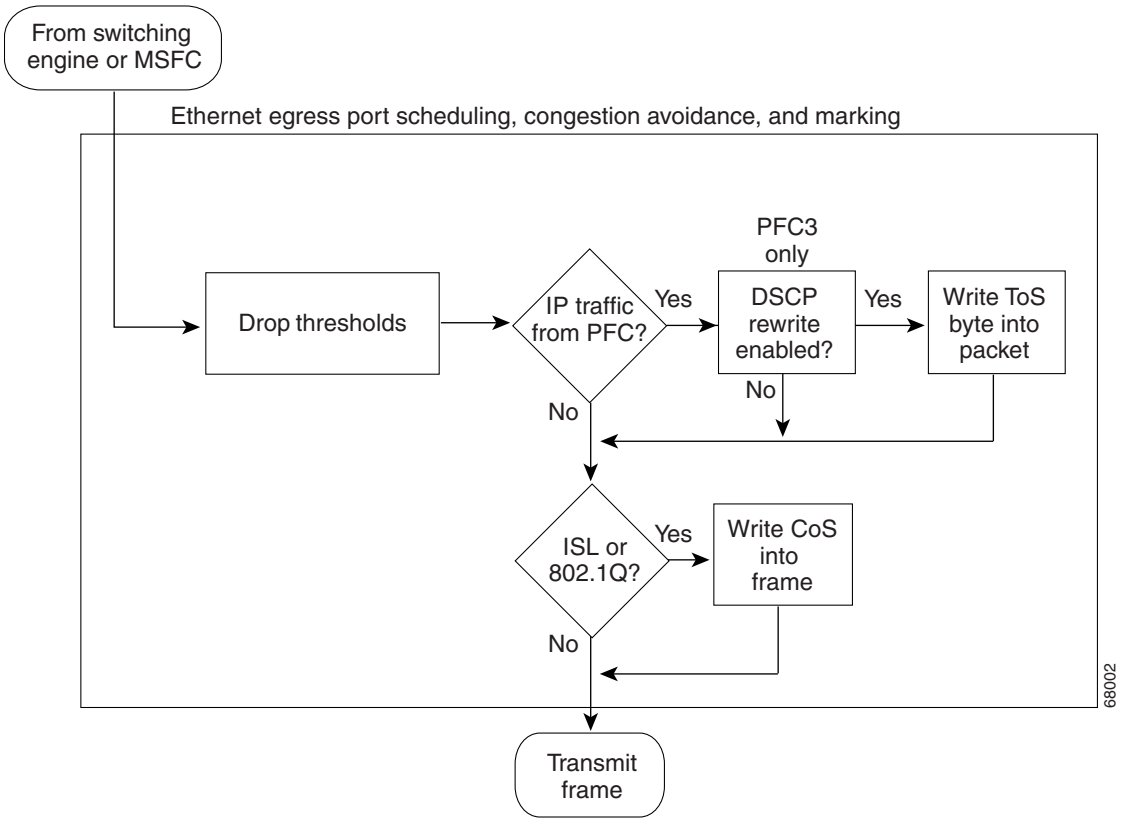
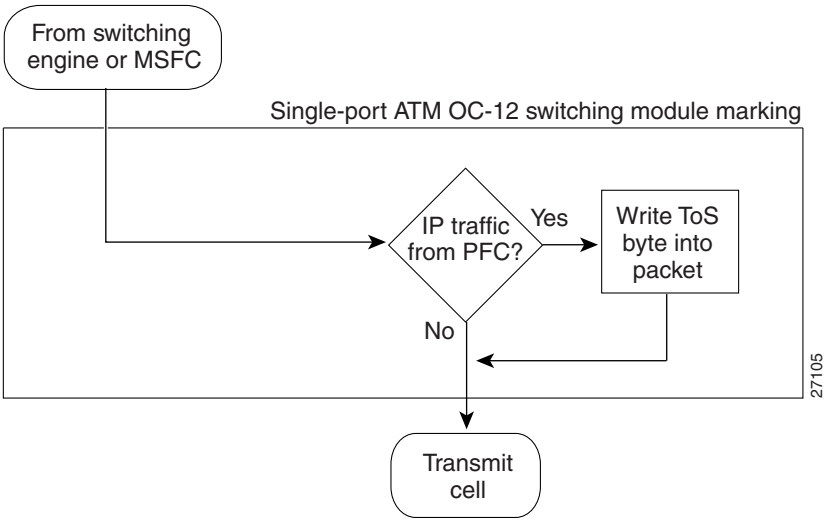


Figure 52-9 Single-Port ATM OC-12 Switching Module Marking



QoS Feature Set Summary

The QoS feature set on your switch is determined by the switching engine on the supervisor engine. Enter the **show module** command for the supervisor engine to display your switching engine configuration. The display shows the "Sub-Type" to be one of the following:

- Supervisor Engine 32 (WS-SUP32-GE-3B) with Policy Feature Card 3B (PFC3B/PFC3BXL)
- Supervisor Engine 720 (WS-SUP720) with PFC3A/PFC3B/PFC3BXL
- Supervisor Engine 2 (WS-X6K-SUP2-2GE) with Layer 3 Switching Engine II (WS-F6K-PFC2—Policy Feature Card 2 or PFC2)
- Supervisor Engine 1 (WS-X6K-SUP1A-2GE or WS-X6K-SUP1-2GE) with one of the following:
  - Layer 3 Switching Engine (WS-F6K-PFC—Policy Feature Card or PFC)
  - Layer 2 Switching Engine II (WS-F6020A)
  - Layer 2 Switching Engine I (WS-F6020)

The Layer 3 Switching Engine (WS-F6K-PFC) and Layer 3 Switching Engine II (WS-F6K-PFC2) support similar feature sets. The two Layer 2 switching engines support the same QoS feature set.

In addition to the features that are supported by the other two Layer 3 switching engines, PFC3A supports these features:

- Egress QoS
- Egress DSCP mutation
- Optional egress DSCP rewrite

These sections describe the QoS feature sets:

Ethernet Ingress Port Features

With any switching engine, QoS supports classification, marking, scheduling, and congestion avoidance using the Layer 2 CoS values at the Ethernet ingress ports. Classification, marking, scheduling, and congestion avoidance at the Ethernet ingress ports do not use or set the Layer 3 IP precedence or DSCP values. With a Layer 3 switching engine, you can configure the Ethernet ingress port trust states that can be used by the switching engine to set the Layer 3 IP precedence or DSCP values and the Layer 2 CoS value. For more information, see the "Ethernet Ingress Port Marking, Scheduling, Congestion Avoidance, and Classification" section on page 52-12.

Layer 3 Switching Engine Features

With PFC3A/PFC3B/PFC3BXL, PFC2, or PFC, QoS supports classification, marking, and policing using the access control lists (ACLs).

PFC3A/PFC3B/PFC3BXL provides QoS for the ingress and egress traffic. PFC2 and PFC provide QoS only for the ingress traffic.

With PFC3, QoS supports map-based egress DSCP mutation, which allows you to remark the egress traffic after it has been subjected to a policing rule, and the option to preserve the received DSCP in the egress traffic.

The ACLs contain the access control entries (ACEs) that specify the Layer 2, 3, and 4 classification criteria, a marking rule, and policers. Marking sets the Layer 3 IP precedence or DSCP values and the Layer 2 CoS value to either the received or configured Layer 2 or Layer 3 values. Policing uses bandwidth limits to either drop or mark the nonconforming traffic. For more information, see the "Classification, Marking, and Policing with a Layer 3 Switching Engine" section on page 52-15.

During processing, a Layer 3 switching engine associates a DSCP value with all traffic, including non-IP traffic. For more information, see the "Internal DSCP Values" section on page 52-16.

Layer 2 Switching Engine Features

With a Layer 2 Switching Engine, QoS can classify traffic using the Layer 2 destination MAC addresses, VLANs, and marking using the Layer 2 CoS values. Classification and marking with a Layer 2 Switching Engine do not use or set the Layer 3 IP precedence or DSCP values. For more information, see the "Classification and Marking on a Supervisor Engine 1 with a Layer 2 Switching Engine" section on page 52-28.

Ethernet Egress Port Features

With any switching engine, QoS supports Ethernet egress port scheduling and congestion avoidance using the Layer 2 CoS values. Ethernet egress port marking sets the Layer 2 CoS values and, with a Layer 3 switching engine, the Layer 3 DSCP values. For more information, see the "Ethernet Egress Port Scheduling, Congestion Avoidance, and Marking" section on page 52-28.

Single-Port ATM OC-12 Switching Module Features

The ingress interface from a single-port ATM OC-12 switching module is untrusted, and QoS sets CoS to zero in all traffic that is received from it. With a Layer 3 switching engine, QoS can mark the IP traffic that is transmitted to a single-port ATM OC-12 switching module with the Layer 3 DSCP values.

Multilayer Switch Feature Card (MSFC, MSFC2, or MSFC3)

QoS marks the IP traffic that is transmitted to an MSFC with the Layer 3 DSCP values. The CoS is zero in all traffic that is sent from an MSFC to the egress ports.

Note The traffic that is Layer 3 switched does not go through the MFSC and retains the CoS value that is assigned by the Layer 3 switching engine.

Ethernet Ingress Port Marking, Scheduling, Congestion Avoidance, and Classification

These sections describe the Ethernet ingress port marking, scheduling, congestion avoidance, and classification:

Overview

The trust state of an Ethernet port determines how it marks, schedules, and classifies the received traffic, and whether or not congestion avoidance is implemented. You can configure the trust state of each port with one of these keywords:

- **untrusted** (default)
- **trust-ipprec** (Layer 3 switching engine only—not supported on **1q4t** ports except Gigabit Ethernet)
- **trust-dscp** (Layer 3 switching engine only—not supported on **1q4t** ports except Gigabit Ethernet)
- **trust-cos**

Note On **1q4t** ports (except Gigabit Ethernet), the **trust-cos** port keyword displays an error message, activates the receive queue drop thresholds, and—as indicated by the error message—does not apply the **trust-cos** trust state to the traffic. You must configure the **trust-cos** ACL that matches the ingress traffic to apply the **trust-cos** trust state.

For more information, see the "Configuring the Trust State of a Port" section on page 52-41.

In addition to the port configuration keywords that are listed above, with a Layer 3 switching engine, QoS uses the **trust-ipprec**, **trust-dscp**, and **trust-cos** ACE keywords. Do not confuse the ACE keywords with the port keywords.

The ports that are configured with the **untrusted** keyword are called "untrusted ports." The ports that are configured with the **trust-ipprec**, **trust-dscp**, or **trust-cos** keywords are called "trusted ports." QoS implements ingress port congestion avoidance only on the ports that are configured with the **trust-cos** keyword.

Ingress port marking, scheduling, and congestion avoidance use the Layer 2 CoS values. Ingress port marking, scheduling, and congestion avoidance do not use or set the Layer 3 IP precedence or DSCP values.

Marking at Untrusted Ports

QoS marks all frames that are received through the untrusted ports with the port CoS value (the default is zero). QoS does not implement ingress port congestion avoidance on the untrusted ports. The traffic goes directly to the switching engine.

Marking at Trusted Ports

When an ISL frame enters the switch through a trusted port, QoS accepts the three least significant bits in the User field as a CoS value. When an 802.1Q frame enters the switch through a trusted port, QoS accepts the User Priority bits as a CoS value. QoS marks all traffic that is received in the other frame types with the port CoS value.

The port CoS value is configurable for each Ethernet port. For more information, see the "Configuring the CoS Value for a Port" section on page 52-41.

Ethernet Ingress Port Scheduling and Congestion Avoidance

QoS does not implement ingress port congestion avoidance on the ports that are configured with the **untrusted**, **trust-ipprec**, or **trust-dscp** keywords. The traffic goes directly to the switching engine.

QoS uses the CoS-value-based receive-queue drop thresholds to avoid congestion in the traffic that is entering the switch through a port that is configured with the **trust-cos** keyword (for more information, see the "Configuring the Trust State of a Port" section on page 52-41).

Receive Queues

Enter the **show port capabilities** command to see the queue structure of a port. The command displays one of the following:

- **rx-(1q8t)** indicates one standard queue with eight configurable tail-drop thresholds.
- **rx-(1q2t)** indicates one standard queue with one configurable tail-drop threshold and one nonconfigurable tail-drop threshold.
- **rx-(1q4t)** indicates one standard queue with four configurable tail-drop thresholds.
- **rx-(1p1q4t)** indicates one strict-priority queue and one standard queue with four configurable tail-drop thresholds.
- **rx-(1p1q0t)** indicates one strict-priority queue and one standard queue with a nonconfigurable threshold.
- **rx-(1p1q8t)** indicates one strict-priority queue and one standard queue with eight configurable WRED-drop thresholds (on the **1p1q8t** ports, the standard queue also has one nonconfigurable tail-drop threshold).

Strict-priority queues are serviced in preference to other queues. QoS services the traffic in a strict-priority queue before servicing the standard queue. When QoS services the standard queue, after receiving a packet, it checks for the traffic in the strict-priority queue. If QoS detects the traffic in the strict-priority queue, it suspends its service of the standard queue and completes the service of all traffic in the strict-priority queue before returning to the standard queue.

Ingress Scheduling

QoS schedules the traffic through the receive queues based on the CoS values. In the default configuration, PFC QoS assigns all traffic with CoS 5 to the strict-priority queue (if present); PFC QoS assigns all other traffic to the standard queue. In the absence of a strict-priority queue, PFC QoS assigns all traffic to the standard queues.

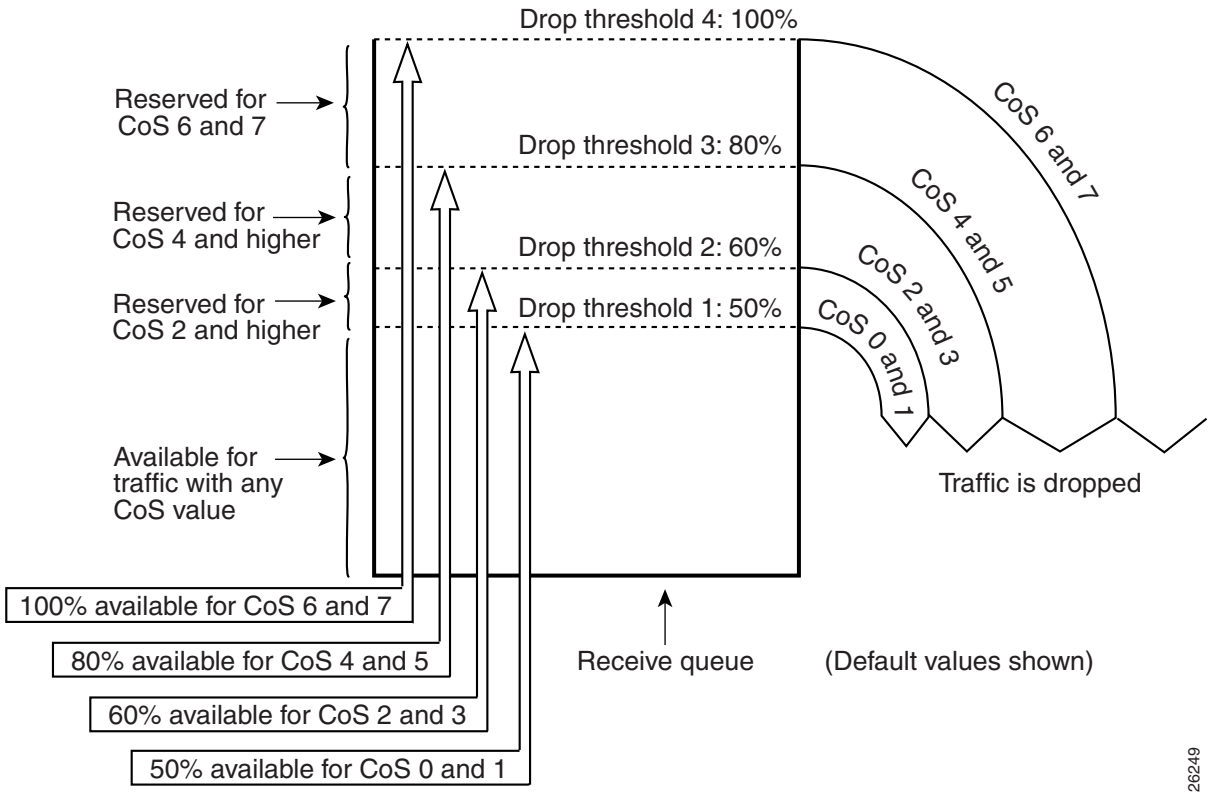
Ingress Congestion Avoidance

If a port is configured with the **trust-cos** keyword, QoS implements CoS-value-based receive-queue drop thresholds to avoid congestion in the received traffic. See the "QoS Default Configuration" section on page 52-30 for the default CoS-to-threshold mapping.

Note For some port types, you can configure the standard receive queue to use both a tail-drop and a WRED-drop threshold by mapping a CoS value to the queue or to the queue and a threshold. The switch uses the tail-drop threshold for the traffic carrying the CoS values that are mapped only to the queue. The switch uses the WRED-drop thresholds for the traffic carrying the CoS values that are mapped to the queue and a threshold. For more information, see the "1p1q8t Receive Queues" section on page 52-71.

Figure 52-10 shows the drop thresholds for a **1q4t** port. The drop thresholds in the other configurations function similarly.

Figure 52-10 Receive-Queue Drop Thresholds



Ethernet Ingress Port Classification Features with a Layer 3 Switching Engine

You can use the **untrusted**, **trust-ipprec**, **trust-dscp**, and **trust-cos** port keywords to classify the traffic on a per-port basis for a Layer 3 switching engine to mark.

The **trust-ipprec** and **trust-dscp** keywords are supported only with a Layer 3 switching engine and are not supported on the **1q4t** ports except Gigabit Ethernet. On the **1q4t** ports (except Gigabit Ethernet), the **trust-cos** port keyword displays an error message, activates the receive-queue drop thresholds, and—as indicated by the error message—does not apply the **trust-cos** trust state to the traffic. You must configure the **trust-cos** ACL that matches the ingress traffic to apply the **trust-cos** trust state.

In addition to the per-port classification, you can create the ACEs that classify the traffic on a per-packet basis (for the IP and IPX traffic, see the "Named IP ACLs" section on page 52-46 and the "Creating or Modifying the Named IPX ACLs" section on page 52-51) or on a per-frame basis (for the other traffic, see the "Creating or Modifying the Named MAC ACLs" section on page 52-53), regardless of the port configuration (see the "Marking Rules" section on page 52-23).

To mark the traffic in response to per-port classification, the traffic must match an ACE that contains the **dscp** ACE keyword (see the "Marking Rules" section on page 52-23). In their default configuration, the ACEs in the default ACLs contain the **dscp** ACE keyword. Table 52-1 lists the per-port classifications and the marking rules that they invoke.

Table 52-1 Marking Based on Per-Port Classification

Port Keyword	ACE Keyword	Marking Rule
untrusted	**dscp**	Set the internal and egress DSCP as specified in the ACE.
trust-ipprec	**dscp**	For the IP traffic, set the internal and egress DSCP from the received Layer 3 IP precedence value. For the other traffic, set the internal and egress DSCP from the received or port Layer 2 CoS value. **Note**—With the **trust-ipprec** port keyword, QoS uses only the IP precedence bits. If the traffic with a DSCP value enters the switch through a port that is configured with the **trust-ipprec** port keyword, the three most significant bits of the DSCP value are interpreted as an IP precedence value; QoS ignores the rest of the DSCP value.
trust-dscp	**dscp**	For the IP traffic, set the internal and egress DSCP from the received Layer 3 DSCP value. For other traffic, set the internal and egress DSCP from the received or port Layer 2 CoS value.
trust-cos	**dscp**	Set the internal and egress DSCP from the received or port Layer 2 CoS value.

QoS uses the configurable mapping tables to set the internal and egress DSCP, which is a 6-bit value, from the CoS and IP precedence, which are 3-bit values (for more information, see the "Internal DSCP Values" section on page 52-16 and the "Configuring the DSCP Value Maps" section on page 52-73).

Classification, Marking, and Policing with a Layer 3 Switching Engine

Note With a Layer 3 switching engine, the Catalyst 6500 series switches provide QoS only for the following frame types: Ethernet_II, Ethernet_802.3, Ethernet_802.2, and Ethernet_SNAP.

These sections describe classification, marking, and policing with a Layer 3 switching engine:

Note Classification with a Layer 3 switching engine uses the Layer 2, 3, and 4 values. Marking with a Layer 3 switching engine uses the Layer 2 CoS values and the Layer 3 IP precedence or DSCP values.

Internal DSCP Values

These sections describe the internal DSCP values:

Internal DSCP Sources

During processing, the priority of all traffic (including non-IP traffic) is represented with an internal DSCP value. QoS derives the internal DSCP value from the following:

- For the **trust-cos** traffic, from the received or port Layer 2 CoS values (the traffic from an untrusted port has the port CoS value and if the traffic from an untrusted port matches a **trust-cos** ACL, QoS derives the internal DSCP value from the port CoS value)
- For the **trust-ipprec** traffic, from the received IP precedence values
- For the **trust-dscp** traffic, from the received DSCP values
- For the **untrusted** traffic, from the port CoS or configured DSCP values

The trust state of the traffic is the trust state of the ingress port unless set otherwise by the matching ACE.

Note A **trust-cos** ACL cannot restore the received CoS in the traffic from the untrusted ports. The traffic from the untrusted ports always has the port CoS value.

QoS uses the configurable mapping tables to derive the internal 6-bit DSCP value from the CoS or IP precedence, which are 3-bit values (see the"Mapping the Received CoS Values to the Internal DSCP Values" section on page 52-73 or the "Mapping the Received IP Precedence Values to the Internal DSCP Values" section on page 52-74).

Egress DSCP and CoS Sources

For the egress IP traffic, QoS creates a ToS byte from the internal DSCP value (which you can set equal to an IP precedence value) and sends it to the egress port to be written into the IP packets. For the **trust-dscp** and **untrusted** IP traffic, the ToS byte includes the original 2 least-significant bits from the received ToS byte.

For all egress traffic, QoS uses a configurable mapping table to derive a CoS value from the internal DSCP value that is associated with the traffic (see the "Mapping the Internal DSCP Values to the Egress CoS Values" section on page 52-74). QoS sends the CoS value to the Ethernet egress ports for use in scheduling and to be written into the ISL and 802.1Q frames.

ACLs

QoS uses the ACLs that contain the ACEs. The ACEs specify the classification criteria, a marking rule, and the policers. QoS compares the received traffic to the ACEs in the ACLs until a match occurs. When the traffic matches the classification criteria in an ACE, QoS marks and polices the packet as specified in the ACE and makes no further comparisons.

There are three ACL types: IP, and with a Layer 3 switching engine, IPX and MAC. QoS compares the traffic of each type (IP, IPX, and MAC) only to the corresponding ACL type (see Table 52-2).

Table 52-2 Supported EtherType Field Values

ACL Type	EtherType Field Value	Protocol
IP	0x0800	IP
IPX [1]	0x8137 and 0x8138	IPX
MAC[2]	0x0600 and 0x0601	XNS
	0x0BAD and 0x0BAF	Banyan VINES
	0x6000-0x6009 and 0x8038-0x8042	DECnet
	0x809b and 0x80f3	AppleTalk

1. The PFC3 does not provide QoS for the IPX traffic.
2. The QoS MAC ACLs that do not include an EtherType parameter match the traffic with any value in the EtherType field, which allows MAC-level QoS to be applied to any traffic except IP and IPX.

QoS supports the user-created *named* ACLs, each containing an ordered list of ACEs, and the user-configurable *default* ACLs, each containing a single ACE.

Named ACLs

You create a named ACL when you enter an ACE with a new ACL name. You add an ACE to an existing ACL when you enter an ACE with the name of the existing ACL.

You can specify the classification criteria for each ACE in a named ACL. The classification criteria can be specific values or wildcards (for more information, see the "Creating or Modifying ACLs" section on page 52-45).

These sections describe the classification criteria that can be specified in a named ACL:

IP ACE Layer 3 Classification Criteria

You can create the IP ACEs that match the traffic with the specific Layer 3 values by including these Layer 3 parameters (see the "Named IP ACLs" section on page 52-46):

- IP source address and mask, entered as specific values or with the **any** keyword or with the **host** keyword and a host address.
- IP destination address and mask, entered as specific values or with the **any** keyword or with the **host** keyword and a host address.
- DSCP value (0–63) or IP precedence that is specified with a numeric value (0–7) or these keywords:
  - **Network** (IP precedence 7)
  - **Internet** (IP precedence 6)
  - **Critical** (IP precedence 5)
  - **Flash-override** (IP precedence 4)
  - **Flash** (IP precedence 3)
  - **Immediate** (IP precedence 2)
  - **Priority** (IP precedence 1)
  - **Routine** (IP precedence 0)

Note The IP ACEs that do not include a DSCP or IP precedence value parameter match all DSCP or IP precedence values.

IP ACE Layer 4 Protocol Classification Criteria

You can create the IP ACEs that match the specific Layer 4 protocol traffic by including a Layer 4 protocol parameter (see the "IP ACLs for Other Layer 4 Protocols" section on page 52-49). You can specify the protocol numerically (0–255) or with these keywords: **ahp** (51), **eigrp** (88), **esp** (50), **gre** (47), **igrp** (9), **icmp** (1), **igmp** (2), **igrp** (9), **ip** (0), **ipinip** (4), **nos** (94), **ospf** (89), **pcp** (108), **pim** (103), **tcp** (6), or **udp** (17).

Note The IP ACEs that do not include a Layer 4 protocol parameter or that include the **ip** keyword match all IP traffic.

IP ACE Layer 4 TCP Classification Criteria

You can create the Transmission Control Protocol (TCP) ACEs that match the traffic for the specific TCP ports by including the TCP source and/or destination port parameters (for more information, see the "IP ACEs for TCP Traffic" section on page 52-47).

You can specify the TCP port parameters numerically (0–65535) or with these keywords:

Keyword	Port		Keyword	Port		Keyword	Port		Keyword	Port
bgp	179		**ftp**	21		**lpd**	515		**telnet**	23
chargen	19		**ftp-data**	20		**nntp**	119		**time**	37
daytime	13		**gopher**	70		**pop2**	109		**uucp**	540
discard	9		**hostname**	101		**pop3**	110		**whois**	43
domain	53		**irc**	194		**smtp**	25		**www**	80
echo	7		**klogin**	543		**sunrpc**	111			
finger	79		**kshell**	544		**tacacs**	49			

Note The TCP ACEs that do not include a Layer 4 TCP port parameter match all TCP traffic.

IP ACE Layer 4 UDP Classification Criteria

You can create the User Datagram Protocol (UDP) ACEs that match the traffic for specific UDP source and/or destination ports by including the UDP port parameters. For more information, see the "IP ACEs for UDP Traffic" section on page 52-48.

You can specify the UDP port parameters numerically (0–65535) or with these keywords:

Keyword	Port		Keyword	Port		Keyword	Port		Keyword	Port
biff	512		**echo**	7		**rip**	520		**talk**	517
bootpc	68		**mobile-ip**	434		**snmp**	161		**tftp**	69
bootps	67		**nameserver**	42		**snmptrap**	162		**time**	37
discard	9		**netbios-dgm**	138		**sunrpc**	111		**who**	513
dns	53		**netbios-ns**	137		**syslog**	514		**xdmcp**	177
dnsix	195		**ntp**	123		**tacacs**	49			

Note The UDP ACEs that do not include a Layer 4 UDP port parameter match all UDP traffic.

IP ACE Layer 4 ICMP Classification Criteria

You can create the Internet Control Management Protocol (ICMP) ACEs that match the traffic containing specific ICMP messages by including the ICMP types and optionally, the ICMP codes. For more information, see the "IP ACEs for ICMP Traffic" section on page 52-48.

You can specify the ICMP types and codes numerically (0–255) or with these keywords:

Keyword	Type	Code	Keyword	Type	Code
administratively-prohibited	3	13	**net-tos-unreachable**	3	11
alternate-address[1]	6	—	**net-unreachable**	3	0
conversion-error	31	0	**network-unknown**	3	6
dod-host-prohibited	3	10	**no-room-for-option**	12	2
dod-net-prohibited	3	9	**option-missing**	12	1
echo	8	0	**packet-too-big**	3	4
echo-reply	0	0	**parameter-problem**	12	0
general-parameter-problem[1]	12	—	**port-unreachable**	3	3
host-isolated	3	8	**precedence-unreachable**	3	15
host-precedence-unreachable	3	14	**protocol-unreachable**	3	2
host-redirect	5	1	**reassembly-timeout**	11	1
host-tos-redirect	5	3	**redirect**[1]	5	—
host-tos-unreachable	3	12	**router-advertisement**	9	0
host-unknown	3	7	**router-solicitation**	10	0
host-unreachable	3	1	**source-quench**	4	0
information-reply	16	0	**source-route-failed**	3	5
information-request	15	0	**time-exceeded**[1]	11	—
mask-reply	18	0	**timestamp-reply**	14	0
mask-request	17	0	**timestamp-request**	13	0
mobile-redirect	32	0	**traceroute**	30	0
net-redirect	5	0	**ttl-exceeded**	11	0
net-tos-redirect	5	2	**unreachable**[1]	3	—

1. Matches all code values

Note The ICMP ACEs with only a Layer 4 ICMP *type* parameter match all *code* values for that *type* value. The ICMP ACEs that do not include any Layer 4 ICMP type and code parameters match all ICMP traffic.

IP ACE Layer 4 IGMP Classification Criteria

You can create the IGMP ACEs that match the traffic containing the specific IGMP messages by including an IGMP type parameter (for more information, see the "IP ACEs for IGMP Traffic" section on page 52-49). You can specify the IGMP type numerically (0–255) or with these keywords: **host-query** (1), **host-report** (2), **dvmrp** (3), **pim** (4), or **trace** (5).

Note

- QoS supports multicast traffic when IGMP snooping is enabled.
- QoS does not support the IGMP traffic when IGMP snooping is enabled.
- QoS supports IGMP classification using version 1 four-bit Type fields.
- The IGMP ACEs that do not include a Layer 4 IGMP type parameter match all IGMP traffic.

IPX ACE Classification Criteria

Note PFC3 does not provide QoS for the IPX traffic. See the "MAC ACE Layer 2 Classification Criteria" section on page 52-21 for information about how to use MAC ACLs to filter IPX traffic.

You can create the IPX ACEs that match the specific IPX traffic by including these parameters (for more information, see the "Creating or Modifying the Named IPX ACLs" section on page 52-51):

- IPX source network (-1 matches any network number)
- Protocol, which can be specified numerically (0–255) or with these keywords: **any**, **ncp** (17), **netbios** (20), **rip** (1), **sap** (4), **spx** (5)
- IPX ACEs support the following optional parameters:
  - IPX destination network (-1 matches any network number)
  - If you specify an IPX destination network, the IPX ACEs support the following optional parameters: an IPX destination network mask (-1 matches any network number), an IPX destination node, and an IPX destination node mask

MAC ACE Layer 2 Classification Criteria

You can create the MAC ACEs that match the specific Ethernet traffic by including these Layer 2 parameters (for more information, see the "Creating or Modifying the Named MAC ACLs" section on page 52-53):

- Ethernet source and destination addresses and masks, entered as specific values or with the **any** keyword or with the **host** keyword and a host Ethernet address
- Optionally, an EtherType parameter from this list:
  - 0x809B (or **ethertalk**)
  - 0x80F3 (or **aarp**)
  - 0x6001 (or **dec-mop-dump**)
  - 0x6002 (or **dec-mop-remote-console**)
  - 0x6003 (or **dec-phase-iv**)
  - 0x6004 (or **dec-lat**)

  - 0x6005 (or **dec-diagnostic-protocol**)
  - 0x6007 (or **dec-lavc-sca**)
  - 0x6008 (or **dec-amber**)
  - 0x6009 (or **dec-mumps**)
  - 0x8038 (or **dec-lanbridge**)
  - 0x8039 (or **dec-dsm**)
  - 0x8040 (or **dec-netbios**)
  - 0x8041 (or **dec-msdos**)
  - 0x8042 (no keyword)
  - 0x0BAD (no keyword)
  - 0x0baf (or **banyan-vines-echo**)
  - 0x0600 (or **xerox-ns-idp**)
- Optionally with PFC3A, an EtherType parameter from this list:
  - 0x8137 (or **ipx-arpa**)
  - 0xffff for non-ARPA IPX

The QoS MAC ACLs that do not include an EtherType parameter match the traffic with any value in the EtherType field, which allows the MAC-level QoS to be applied to any traffic except IP and IPX.

Default ACLs

There are three default ACLs, one each for IP, and with a Layer 3 switching engine, IPX and MAC traffic. Each ACL has a single ACE that has a configurable marking rule and configurable policers. The default ACLs have nonconfigurable classification criteria that matches all traffic. QoS compares any traffic with a supported EtherType field value that does not match a named ACL to the default ACLs. The unmatched IP traffic matches the default IP ACL. The unmatched IPX traffic matches the default IPX ACL. The unmatched Ethernet traffic matches the default MAC ACL.

Note All traffic matches an ACE in an ACL, either an ACE in a named ACL or one of the default ACLs, because the default ACLs match all traffic.

Marking Rules

Note PFC2 cannot not mark IPX or MAC traffic. PFC3 does not provide QoS for IPX traffic.

The marking rules specify how QoS marks the traffic when the traffic matches the filtering parameters in an ACE (see the "ACE Name, Marking Rule, Policing, and Filtering Syntax" section on page 52-46). QoS supports four marking rules that are specified with the following four ACE keywords: **trust-dscp**, **trust-ipprec**, **trust-cos**, and **dscp**. Each ACE contains one of the keywords.

The marking rules are as follows:

- **trust-dscp** (IP ACLs only)—Instructs QoS to set the internal and egress DSCP from the received DSCP values (see the "Internal DSCP Values" section on page 52-16).
- **trust-ipprec** (IP ACLs only)—Instructs QoS to set the internal and egress DSCP from the received IP precedence values.

  Note With the **trust-ipprec** port keyword, QoS uses only the IP precedence bits. If the traffic with a DSCP value enters the switch through a port that is configured with the **trust-ipprec** port keyword, the three most significant bits of the DSCP value are interpreted as an IP precedence value; QoS ignores the rest of the DSCP value.

- **trust-cos** (all ACLs except IPX and MAC with PFC2 and IPX with PFC3)—Instructs QoS to set the internal and egress DSCP from the received or port CoS values. In the traffic from the ports that are configured with the **trust-cos** keyword, QoS uses the CoS value that is received in the ISL and 802.1Q frames; in all other cases, QoS uses the CoS value that is configured on the port (default is zero).
- **dscp** (all ACLs except IPX and MAC with PFC2 and IPX with PFC3)—Instructs QoS to mark the traffic as indicated by the port trust keywords:
  - In the IP traffic from the ingress ports that are configured with the **trust-dscp** port keyword, the **dscp** ACE keyword instructs QoS to set the internal and egress DSCP values from the received DSCP values. In the non-IP traffic, QoS sets the DSCP from the received or port CoS value.
  - In the IP traffic from the ingress ports that are configured with the **trust-ipprec** port keyword, the **dscp** ACE keyword instructs QoS to set the internal and egress DSCP values from the received IP precedence values. In the non-IP traffic, QoS sets the DSCP value from the received or port CoS value.
  - In the traffic from the ingress ports that are configured with the **trust-cos** port keyword, the **dscp** ACE keyword instructs QoS to set the internal and egress DSCP values from the received or port CoS values.
  - In the traffic from the ingress ports that are configured with the **untrusted** port keyword, the **dscp** ACE keyword instructs QoS to set the internal and egress DSCP values from the DSCP value in the ACE.

  Note The default configuration of the ACEs in the default ACLs contains the **dscp** ACE keyword, which supports the per-port classification of the traffic. With the default values, the ACEs in the default ACLs apply DSCP zero to the traffic from the ingress ports that are configured with the **untrusted** port keyword.

QoS uses the configurable mapping tables to set the DSCP value, which is 6 bits, from the CoS and IP precedence, which are 3-bit values (for more information, see the "Mapping the Received CoS Values to the Internal DSCP Values" section on page 52-73 and the "Mapping the Received IP Precedence Values to the Internal DSCP Values" section on page 52-74).

Policers

You can create the named policers that specify the bandwidth utilization limits, which you can apply to the traffic by including the policer name in an ACE (for more information, see the "Creating Policers" section on page 52-42).

Policing uses a token bucket scheme. As the packets arrive, the packet size in bytes is added to the bucket level. Every 0.25 ms, a value equal to the token rate is subtracted from the bucket level.

You specify the bandwidth utilization limits as an average rate and a maximum burst size. The packets that exceed these limits are "out of profile." The traffic is in profile as long as it flows in at an average rate and never bursts beyond the burst size.

With PFC and PFC2, the policing rates use the Layer 3 packet size. With PFC3, the policing rates use the Layer 2 frame size.

In each policer, you specify if the out-of-profile packets are to be dropped or to have a new DSCP value applied to them (applying a new DSCP value is called "markdown"). Because the out-of-profile packets do not retain their original priority, they are not counted as part of the bandwidth that is consumed by the in-profile packets.

For all policers, QoS uses a configurable table that maps the received DSCP values to the marked-down DSCP values (for more information, see the "Mapping the DSCP Markdown Values" section on page 52-75). When the markdown occurs, QoS gets the marked-down DSCP value from the table. You cannot specify a marked-down DSCP value in the individual policers.

Note By default, the markdown table is configured so that no markdown occurs; the marked-down DSCP values are equal to the received DSCP values. To enable the markdown, configure the table appropriately for your network.

You give each policer a unique name when you create it and then use the name to include the policer in an ACE. The same policer can be used in multiple ACEs.

You can create these policers:

- Microflow—QoS applies the bandwidth limit that is specified in a microflow policer separately to each flow that matches any ACEs that use that particular microflow policer. You can create up to 63 microflow policers.
- Aggregate—QoS applies the bandwidth limits that are specified in an aggregate policer cumulatively to all flows that match any ACEs that use that particular aggregate policer. You can create up to 1023 aggregate policers.
- With PFC2 and PFC3A, you can specify a dual rate aggregate policer with a normal rate and an excess rate:
  - Normal rate—The packets exceeding this rate are marked down.
  - Excess rate—The packets exceeding this rate are either marked down or dropped as specified by the drop indication flag.

Note The drop indication flag applies to the excess rate policer and cannot be set for the normal rate policer. To achieve the effect of a drop indication flag for the normal rate aggregate policer, set the excess rate equal to the normal rate and set the drop indication flag. Alternatively, you can set the normal rate without specifying an excess rate, which automatically sets the excess rate to the normal rate when the drop indicator flag is on.

You can include both a microflow policer and an aggregate policer in each ACE to police a flow that is based on both its own bandwidth utilization and on its bandwidth utilization combined with that of the other flows.

For example, you could create a microflow policer named "group_individual" with the bandwidth limits that are suitable for the flows of the individuals in a group, and you could create an aggregate policer named "group_all" with the bandwidth limits that are suitable for the group as a whole. You could include both policers in the ACEs that match the group's traffic. The combination would affect the individuals separately and the group cumulatively.

For the ACEs that include both a microflow policer and an aggregate policer, QoS responds to an out-of-profile status from either policer, and as specified by the policer, applies a new DSCP value or drops the packet. If both policers return an out-of-profile status, then if either policer specifies that the packet is to be dropped, it is dropped; otherwise, QoS applies a new DSCP value.

Follow these guidelines when creating policers:

- You can include a microflow policer in the IP ACEs. You cannot include a microflow policer in the IPX or MAC ACEs. The IPX and MAC ACEs support only the aggregate policers.
- By default, the microflow policers do not affect the bridged traffic. To enable microflow policing of the bridged traffic, enter the **set qos bridged-microflow-policing** command (for more information, see the "Enabling or Disabling Microflow Policing of Bridged Traffic" section on page 52-62).
- With a Layer 3 Switching Engine II, to do any microflow policing, you must enable microflow policing of the bridged traffic.
- With an MSFC, QoS does not apply the microflow policers to the Multilayer Switching (MLS) candidate frames (MSFC2 does not use candidate and enabler frames).
- To avoid inconsistent results, all ACEs that include the same aggregate policer must use the same ACE keyword: **trust-dscp**, **trust-ipprec**, **trust-cos**, or **dscp**. If the ACE uses the **dscp** keyword, all traffic that matches the ACE must come through the ports that are configured with the same port keyword: **trust-dscp**, **trust-ipprec**, **trust-cos**, or **untrusted**. If the ACL is attached to a VLAN, you must configure all ports in the VLAN with the same port keyword.

PFC2 Policing Decisions

With PFC2, the policing decision consists of two levels:

- Normal Police Level—Set if either the microflow policer or the aggregate normal rate policer returns an out-of-profile decision.
- Excess Police Level—Set if the aggregate excess rate policer returns an out-of-profile decision.

The packets are dropped if the excess rate aggregate policer returns an out-of-profile decision and the drop indication flag is set or if the microflow policer returns an out-of-profile decision and the drop indication flag is set.

If an excess police level is set, the excess DSCP mapping is used to replace the original DSCP value with a marked-down value. If only a normal police level is set, the normal DSCP mapping is used. The excess police level has precedence for selecting the mapping rules when both police levels are set because the excess police level represents the worst out-of-profile transgression.

PFC3 Policing Decisions

In addition to PFC2 policing decisions, PFC3 supports egress QoS. These sections describe the PFC3 policing decisions:

- Policing Hardware-Forwarded LAN Traffic, page 52-26
- Policing Software-Forwarded LAN Traffic, page 52-26
- Policing Software-Forwarded WAN Traffic, page 52-26

Policing Hardware-Forwarded LAN Traffic

The hardware-forwarded LAN traffic (traffic that is forwarded by PFC3) can be subject to both an ingress and an egress policing rule. When the LAN traffic is subject to both an ingress and an egress policing rule, QoS evaluates both the rules simultaneously and applies the most severe rule. Because the policing rules are evaluated simultaneously, the markdown from an ingress policing rule is never used as the basis for the egress policing markdown.

Policing Software-Forwarded LAN Traffic

The software-forwarded LAN traffic (LAN traffic that is forwarded in the software by the MSFC) can be subject to both an ingress and an egress policing rule. When the software-forwarded traffic is subject to both an ingress and an egress policing rule, QoS evaluates the rules sequentially. The markdown from an ingress policing rule can be the basis for the egress policing markdown.

Policing Software-Forwarded WAN Traffic

PFC3 can provide egress QoS for the software-forwarded WAN traffic. The software-forwarded WAN traffic is subject only to an egress policing rule.

Attaching ACLs

You can configure each port for either port-based QoS (default) or VLAN-based QoS (see the "Enabling Port-Based or VLAN-Based QoS" section on page 52-40) and attach the ACLs to the selected interface (see the "Attaching an ACL to an Interface" section on page 52-56). You can attach up to three named ACLs, one of each type (IP, IPX, and Ethernet) to each port and VLAN.

On the ports that are configured for VLAN-based QoS, you can attach the named ACLs to the port's VLAN. For a trunk, you can attach the named ACLs to any VLANs that are allowed on the trunk as follows:

- On a port that is configured for VLAN-based QoS, the traffic that is received through the port is compared to any named ACLs that are attached to the port's VLAN. If you do not attach any named ACLs to the port's VLAN, or if the traffic does not match an ACE in a named ACL, QoS compares the traffic that is received through the port to the default ACLs.
- On a trunk that is configured for VLAN-based QoS, the traffic that is received through the port is compared to any named ACLs that are attached to the traffic's VLAN. For the traffic in the VLANs that have no named ACLs attached, or if the traffic does not match an ACE in a named ACL, QoS compares the traffic to the default ACLs.

On the ports that are configured for port-based QoS, you can attach the named ACLs to the port as follows:

- On a port that is configured for the port-based QoS, the traffic that is received through the port is compared to any named ACLs that are attached to the port. If you do not attach any named ACLs to the port, or if the traffic does not match an ACE in a named ACL, QoS compares the traffic that is received through the port to the default ACLs.
- On a trunk that is configured for port-based QoS, the traffic in all VLANs that are received through the port is compared to any named ACLs that are attached to the port. If you do not attach any named ACLs to the port, or if the traffic does not match an ACE in a named ACL, QoS compares the traffic that is received through the port to the default ACLs.

With PFC3, you can configure the ingress and egress QoS. To configure the ingress QoS, you attach the QoS ACLs to the ports and to the VLANs with the **input** keyword. To configure the egress QoS, you attach the QoS ACLs to the VLANs with the **output** keyword. The egress QoS does not use the port-based QoS (default) or VLAN-based QoS setting.

PFC3 Egress DSCP Mutation

PFC3 supports map-based egress DSCP mutation. You can configure up to 15 DSCP-to-DSCP mutation maps and apply the maps to the VLANs. QoS remarks the internal DSCP value in the egress traffic in the VLANs.

Final Layer 3 Switching Engine CoS and ToS Values

With a Layer 3 switching engine, QoS associates the CoS and ToS values with the traffic as specified by the marking rules and policers in the ACE that the traffic matches (see the "Internal DSCP Values" section on page 52-16). The associated CoS and ToS are used at the Ethernet egress port (see the "Ethernet Egress Port Scheduling, Congestion Avoidance, and Marking" section on page 52-28).

With PFC3A, you can configure QoS to use the received DSCP value in the egress ToS byte, instead of the DSCP that is created by marking and policing.

Classification and Marking on a Supervisor Engine 1 with a Layer 2 Switching Engine

On a Supervisor Engine 1 with a Layer 2 Switching Engine, QoS can classify the traffic that is addressed to the specified MAC address/VLAN pairs to be marked with a configured CoS value (for more information, see the "QoS Terminology" section on page 52-2 and the "Mapping a CoS Value to a Host Destination MAC Address/VLAN Pair" section on page 52-61).

Note Classification and marking on a Supervisor Engine 1 with a Layer 2 Switching Engine uses the Layer 2 CoS values. Classification and marking on a Supervisor Engine 1 with a Layer 2 Switching Engine does not use or set the Layer 3 IP precedence or DSCP values.

Ethernet Egress Port Scheduling, Congestion Avoidance, and Marking

These sections describe Ethernet egress port scheduling, congestion avoidance, and marking:

Overview

QoS schedules the traffic through the transmit queues based on the CoS values and uses the CoS-value-based transmit-queue drop thresholds to avoid congestion in the traffic that is transmitted from the Ethernet ports.

Note Ethernet egress port scheduling and congestion avoidance uses the Layer 2 CoS values. Ethernet egress port marking writes the Layer 2 CoS values, and for the IP traffic, the Layer 3 ToS byte.

Transmit Queues

Enter the **show port capabilities** command to see the queue structure of a port. The command displays one of the following:

- **tx-(2q2t)** indicates two standard queues, each with two configurable tail-drop thresholds.
- **tx-(1p2q1t)** indicates one strict-priority queue and two standard queues, each with one configurable WRED-drop threshold (on the **1p2q1t** ports, each standard queue also has one nonconfigurable tail-drop threshold).
- **tx-(1p2q2t)** indicates one strict-priority queue and two standard queues, each with two configurable WRED-drop thresholds.
- **tx-(1p3q1t)** indicates one strict-priority queue and three standard queues, each with one configurable WRED-drop threshold (on the **1p3q1t** ports, each standard queue also has one nonconfigurable tail-drop threshold).

- **tx-(1p3q8t)** indicates one strict-priority queue and three standard queues, each with eight configurable WRED-drop thresholds (on the **1p3q8t** ports, each standard queue also has one nonconfigurable tail-drop threshold).
- **tx-(1p7q8t)** indicates one strict-priority queue and seven standard queues, each with eight configurable WRED-drop thresholds (on the **1p7q8t** ports, each standard queue also has one nonconfigurable tail-drop threshold).

For the port types with a strict-priority queue, the switch services the traffic in the strict-priority transmit queue before servicing the standard queues. When the switch is servicing a standard queue, after transmitting a packet, it checks for the traffic in the strict-priority queue. If the switch detects the traffic in the strict-priority queue, it suspends its service of the standard queue and completes the service of all the traffic in the strict-priority queue before returning to the standard queue.

Scheduling and Congestion Avoidance

QoS implements the CoS-value-based transmit-queue drop thresholds to avoid congestion in the transmitted traffic. See the "QoS Default Configuration" section on page 52-30 for the default CoS-to-threshold mapping.

For some port types, you can configure each standard transmit queue to use both a nonconfigurable 100-percent tail-drop threshold and a configurable WRED-drop threshold (see the "1p3q1t Transmit Queues" section on page 52-70 and the "1p2q1t, 1p3q8t, and 1p7q8t Transmit Queues" section on page 52-72). The switch uses the tail-drop thresholds for the traffic carrying the CoS values that are mapped only to a queue. The switch uses the WRED-drop thresholds for the traffic carrying the CoS values that are mapped to a queue and a threshold.

Marking

When the traffic is transmitted from the switch, QoS writes the ToS byte into the IP traffic (Layer 3 switching engine only) and the CoS value that was used for scheduling and congestion avoidance into the ISL or 802.1Q traffic (for more information, see the "Final Layer 3 Switching Engine CoS and ToS Values" section on page 52-27).

QoS Statistics Data Export

QoS statistics data export generates per-port and per-aggregate policer utilization information and forwards this information in the UDP packets to traffic monitoring, planning, or accounting applications. You can enable QoS statistics data export per port or per aggregate policer. The statistics data that is generated per port consists of the counts of the input and output packets and bytes. The aggregate policer statistics consists of the counts of the allowed packets and the counts of the packets exceeding the policed rate.

The QoS statistics data collection occurs periodically at a fixed interval, but the interval at which the data is exported is configurable. QoS statistics collection is enabled by default, and data export is disabled by default for all the ports and all the aggregate policers that are configured on the Catalyst 6500 series switch.

Note

- Per-port counter information and utilization statistics are not available for the ATM ports.
- QoS statistics data export is completely separate from TopN and NetFlow Data Export and does not interact with either of these features.

For information on configuring QoS statistics data export, see the "Configuring QoS Statistics Data Export" section on page 52-89.

QoS Default Configuration

Table 52-3 shows the QoS default configuration.

Table 52-3 QoS Default Configuration

Feature	Default Value
QoS enable state	Disabled **Note**—With QoS enabled and all other QoS parameters at the default values, QoS sets Layer 3 DSCP to zero and Layer 2 CoS to zero in all traffic that is transmitted from the switch.
DSCP Rewrite	Enabled
Egress DSCP Mutation	Disabled
Port CoS value	0
IntraVLAN microflow policing	Disabled
CoS to internal DSCP map (internal DSCP set from CoS values)	CoS 0 = DSCP 0 CoS 1 = DSCP 8 CoS 2 = DSCP 16 CoS 3 = DSCP 24 CoS 4 = DSCP 32 CoS 5 = DSCP 40 CoS 6 = DSCP 48 CoS 7 = DSCP 56
IP precedence to internal DSCP map (internal DSCP set from IP precedence values)	IP precedence 0 = DSCP 0 IP precedence 1 = DSCP 8 IP precedence 2 = DSCP 16 IP precedence 3 = DSCP 24 IP precedence 4 = DSCP 32 IP precedence 5 = DSCP 40 IP precedence 6 = DSCP 48 IP precedence 7 = DSCP 56

Table 52-3 QoS Default Configuration (continued)

Feature	Default Value
Internal DSCP to egress CoS map (egress CoS set from internal DSCP values)	DSCP 0–7 = CoS 0 DSCP 8–15 = CoS 1 DSCP 16–23 = CoS 2 DSCP 24–31 = CoS 3 DSCP 32–39 = CoS 4 DSCP 40–47 = CoS 5 DSCP 48–55 = CoS 6 DSCP 56–63 = CoS 7
Marked-down DSCP from DSCP map	Marked-down DSCP value equals original DSCP value (no markdown)
Policers	None
Named ACLs	None
Default ACLs	Supports per-port classification and marking, sets DSCP to 0 in traffic from the untrusted ports, no policing
COPS[1] support	Disabled
RSVP support	Disabled
QoS statistics data export	Disabled
With QoS enabled	
Runtime—Port based or VLAN based	Port based
Config—Port based or VLAN based	Port based
Port trust state	Untrusted
2q2t transmit-queue size percentages	• Low priority: 80% • High priority: 20%
1p1q0t receive-queue size percentages	• Standard: 80% • Strict priority: 20%
1p2q2t transmit-queue size percentages	• Low priority: 70% • High priority: 15% • Strict priority: 15%
1p2q1t transmit-queue size percentages	• Low priority: 70% • High priority: 15% • Strict priority: 15%
1p3q8t transmit-queue size percentages	• Low priority: 65% • Medium priority: 15% • High priority: 15% • Strict priority: 5%

Table 52-3 QoS Default Configuration (continued)

Feature	Default Value
1p7q8t transmit-queue size percentages	• Standard queue 1 (lowest priority): 25% • Standard queue 2: 15% • Standard queue 3: 15% • Standard queue 4: 10% • Standard queue 5: 10% • Standard queue 6: 10% • Standard queue 7 (highest priority): 10% • Strict priority: 5%
1p3q8t standard transmit-queue low:medium:high priority bandwidth allocation ratio	20:100:200
1p7q8t standard transmit-queue lowest-to-highest priority bandwidth allocation ratio	10:20:30:40:40:70:70
1p2q1t standard transmit-queue low:high priority bandwidth allocation ratio	100:255
2q2t, **1p2q2t**, and **1p2q1t** standard transmit-queue low:high priority bandwidth allocation ratio	5:255
1p3q1t standard transmit-queue low:medium:high-priority bandwidth allocation ratio	100:150:200
1q4t/2q2t receive and transmit-queue CoS value/drop-threshold mapping	• Receive queue 1/drop threshold 1 (50%) and transmit queue 1/drop threshold 1 (80%): CoS 0 and 1 • Receive queue 1/drop threshold 2 (60%) and transmit queue 1/drop threshold 2 (100%): CoS 2 and 3 • Receive queue 1/drop threshold 3 (80%) and transmit queue 2/drop threshold 1 (80%): CoS 4 and 5 • Receive queue 1/drop threshold 4 (100%) and transmit queue 2/drop threshold 2 (100%): CoS 6 and 7
1q2t port receive-queue CoS value/drop-threshold mapping and threshold percentages	• Receive queue 1/drop threshold 1: – CoS 0, 1, 2, 3, and 4 – Drop threshold: 80% • Receive queue 1/drop threshold 2: – CoS 5, 6, and 7 – Drop threshold: 100% (not configurable) **Note** **1p2q2t** transmit queues same as **1p1q4t/1p2q2t**.

Table 52-3 QoS Default Configuration (continued)

Feature	Default Value
1p1q4t/1p2q2t port receive and transmit-queue CoS value/drop-threshold mapping and threshold percentages	• Strict-priority receive queue 1 and strict-priority transmit queue 1: CoS 5 • Receive queue 1/drop threshold 1 and transmit queue 1/drop threshold 1: – CoS 0 and 1 – Transmit queue low and high WRED-drop thresholds: 40% and 70% • Receive queue 1/drop threshold 2 and transmit queue 1/drop threshold 2: – CoS 2 and 3 – Transmit queue low and high WRED-drop thresholds: 70% and 100% • Receive queue 1/drop threshold 3 and transmit queue 2/drop threshold 1: – CoS 4 – Transmit queue low and high WRED-drop thresholds: 40% and 70% • Receive queue 1/drop threshold 4 and transmit queue 2/drop threshold 2: – CoS 6 and 7 – Transmit queue low and high WRED-drop thresholds: 70% and 100%
1p1q0t receive-queue CoS value mapping	• Receive queue 1 (standard) nonconfigurable 100% tail-drop threshold: CoS 0, 1, 2, 3, 4, 6, and 7 • Receive queue 2 (strict priority): CoS 5
1q8t receive-queue CoS value/drop-threshold mapping	• Threshold 1: 50% (CoS 0) • Threshold 2: 50% • Threshold 3: 60% (CoS 1, 2, 3, 4) • Threshold 4: 60% • Threshold 5: 80% (CoS 6 and 7) • Threshold 6: 80% • Threshold 7: 100% (CoS 5) • Threshold 8: 100%

Table 52-3 QoS Default Configuration (continued)

Feature	Default Value
1p3q8t transmit-queue CoS value/drop-threshold mapping	• Standard transmit queue 1 (low priority) low and high WRED-drop thresholds: – Threshold 1—70% and 100% (CoS 0) – Thresholds 2 through 8—100% and 100% • Standard transmit queue 2 (medium priority) low and high WRED-drop thresholds: – Threshold 1—70% and 100% (CoS 1 and 2) – Thresholds 2 through 8—100% and 100% • Standard transmit queue 3 (high priority) low and high WRED-drop thresholds: – Thresholds 1 and 2—40% and 70% – Thresholds 3 and 4—50% and 80% – Threshold 5—60% and 90% (CoS 3 and 4) – Threshold 6—60% and 90% – Threshold 7—70% and 100% (CoS 6 and 7) – Threshold 8—70% and 100% • Strict transmit queue 4: CoS 5

Table 52-3 QoS Default Configuration (continued)

Feature	Default Value
1p7q8t transmit-queue CoS value/drop-threshold mapping	• Standard transmit queue 1 (lowest priority) low and high WRED-drop thresholds: – Threshold 1—70% and 100% (CoS 0) – Thresholds 2 through 8—100% and 100% • Standard transmit queue 2 low and high WRED-drop thresholds: – Threshold 1—70% and 100% (CoS 1) – Thresholds 2 through 8—100% and 100% • Standard transmit queue 3 low and high WRED-drop thresholds: – Threshold 1—70% and 100% (CoS 2) – Thresholds 2 through 8—100% and 100% • Standard transmit queue 4 low and high WRED-drop thresholds: – Threshold 1—70% and 100% (CoS 3) – Thresholds 2 through 8—100% and 100% • Standard transmit queue 5 low and high WRED-drop thresholds: – Threshold 1—70% and 100% (CoS 4) – Thresholds 2 through 8—100% and 100% • Standard transmit queue 6 low and high WRED-drop thresholds: – Threshold 1—100% and 100% – Threshold 2—70% and 100% (CoS 6) – Thresholds 3 through 8—100% and 100% • Standard transmit queue 7 low and high WRED-drop thresholds: – Threshold 1—100% and 100% – Threshold 2—70% and 100% (CoS 7) – Thresholds 3 through 8—100% and 100% • Strict transmit queue 8: CoS 5
1p3q1t transmit-queue CoS value/drop-threshold mapping	• Standard transmit queue 1 (low priority) tail-drop threshold: – CoS 0 and 1 – Low and high WRED-drop threshold: 70% and 100% • Standard transmit queue 2 (medium priority) tail-drop threshold: – CoS 2, 3, and 4 – Low and high WRED-drop threshold: 70% and 100% • Standard transmit queue 3 (high priority) tail-drop threshold: – CoS 6 and 7 – Low and high WRED-drop threshold: 70% and 100% • Standard transmit queue 4 (strict priority): CoS 5

Table 52-3 QoS Default Configuration (continued)

Feature	Default Value
1p1q8t receive-queue port CoS value/drop-threshold mapping	• Receive queue 1 (standard) WRED-drop threshold: CoS 0, 1, 2, 3, 4, 6, and 7: – Drop threshold 1: CoS 0 Low WRED threshold: 40% High WRED-drop threshold: 70% – Drop threshold 2: CoS 1 Low WRED threshold: 40% High WRED-drop threshold: 70% – Drop threshold 3: CoS 2 Low WRED threshold: 50% High WRED-drop threshold: 80% – Drop threshold 4: CoS 3 Low WRED threshold: 50% High WRED-drop threshold: 80% – Drop threshold 5: CoS 4 Low WRED threshold: 60% High WRED-drop threshold: 90% – Drop threshold 6: CoS 6 Low WRED threshold: 60% High WRED-drop threshold: 90% – Drop threshold 6: CoS 7 Low WRED threshold: 70% High WRED-drop threshold: 100% • Receive queue 2 (strict priority): CoS 5
1p2q1t transmit-queue port CoS value/drop-threshold mapping	• Standard transmit queue 1 (low priority) WRED-drop threshold: – CoS 0, 1, 2, and 3 – Low WRED threshold: 70% – High WRED-drop threshold: 100% • Standard transmit queue 2 (high priority) WRED-drop threshold: – CoS 4, 6, or 7 – Low WRED threshold: 70% – High WRED-drop threshold: 100% • Strict transmit queue 3: CoS 5
With QoS disabled	
Runtime—Port based or VLAN based	VLAN based
Config—Port based or VLAN based	Port based
Port trust state	**trust-cos** (Layer 2 switching engine) **trust-dscp** (Layer 3 switching engine)
Receive-queue drop-threshold percentages	All thresholds set to 100%

Table 52-3 QoS Default Configuration (continued)

Feature	Default Value
Transmit-queue drop-threshold percentages	All thresholds set to 100%
Transmit-queue low-priority/high-priority bandwidth allocation ratio	255:1
Transmit-queue size ratio	• Low priority: 100% • High priority: Not used
CoS value/drop-threshold mapping	Receive-drop threshold 1 and transmit-queue 1/drop threshold 1: CoS 0–7

1. COPS = Common Open Policy Service

QoS Configuration Guidelines and Restrictions

QoS has the following hardware granularity for the committed information rate (CIR) and the peak information rate (PIR) values:

CIR and PIR Rate Value Range			Granularity	
1 to	2097152	(2 Mbs)	65536	(64 Kb)
2097153 to	4194304	(4 Mbs)	131072	(128 Kb)
4194305 to	8388608	(8 Mbs)	262144	(256 Kb)
8388609 to	16777216	(16 Mbs)	524288	(512 Kb)
16777217 to	33554432	(32 Mbs)	1048576	(1 Mb)
33554433 to	67108864	(64 Mbs)	2097152	(2 Mb)
67108865 to	134217728	(128 Mbs)	4194304	(4 Mb)
134217729 to	268435456	(256 Mbs)	8388608	(8 Mb)
268435457 to	536870912	(512 Mbs)	16777216	(16 Mb)
536870913 to	1073741824	(1 Gps)	33554432	(32 Mb)
1073741825 to	2147483648	(2 Gps)	67108864	(64 Mb)
2147483649 to	4294967296	(4 Gps)	134217728	(128 Mb)
4294967297 to	8000000000	(8 Gps)	268435456	(256 Mb)

Within each range, PFC QoS programs the PFC hardware with the rate values that are multiples of the granularity values.

QoS has the following hardware granularity for the CIR and PIR token bucket (burst) sizes:

CIR and PIR Token Bucket Size Range	Granularity
1 to 32768 (32 KB)	1024 (1 KB)
32769 to 65536 (64 KB)	2048 (2 KB)
65537 to 131072 (128 KB)	4096 (4 KB)
131073 to 262144 (256 KB)	8192 (8 KB)
262145 to 524288 (512 KB)	16384 (16 KB)
524289 to 1048576 (1 MB)	32768 (32 KB)
1048577 to 2097152 (2 MB)	65536 (64 KB)
2097153 to 4194304 (4 MB)	131072 (128 KB)
4194305 to 8388608 (8 MB)	262144 (256 KB)
8388609 to 16777216 (16 MB)	524288 (512 KB)
16777217 to 33554432 (32 MB)	1048576 (1 MB)

Configuring QoS on the Switch

These sections describe how to configure QoS on the Catalyst 6500 series switches:

Note Some QoS **show** commands support the **config** and **runtime** keywords. Use the **runtime** keyword to display the QoS values that are currently programmed into the hardware. When you disable QoS, the display with the **runtime** keyword is "QoS is disabled." Use the **config** keyword to display the values from the commands that have been entered but which may not currently be programmed into the hardware (for example, the locally configured QoS values that are currently not used because COPS has been selected as the QoS policy source or the QoS values that are configured when QoS is disabled).

Enabling QoS

To enable QoS, perform this task in privileged mode:

Task	Command
Enable QoS on the switch.	**set qos** {**enable** | **disable**}

This example shows how to enable QoS:

```
Console> (enable) set qos enable
QoS is enabled.
Console> (enable)
```

Enabling DSCP Rewrite

Note Only PFC3 supports the configuration commands in this section.

To enable DSCP rewrite, which uses the DSCP value from marking and policing as the egress DSCP value, perform this task in privileged mode:

Task	Command
Enable DSCP rewrite on the switch.	**set qos dscp-rewrite enable**

This example shows how to enable DSCP rewrite:

```
Console> (enable) set qos dscp-rewrite enable
DSCP rewrite has been globally enabled.
Console> (enable)
```

Disabling DSCP Rewrite

Note Only PFC3 supports the configuration commands in this section.

To disable DSCP rewrite, which uses the received DSCP value as the egress DSCP value, perform this task in privileged mode:

Task	Command
Disable DSCP rewrite on the switch.	**set qos dscp-rewrite disable**

This example shows how to disable DSCP rewrite:

```
Console> (enable) set qos dscp-rewrite disable
DSCP rewrite has been globally disabled.
Console> (enable)
```

Enabling Port-Based or VLAN-Based QoS

Note Supervisor Engine 1 with a Layer 2 Switching Engine does not support the commands in this section.

By default, QoS uses the ACLs that are attached to the ports. On a per-port basis, you can configure QoS to use the ACLs that are attached to a VLAN. To enable VLAN-based QoS on a port, perform this task in privileged mode:

	Task	Command
Step 1	Enable VLAN-based QoS on a port.	**set port qos** *mod/port* {**port-based** | **vlan-based**}
Step 2	Verify the configuration.	**show port qos** *mod/port*

For more information, see the "Attaching ACLs" section on page 52-26.

This example shows how to enable the VLAN-based QoS on a port:

```
Console> (enable) set port qos 1/1-2 vlan-based
Hardware programming in progress...
QoS interface is set to vlan-based for ports 1/1-2.
Console> (enable)
```

Changing a port from port-based to VLAN-based QoS detaches all ACLs from the port. Any ACLs that are attached to the VLAN apply to the port immediately (for more information, see the "Attaching an ACL to an Interface" section on page 52-56).

Configuring the Trust State of a Port

This command configures the trust state of a port (for more information, see the "Ethernet Ingress Port Marking, Scheduling, Congestion Avoidance, and Classification" section on page 52-12). By default, all ports are untrusted.

To configure the trust state of a port, perform this task in privileged mode:

	Task	Command
Step 1	Configure the trust state of a port.	**set port qos trust** {**untrusted** | **trust-cos** | **trust-ipprec** | **trust-dscp**}
Step 2	Verify the configuration.	**show port qos**

Note the following syntax guidelines when configuring the trust state of a port:

- The **trust-ipprec** and **trust-dscp** keywords are supported only with a Layer 3 switching engine.
- **1q4t** ports (except Gigabit Ethernet) do not support the **trust-ipprec** and **trust-dscp** port keywords. You must configure a **trust-ipprec** or **trust-dscp** ACL that matches the ingress traffic to apply the **trust-ipprec** or **trust-dscp** trust state.
- On the **1q4t** ports (except Gigabit Ethernet), the **trust-cos** port keyword displays an error message, activates the receive-queue drop thresholds, and—as indicated by the error message—does not apply the **trust-cos** trust state to the traffic. You must configure a **trust-cos** ACL that matches the ingress traffic to apply the **trust-cos** trust state.

This example shows how to configure port 1/1 with the **trust-cos** keyword:

```
Console> (enable) set port qos 1/1 trust trust-cos
Port 1/1 qos set to trust-cos
Console> (enable)
```

Note Only the ISL or 802.1Q frames carry the CoS values. Configure the ports with the **trust-cos** keyword only when the received traffic is ISL or 802.1Q frames carrying the CoS values that you know to be consistent with the network policy.

Configuring the CoS Value for a Port

Note Whether or not QoS uses the CoS value that is applied with the **set port qos ...cos** command depends on the trust state of the port and the trust state of the traffic that is received through the port. The **set port qos ... cos** command does not configure the trust state of the port or the trust state of the traffic that is received through the port. To use the CoS value that is applied with the **set port qos ... cos** command, configure a trust-CoS ACL that matches the ingress traffic; or for a port that receives no tagged traffic, configure the port to trust CoS.

The unmarked frames from the ports that are configured as trusted and all frames from the ports that are configured as untrusted are assigned the CoS value that is specified with this command.

To configure the CoS value for a port, perform this task in privileged mode:

	Task	Command
Step 1	Configure the CoS value for a port.	**set port qos cos** *cos_value*
Step 2	Verify the configuration.	**show port qos**

This example shows how to configure the port CoS value to 3 for port 1/1:

```
Console> (enable) set port qos 1/1 cos 3
Port 1/1 qos cos set to 3
Console> (enable)
```

To revert to the default CoS value for a port, perform this task in privileged mode:

	Task	Command
Step 1	Revert to the default CoS value for a port.	**clear port qos cos**
Step 2	Verify the configuration.	**show port qos**

This example shows how to revert to the default CoS value for port 1/1:

```
Console> (enable) clear port qos 1/1 cos
Port 1/1 qos cos setting cleared.
Console> (enable)
```

Creating Policers

Note Supervisor Engine 1 with a Layer 2 Switching Engine does not support the commands in this section.

To create a policer, perform this task in privileged mode:

	Task	Command
Step 1	Create a policer.	**set qos policer microflow** *microflow_name* {**rate** *rate*} {**burst** *burst_value*} {**drop** | **policed-dscp**} With PFC or PFC2: **set qos policer aggregate** *aggregate_name* {**rate** *rate*} {**burst** *burst_value*} {**drop** | **policed-dscp**} With PFC2 or PFC3A: **set qos policer aggregate** *aggregate_name* {**rate** *rate*} **policed-dscp** {**erate** *erate_value*} {**drop** | **policed-dscp**} **burst** *burst_value* [**eburst** *eburst_value*]
Step 2	Verify the configuration.	**show qos policer** {**config** | **runtime**} {**microflow** | **aggregate** | **all**}

For more information, see the "Policers" section on page 52-24 and the "PFC2 Policing Decisions" section on page 52-25.

The *policer_name* parameter can be up to 31 characters, is case sensitive, and may include a–z, A–Z, 0–9, the dash character (-), the underscore character (_), and the period character (.).The policer names must start with an alphabetic character (not a digit) and must be unique across all microflow and aggregate policers. You cannot use the keywords from any command as a policer name.

The valid values for the *rate* and *erate* parameters are 32 Kbps (entered as 32) to 32 Gbps (entered as 32000000). To classify all traffic as out of profile, set the *rate* parameter to zero (0). Set the *erate* parameter to a higher value than the *rate* parameter. PFC1 and PFC2 have the following hardware granularity for rate values:

Rate Value Range	Granularity	Rate Value Range	Granularity
1 to 1000 (1 Mbs)	32768 (32 K)	64001 to 128000 (128 Mbs)	4194304 (4 M)
1001 to 2000 (2 Mbs)	65536 (64 K)	128001 to 256000 (256 Mbs)	8388608 (8 M)
2001 to 4000 (4 Mbs)	131072 (128 K)	256001 to 512000 (512 Mbs)	16777216 (16 M)
4001 to 8000 (8 Mbs)	262144 (256 K)	512001 to 1024000 (1 Gps)	33554432 (32 M)
8001 to 16000 (16 Mbs)	524288 (512 K)	1024001 to 2048000 (2 Gps)	67108864 (64 M)
16001 to 32000 (32 Mbs)	1048576 (1 M)	2048001 to 4096000 (4 Gps)	134217728 (128 M)
32001 to 64000 (64 Mbs)	2097152 (2 M)	4096001 to 8192000 (8 Gps)	268435456 (256 M)

Within each range, QoS programs the hardware with the rate values that are multiples of the granularity values.

The valid values for the *burst* and *eburst* parameters are 1 Kb (entered as 1) to 256 Mb (entered as 256000). When configuring the *burst* and *eburst* parameters, note the following:

- The **burst** keyword, the *burst_value* parameter, the optional **eburst** keyword, and the *eburst_value* parameter set the token bucket sizes.
- The token bucket size defines the maximum number of the in-profile bytes that can be transmitted every 0.25 millisecond.
- To sustain a specific rate, set the token bucket size to be at least the *rate* divided by 4000, because the tokens are removed from the bucket every 1/4000th of a second (0.25 millisecond) and the bucket needs to be at least as large as the burst size to sustain the specified rate.
- If you do not enter the **eburst** keyword and the *eburst_value* parameter, QoS sets both token buckets to the size that is configured with the **burst** keyword and the *burst_value* parameter.
- Because any packet larger than the burst size is considered an out-of-profile packet, make sure that the burst size is greater than or equal to the largest packet size being policed.
- QoS programs the hardware with the values that are multiples of 32K (32,768), not with the specific value entered.

Enter either the **drop** keyword to cause all out-of-profile packets to be dropped or the **policed-dscp** keyword to cause all out-of-profile packets with the normal rate to be marked down as specified in the normal markdown DSCP map (for more information, see the "Mapping the DSCP Markdown Values" section on page 52-75).

This example shows how to create a microflow policer with a 1-Mbps rate limit and a 10-Mb burst limit that marks down out-of-profile traffic:

```
Console> (enable) set qos policer microflow my-micro rate 1000 burst 10000 policed-dscp
Hardware programming in progress...
QoS policer for microflow my-micro created successfully.
Console> (enable)
```

For PFC2 or PFC3A, this example shows how to create an aggregate excess rate policer with a 64-Kbps rate limit and a 128-Kb burst limit that drops the traffic exceeding these values:

```
Console> (enable) set qos policer aggregate test rate 64 burst 128 drop
QoS policer for aggregate test created successfully.
Console> (enable) show qos policer config aggregate test
QoS aggregate policers:
QoS aggregate policers:
Aggregate name                Normal rate (kbps) Burst size (kb) Normal action
----------------------------- ------------------ --------------- -------------
test                                        64          128    policed-dscp
                              Excess rate (kbps) Burst size (kb) Excess action
                              ------------------ --------------- -------------
                                            64          128    drop
                              ACL attached
                              -----------------------------------

Console> (enable)
```

For PFC2 or PFC3A, this example shows how to create an aggregate excess rate policer with a 64-Kbps rate limit and a 100-Kb burst limit that will cause all out-of-profile packets to be marked down as specified in the normal markdown DSCP map:

```
Console> (enable) set qos policer aggregate test2 rate 64 burst 100 policed-dscp
QoS policer for aggregate test2 created successfully.
Console> (enable) show qos policer config aggregate test2
QoS aggregate policers:
Aggregate name                Normal rate (kbps)  Burst size (kb) Normal action
----------------------------- ------------------  --------------- -------------
test2                                       64             100  policed-dscp
                              Excess rate (kbps)  Burst size (kb) Excess action
                              ------------------  --------------  ---------------
                                         8000000             100  policed-dscp
                              ACL attached
                              -----------------------------------


Console> (enable)
```

For PFC2 or PFC3A, this example shows how to create an aggregate excess rate policer with a 64-Kbps rate limit and a 128-Kb burst limit that will cause the traffic that exceeds the normal rate of 64 Kbps and a burst size of 96 Kb to be marked down as specified in the normal markdown DSCP map, and the traffic that exceeds 128 Kbps and a burst size of 96 Kb to be dropped:

```
Console> (enable) set qos policer aggregate test3 rate 64 policed-dscp erate 128 drop
burst 96
QoS policer for aggregate test3 created successfully.
Console> (enable) show qos policer config aggregate test3
QoS aggregate policers:
Aggregate name                Normal rate (kbps)  Burst size (kb) Normal action
----------------------------- ------------------  --------------- -------------
test3                                       64              96 policed-dscp
                              Excess rate (kbps)  Burst size (kb) Excess action
                              ------------------  --------------- ---------------
                                            128              96 drop
                              ACL attached
                              -----------------------------------


Console> (enable)
```

Deleting Policers

Note You can delete the policers only if they are not attached to any interfaces (for more information, see the "Detaching an ACL from an Interface" section on page 52-57).

To delete one or all policers, perform this task in privileged mode:

	Task	Command
Step 1	Delete one or all policers.	**clear qos policer** {**microflow** | **aggregate**} {*policer_name* | **all**}
Step 2	Verify the configuration.	**show qos policer** {**config** | **runtime**} {**microflow** | **aggregate** | **all**}

This example shows how to delete the microflow policer named my_micro:

```
Console> (enable) clear qos policer microflow my_micro
my_micro QoS microflow policer cleared.
Console> (enable)
```

Creating or Modifying ACLs

Note Supervisor Engine 1 with a Layer 2 Switching Engine does not support the commands in this section.

These sections describe how to create and modify the ACLs:

ACL Names

The ACL names can be up to 31 characters, are case sensitive, and may include a–z, A–Z, 0–9, the dash character (-), the underscore character (_), and the period character (.). The ACL names must start with an alphabetic character and must be unique across all QoS ACLs of all types. You cannot use the keywords from any command as an ACL name.

ACE Name, Marking Rule, Policing, and Filtering Syntax

The ACE command syntax is organized as follows:
ACL_command ACL_type_and_name marking_rule policing_rule filtering

For example, in an IP ACE, the command syntax is as follows:

set qos acl ip *acl_name* {**dscp** *dscp_value* | **trust-cos** | **trust-ipprec** | **trust-dscp**} [**microflow** *microflow_name*] [**aggregate** *aggregate_name*] *src_ip_spec* [**precedence** *precedence* | **dscp-field** *dscp*] [**before** *editbuffer_index* | **modify** *editbuffer_index*]

- **set qos acl ip** *acl_name*—Creates a named ACL of the specified type or adds the ACE to the ACL if it already exists. See the "ACL Names" section on page 52-45.
- {**dscp** *dscp_value* | **trust-cos** | **trust-ipprec** | **trust-dscp**}—Selects a marking rule. See the "Marking Rules" section on page 52-23.
- [**microflow** *microflow_name*] [**aggregate** *aggregate_name*]—Optionally configures policing in the ACE. See the "Policers" section on page 52-24.
- *src_ip_spec* [**precedence** *precedence* | **dscp-field** *dscp*]—The rest of the parameters, except the **editbuffer** keywords, configure filtering.

Named IP ACLs

These sections describe how to create or modify the IP ACLs:

Source and Destination IP Addresses and Masks

In the IP ACEs, specify the source and destination IP addresses and masks (represented by the *src_ip_spec* and *dest_ip_spec* parameters in the following sections) in the form *ip_address mask*. The mask is mandatory. Use one bits, which need not be contiguous, where you want the wildcards.

Use any of the following formats for the address and mask:

- Four-part dotted-decimal 32-bit values
- The keyword **any** as an abbreviation for a wildcard address and wildcard mask of 0.0.0.0 255.255.255.255
- The abbreviation **host** *ip_address* for an address and wildcard mask of *ip_address* 0.0.0.0

Port Operator Parameters

In the IP ACEs, the *operator* parameter can be one of the following:

- **lt** (less than)
- **gt** (greater than)
- **eq** (equal)
- **neq** (not equal)
- **range** (with a pair of port parameters)

See the "Layer 4 Operations Configuration Guidelines" section on page 15-23 for restrictions that apply to the QoS ACLs.

Precedence Parameter Options

For the *precedence* parameter keyword options in the IP ACEs, see the "IP ACE Layer 3 Classification Criteria" section on page 52-18.

IP ACEs for TCP Traffic

To create or modify an IP ACE for the TCP traffic, perform this task in privileged mode:

	Task	Command
Step 1	Create or modify an IP ACE for the TCP traffic.	**set qos acl ip** {*acl_name*} {{**dscp** *dscp_value*} | **trust-cos** | **trust-ipprec** | **trust-dscp**} [**microflow** *microflow_name*] [**aggregate** *aggregate_name*] **tcp** {*src_ip_spec*} [{*operator*} {*port*} [*port*]] {*dest_ip_spec*} [{*operator*} {*port*} [*port*]] [**established**] [**precedence** *precedence_value* | **dscp-field** *dscp*] [**before** *editbuffer_index* | **modify** *editbuffer_index*]
Step 2	Verify the configuration.	**show qos acl info** {**acl_name** | **all**} **editbuffer** [*editbuffer_index*]

For the *port* parameter keyword options, see the "IP ACE Layer 4 TCP Classification Criteria" section on page 52-19.

The **established** keyword matches the traffic with the ACK or RST bits set.

This example shows how to create an IP ACE for the TCP traffic:

```
Console> (enable) set qos acl ip my_IPacl trust-ipprec microflow my-micro aggregate my-agg
tcp any any
my_IPacl editbuffer modified. Use 'commit' command to apply changes.
Console> (enable)
```

IP ACEs for UDP Traffic

To create or modify an IP ACE for the UDP traffic, perform this task in privileged mode:

	Task	Command
Step 1	Create or modify an IP ACE for the UDP traffic.	**set qos acl ip** {*acl_name*} {{**dscp** *dscp_value*} | **trust-cos** | **trust-ipprec** | **trust-dscp**} [**microflow** *microflow_name*] [**aggregate** *aggregate_name*] **udp** {*src_ip_spec*} [{*operator*} {*port*} [*port*]] {*dest_ip_spec*} [{*operator*} {*port*} [*port*]] [**precedence** *precedence_value* | **dscp-field** *dscp*] [**before** *editbuffer_index* | **modify** *editbuffer_index*]
Step 2	Verify the configuration.	**show qos acl info** {**acl_name** | **all**} **editbuffer** [*editbuffer_index*]

For the *port* parameter keyword options, see the "IP ACE Layer 4 UDP Classification Criteria" section on page 52-19.

This example shows how to create an IP ACE for the UDP traffic:

```
Console> (enable) set qos acl ip my_IPacl trust-ipprec microflow my-micro aggregate my-agg
udp any any
my_IPacl editbuffer modified. Use 'commit' command to apply changes.
Console> (enable)
```

IP ACEs for ICMP Traffic

To create or modify an IP ACE for the ICMP traffic, perform this task in privileged mode:

	Task	Command
Step 1	Create or modify an IP ACE for the ICMP traffic.	**set qos acl ip** *acl_name* {**dscp** *dscp* | **trust-cos** | **trust-ipprec** | **trust-dscp**} [**microflow** *microflow_name*] [**aggregate** *aggregate_name*] **icmp** *src_ip_spec dest_ip_spec* [*icmp_type* [*icmp_code*] | *icmp_message*] [**precedence** *precedence* | **dscp-field** *dscp*] [**before** *editbuffer_index* | **modify** *editbuffer_index*]
Step 2	Verify the configuration.	**show qos acl info** {**acl_name** | **all**} **editbuffer** [*editbuffer_index*]

For the *icmp_code* and *icmp_type* parameter keyword options, see the "IP ACE Layer 4 ICMP Classification Criteria" section on page 52-19.

This example shows how to create an IP ACE for ICMP *echo* traffic:

```
Console> (enable) set qos acl ip my_IPacl trust-ipprec microflow my-micro aggregate my-agg
icmp any any echo
my_IPacl editbuffer modified. Use 'commit' command to apply changes.
Console> (enable)
```

IP ACEs for IGMP Traffic

Note QoS does not support the IGMP traffic when IGMP snooping is enabled.

To create or modify an IP ACE for the IGMP traffic, perform this task in privileged mode:

	Task	Command
Step 1	Create or modify an IP ACE for the IGMP traffic.	**set qos acl ip** *acl_name* {**dscp** *dscp_value* | **trust-cos** | **trust-ipprec** | **trust-dscp**} [**microflow** *microflow_name*] [**aggregate** *aggregate_name*] **igmp** *src_ip_spec dest_ip_spec* [*igmp_type*] [**precedence** *precedence_value* | **dscp-field** *dscp*] [**before** *editbuffer_index* | **modify** *editbuffer_index*]
Step 2	Verify the configuration.	**show qos acl info** {**acl_name** | **all**} **editbuffer** [*editbuffer_index*]

For the *igmp_type* parameter keyword options, see the "IP ACE Layer 4 IGMP Classification Criteria" section on page 52-21.

This example shows how to create an IP ACE for the IGMP Protocol Independent Multicast (PIM) traffic:

```
Console> (enable) set qos acl ip my_IPacl trust-ipprec microflow my-micro aggregate my-agg
igmp any any pim
my_IPacl editbuffer modified. Use 'commit' command to apply changes.
Console> (enable)
```

IP ACLs for Other Layer 4 Protocols

To create or modify a named IP ACL with additional parameters that match all Layer 4 protocols, perform this task in privileged mode:

	Task	Command
Step 1	Create or modify an IP ACE.	**set qos acl ip** *acl_name* {**dscp** *dscp_value* | **trust-cos** | **trust-ipprec** | **trust-dscp**} [**microflow** *microflow_name*] [**aggregate** *aggregate_name*] *protocol src_ip_spec dest_ip_spec* [**precedence** *precedence_value* | **dscp-field** *dscp*] [**before** *editbuffer_index* | **modify** *editbuffer_index*]
Step 2	Verify the configuration.	**show qos acl info** {**acl_name** | **all**} **editbuffer** [*editbuffer_index*]

Note With software Release 8.3(1) and later, the ACLs with the **output** keyword applied also support the **trust-cos** and **trust-ipprec** keywords.

For the *protocol* parameter keyword options, see the "IP ACE Layer 4 Protocol Classification Criteria" section on page 52-18.

This example shows how to create an IP ACE for the IPINIP traffic:

```
Console> (enable) set qos acl ip my_IPacl trust-ipprec microflow my-micro aggregate my-agg
ipinip any any
my_IPacl editbuffer modified. Use 'commit' command to apply changes.
Console> (enable)
```

IP ACEs for Any IP Traffic

To create or modify an IP ACE that matches all IP traffic, perform this task in privileged mode:

	Task	Command
Step 1	Create or modify an IP ACE.	**set qos acl ip** *acl_name* {**dscp** *dscp* | **trust-cos** | **trust-ipprec** | **trust-dscp**} [**microflow** *microflow_name*] [**aggregate** *aggregate_name*] *src_ip_spec* [**precedence** *precedence* | **dscp-field** *dscp*] [**before** *editbuffer_index* | **modify** *editbuffer_index*]
Step 2	Verify the configuration.	**show qos acl info** {**acl_name** | **all**} **editbuffer** [*editbuffer_index*]

This example shows how to create an IP ACE:

```
Console> (enable) set qos acl ip my_IPacl trust-ipprec microflow my-micro aggregate my-agg
any
my_IPacl editbuffer modified. Use 'commit' command to apply changes.
Console> (enable)
```

Modifying the Default IP ACLs

These sections describe how to modify the default IP ACLs:

- Modifying the Default IP Ingress ACL, page 52-50
- Modifying the Default IP Egress ACL, page 52-51

Modifying the Default IP Ingress ACL

To modify the default IP ingress ACL, perform this task in privileged mode:

	Task	Command
Step 1	Modify the default IP ACL.	**set qos acl default-action ip** {**dscp** *dscp* | **trust-cos** | **trust-ipprec** | **trust-dscp**} [**microflow** *microflow_name*] [**aggregate** *aggregate_name*] [**input**]
Step 2	Verify the configuration.	**show qos acl info default-action** {**ip** | **ipx** | **mac** | **all**}

Note Only PFC3 supports the **input** keyword.

For more information, see the "Default ACLs" section on page 52-22.

This example shows how to modify the default IP ACL:

```
Console> (enable) set qos acl default-action ip dscp 5 microflow my-micro aggregate my-agg
QoS default-action for IP ACL is set successfully.
Console> (enable)
```

Modifying the Default IP Egress ACL

Note Only PFC3 supports the configuration commands in this section.

To modify the default IP egress ACL, perform this task in privileged mode:

	Task	Command
Step 1	Modify the default IP ACL.	**set qos acl default-action ip** {**dscp** *dscp* | **trust-dscp**} [**aggregate** *aggregate_name*] **output**
Step 2	Verify the configuration.	**show qos acl info default-action ip**

Note In the default egress ACL, the **trust-dscp** keywords cause the ACL to trust the internal DSCP value, not a received DSCP value (see the "Internal DSCP Values" section on page 52-16).

For more information, see the "Default ACLs" section on page 52-22.

This example shows how to modify the default IP ACL:

```
Console> (enable) set qos acl default-action ip dscp 5 microflow my-micro aggregate my-agg
QoS default-action for IP ACL is set successfully.
Console> (enable)
```

Creating or Modifying the Named IPX ACLs

Note PFC3 does not provide QoS for the IPX traffic. See the "MAC ACE Layer 2 Classification Criteria" section on page 52-21 for information about how to use the MAC ACLs to filter the IPX traffic.

To create or modify a named IPX ACL, perform this task in privileged mode:

	Task	Command
Step 1	Create or modify a named IPX ACL.	With a PFC: **set qos acl ipx** *acl_name* {**dscp** *dscp_value* | **trust-cos**} [**aggregate** *aggregate_name*] *protocol src_net* [*dest_net*[.*dest_node*] [[*dest_net_mask*] .*dest_node_mask*]] [**before** *editbuffer_index* | **modify** *editbuffer_index*] With PFC2: **set qos acl ipx** *acl_name* **aggregate** *aggregate_name protocol src_net* [*dest_net*[.*dest_node*] [[*dest_net_mask*] .*dest_node_mask*]] [**before** *editbuffer_index* | **modify** *editbuffer_index*]
Step 2	Verify the configuration.	**show qos acl info** {**acl_name** | **all**} **editbuffer** [*editbuffer_index*]

The *protocol* parameter can be specified numerically (0–255) or with these keywords: **any**, **ncp** (17), **netbios** (20), **rip** (1), **sap** (4), or **spx** (5).

The *src_net* and *dest_net* parameters are IPX network numbers, entered as up to 8 hexadecimal digits in the range 1 to FFFFFFFE (-1 matches any network number). You do not need to enter leading zeros.

If you specify an IPX destination network, the IPX ACEs support the following optional parameters:

- An IPX destination network mask, entered as up to 8 hexadecimal digits in the range 1 to FFFFFFFE (-1 matches any network number). Use one bits, which need not be contiguous, where you want the wildcards.
- An IPX destination node, entered as 12 hexadecimal digits (48 bits), formatted as a dotted triplet of four-digit hexadecimal digits each (xxxx.xxxx.xxxx).
- If you specify an IPX destination node, the IPX ACEs support an IPX destination node mask, entered as 12 hexadecimal digits (48 bits), formatted as a dotted triplet of four-digit hexadecimal digits each (xxxx.xxxx.xxxx). Use one bits, which need not be contiguous, where you want the wildcards.

This example shows how to create an IPX ACE:

```
Console> (enable) set qos acl ipx my_IPXacl trust-cos aggregate my-agg -1
my_IPXacl editbuffer modified. Use 'commit' command to apply changes.
Console> (enable)
```

Creating or Modifying the Named MAC ACLs

To create or modify a named MAC ACL, perform this task in privileged mode:

	Task	Command
Step 1	Create or modify a named MAC ACL.	With a PFC, PFC2, or PFC3: **set qos acl mac** *acl_name* {**dscp** *dscp_value* | **trust-cos**} [**aggregate** *aggregate_name*] *src_mac_spec dest_mac_spec* [*ethertype*] [**before** *editbuffer_index* | **modify** *editbuffer_index*]
Step 2	Verify the configuration.	**show qos acl info** {**acl_name** | **all**} **editbuffer** [*editbuffer_index*]

Enter the *src_mac_spec* and *dest_mac_spec* parameters as a MAC address and a mask. Each parameter is 12 hexadecimal digits (48 bits), formatted as dash-separated pairs. Use one bits, which need not be contiguous, where you want the wildcards. Use the **any** keyword for a MAC address and mask of 0-0-0-0-0-0 ff-ff-ff-ff-ff-ff. Use the **host** keyword with a MAC address to specify an all-zero mask (*mac_address* 0-0-0-0-0-0).

Enter the *ethertype* parameter as 4 hexadecimal digits (16 bits) prefaced with **0x** (for example, 0x0600) or as a keyword (see the "MAC ACE Layer 2 Classification Criteria" section on page 52-21).

This example shows how to create a MAC ACE:

```
Console> (enable) set qos acl mac my_MACacl trust-cos aggregate my-agg any any
my_MACacl editbuffer modified. Use 'commit' command to apply changes.
Console> (enable)
```

Note The QoS MAC ACLs that do not include an EtherType parameter match the traffic with any value in the EtherType field, which allows the MAC-level QoS to be applied to any traffic except IP and IPX.

Creating or Modifying the Default IPX and MAC ACLs

Note PFC3 does not provide QoS for IPX traffic.

To create or modify the default IPX or MAC ACL, perform this task in privileged mode:

	Task	Command
Step 1	Create or modify the default IPX or MAC ACL.	With a PFC: **set qos acl default-action** {**ipx** | **mac**} {**dscp** *dscp* | **trust-cos**} [**aggregate** *aggregate_name*] With PFC2: **set qos acl default-action** {**ipx** | **mac**} **aggregate** *aggregate_name* With PFC3: **set qos acl default-action mac aggregate** *aggregate_name*
Step 2	Verify the configuration.	**show qos acl info default-action** {**ip** | **ipx** | **mac** | **all**}

For more information, see the "Default ACLs" section on page 52-22.

This example shows how to modify the default IPX ACL:

```
Console> (enable) set qos acl default-action ipx dscp 5 aggregate my-agg
QoS default-action for IPX ACL is set successfully.
Console> (enable)
```

Note The IPX and MAC ACLs do not support the microflow policers.

Deleting a Named ACL

To delete a named ACL, perform this task in privileged mode:

	Task	Command
Step 1	Delete a named ACL.	**clear qos acl** *acl_name* [*editbuffer_index*]
Step 2	Verify the configuration.	**show qos acl info** {*acl_name* | **all**}

This example shows how to delete the ACL named icmp_acl:

```
Console> (enable) clear qos acl icmp_acl 1
ACL icmp_acl ACE# 1 is deleted.
icmp_acl editbuffer modified. Use 'commit' command to apply changes.
Console> (enable)
```

Reverting to the Default Values in the Default ACLs

To revert to the default values for a default ACL, perform this task in privileged mode:

	Task	Command
Step 1	Revert to the default values for a default ACL.	**clear qos acl default-action** {**ip** | **ipx** | **mac**} [**tx**]
Step 2	Verify the configuration.	**show qos acl info default-action** {**ip** | **ipx** | **mac** | **all**}

This example shows how to revert to the default values for the default IP ACL:

```
Console> (enable) clear qos acl default-action ip
Hardware programming in progress...
QoS default-action for IP ACL is restored to default setting.
Console> (enable)
```

Discarding an Uncommitted ACL

To discard an uncommitted new ACL or uncommitted changes to an existing ACL, perform this task in privileged mode:

	Task	Command
Step 1	Discard an uncommitted ACL.	**rollback qos acl** {*acl_name* | **all**}
Step 2	If you discarded changes to an existing ACL, verify the configuration.	**show qos acl info** {*acl_name* | **all**}

This example shows how to discard an uncommitted ACL named my_acl:

```
Console> (enable) rollback qos acl my_acl
Rollback for QoS ACL my_acl is successful.
Console> (enable)
```

Note The changes to the default ACLs take effect immediately and cannot be discarded.

Committing an ACL

When you create, change, or delete a named ACL, the changes exist temporarily in an edit buffer in memory. To commit the ACL so that it can be used, perform this task in privileged mode:

	Task	Command
Step 1	Commit an ACL.	**commit qos acl** *acl_name*
Step 2	Verify the configuration.	**show config qos acl** {*acl_name* | **all**}

This example shows how to commit an ACL named my_acl:

```
Console> (enable) commit qos acl my_acl
Hardware programming in progress...
ACL my_acl is committed to hardware.
Console> (enable)
```

Note When you commit an ACL that has already been attached to an interface, the new values go into effect immediately. The changes to the default ACLs do not need to be committed.

See the "Configuring and Storing VACLs and QoS ACLs in Flash Memory" section on page 15-64 for information about where the QoS ACLs are stored.

Attaching an ACL to an Interface

Note Supervisor Engine 1 with a Layer 2 Switching Engine does not support the commands in this section.

You can do the following:

- For the ingress traffic, attach one ACL of each type (IP, IPX, MAC Layer) to each VLAN.
- For the ingress traffic, attach one ACL of each type (IP, IPX, MAC Layer) to each port that is configured for port-based QoS. You cannot attach the ACLs to a port that is configured for VLAN-based QoS (for more information, see the "Enabling Port-Based or VLAN-Based QoS" section on page 52-40).
- With PFC3, for the egress traffic, attach an IP ACL to each VLAN.

When an ACL of a particular type (IP, IPX, or MAC Layer) is already attached to an interface, attaching a different ACL of the same type detaches the previous ACL.

To attach an ACL to a port or a VLAN, perform this task in privileged mode:

	Task	Command
Step 1	Attach an ACL to an interface.	**set qos acl map** *acl_name* {*mod/port* [**input**] | *vlan* [**input** | **output**]}
Step 2	Verify the configuration.	**show qos acl map** {**config** | **runtime**} {*acl_name* | *mod/port* | *vlan* | **all**}

Note Only PFC3 supports the **input** and **output** keywords.

Note With software release 8.3(1) and later, the ACLs that are attached to a VLAN with the **output** keyword also support the **trust-cos** and **trust-ipprec** keywords.

This example shows how to attach an ACL named test to VLAN 1 to filter the ingress traffic:

```
Console> (enable) set qos acl map test 1
ACL test is successfully mapped to vlan 1 on input side.
Console> (enable)
```

This example shows how to attach an ACL named test2 to VLAN 1 to filter the egress traffic:

```
Console> (enable) set qos acl map test2 1 output
ACL test2 is successfully mapped to vlan 1 on output side.
Console> (enable)
```

Note The default ACLs do not need to be attached to any interfaces.

Detaching an ACL from an Interface

Note Supervisor Engine 1 with a Layer 2 Switching Engine does not support the commands in this section.

To detach an ACL from a port or a VLAN, perform this task in privileged mode:

	Task	Command
Step 1	Detach an ACL from an interface.	**clear qos acl map** *acl_name* {*mod/port* [**input**] | *vlan* [**input** | **output**] | **all**}
Step 2	Verify the configuration.	**show qos acl map** {**config** | **runtime**} {*acl_name* | *mod/port* | *vlan* | **all**}

Note Only PFC3 supports the **input** and **output** keywords.

This example shows how to detach an ACL named my_acl from port 2/1:

```
Console> (enable) clear qos acl map my_acl 2/1
Hardware programming in progress...
ACL my_acl is detached from port 2/1.
Console> (enable)
```

This example shows how to detach an ACL named my_acl from VLAN 4:

```
Console> (enable) clear qos acl map my_acl 4
Hardware programming in progress...
ACL my_acl is detached from vlan 4.
Console> (enable)
```

Note The default ACLs cannot be detached from any interfaces.

Configuring PFC3 Egress DSCP Mutation

These sections describe how to configure PFC3 egress DSCP mutation:

Configuring a DSCP Mutation Map

PFC3 supports 16 DSCP mutation maps. QoS uses one mutation map for the default mapping. You can configure 15 mutation maps. The mutation maps define the internal DSCP-to-egress DSCP relationships.

To configure a DSCP mutation map, perform this task in privileged mode:

	Task	Command
Step 1	Configure a DSCP mutation map.	**set qos dscp-mutation-map** *map_id internal_dscp_list*:*mutated_dscp*...
Step 2	Verify the configuration.	**show qos maps** {**config** | **runtime**} **dscp-mutation-map** *map_id*

This example shows how to configure DSCP mutation map 1:

```
Console> (enable) set qos dscp-mutation-map 1 30:2
QoS dscp-mutation-map with mutation-table-id 1 has been set correctly.
Console> (enable)
```

This example shows how to verify DSCP mutation map 1:

```
Console> (enable) show qos maps config dscp-mutation-map 1
VLAN ID map:
Map ID  VLANS
------  ----------------------------------------
     1  1,78-1005,1025-4094
DSCP mutation map 1:
DSCP                                Policed DSCP
------------------------------------------------
0                                              0
1                                              1
2                                              1
3                                              1
4                                              1
5                                              1
6                                              1
7                                              1
8                                              1
9                                              9
10                                             1
11                                             1
12                                            12
13                                            13
14                                            14
15                                            15
59                                            59
60                                            60
```

```
61                                      61
62                                      62
63                                      63
Console> (enable)
```

Clearing a Configured DSCP Mutation Map

To clear a configured DSCP mutation map, perform this task in privileged mode:

	Task	Command
Step 1	Clear a configured DSCP mutation map.	**clear qos dscp-mutation-map** *vlan_mapped_id* | **all**
Step 2	Verify the configuration.	**show qos maps** {**config** | **runtime**} **dscp-mutation-map** *map_id*

This example shows how to clear DSCP mutation map 3:

```
Console> (enable) clear qos dscp-mutation-map 3
QoS dscp-mutation-map for mutation-table-id 3 is restored to default.
Console> (enable)
```

Applying a DSCP Mutation Map to a VLAN

To apply a DSCP mutation map to a VLAN, perform this task in privileged mode:

	Task	Command
Step 1	Apply a DSCP mutation map to a VLAN.	**set qos dscp-mutation-table-map** *map_id vlan_list*
Step 2	Verify the configuration.	**show qos maps** {**config** | **runtime**} **mutation-table-id** *map_id*

This example shows how to apply DSCP mutation map 1 to VLANs 3 and 20 through 30:

```
Console> (enable) set qos dscp-mutation-table-map 1 3,20-30
VLAN(s) 3,20-30 are mapped to mutation-table-id 1.
Console> (enable)
```

This example shows how to verify the VLAN-to-mutation map mapping:

```
Console> (enable) show qos maps config mutation-table-id 1
VLAN ID map:
Map ID VLANs
------ -------
1      1,20-30
```

Clearing a DSCP Mutation Map from a VLAN

To clear a DSCP mutation map from a VLAN, perform this task in privileged mode:

	Task	Command
Step 1	Clear a DSCP mutation map from a VLAN.	**clear qos dscp-mutation-table-map** {*map_id* | *vlan_id* | **all**}
Step 2	Verify the configuration.	**show qos maps** {**config** | **runtime**} **dscp-mutation-map** *map_id*

This example shows how to clear the association of any VLANs with DSCP mutation map 2:

```
Console> (enable) clear qos dscp-mutation-table-map 2
All VLANS in mutation-table-id 2 are cleared.
```

This example shows how to clear the association of VLANs 3–33 with any DSCP mutation maps:

```
Console> (enable) clear qos dscp-mutation-table-map 3-33
VLAN(s) 3-33 are removed from mutation-table-ids.
```

This example shows how to clear the association of all VLANs with all DSCP mutation maps:

```
Console> (enable) clear qos dscp-mutation-table-map all
All VLANs are removed from mutation-table-ids.
```

Configuring CoS-to-CoS Maps on 802.1Q Tunnel Ports

Ingress Cos-to-CoS mapping is supported on 802.1Q tunnel ports on the WS-X6704-10GE, WS-X6724-SFP, and WS-X6748-GE-TX switching modules. CoS-to-CoS mapping is disabled on the ports that are not configured as 802.1Q tunnel ports.

Defining a CoS-to CoS Map

To define a CoS-to-CoS map, perform this task in privileged mode:

	Task	Command
Step 1	Define the CoS-to-CoS mapping.	**set qos cos-cos-map** *CoS_value*
Step 2	Verify the configuration.	**show qos maps cos-cos-map** [*mod/port*]

This example shows how to define the CoS-to-CoS map:

```
Console> (enable) set qos cos-cos-map 3 2 1 4 5 6 7 4
QoS cos-cos-map set successfully.
Console> (enable)
```

Enabling a CoS-to CoS Map on a Port

To enable a CoS-to-CoS map on a port, perform this task in privileged mode:

	Task	Command
Step 1	Enable the CoS-to-CoS map on an 802.1Q tunnel port.	**set port qos** *mod/port* **trust trust-cos**
Step 2	Verify the port QoS trust configuration.	**show port qos** *mod/port*

```
Console> (enable) set port qos 1/1 trust trust-cos
Port 1/1 qos set to trust-cos
Console> (enable)
```

Note The CoS-to-CoS map is automatically disabled when the port trust is not **trust-cos** for the 802.1Q tunnel ports.

Clearing a CoS-to-CoS Map

To clear a CoS-to-CoS map, perform this task in privileged mode:

	Task	Command
Step 1	Clear the CoS-to-CoS map on an 802.1Q tunnel port.	**clear qos cos-cos-map**
Step 2	Verify the port QoS configuration.	**show qos maps cos-cos-map** [*mod/port*]

This example shows how to clear the CoS-to-CoS mapping:

```
Console> (enable) clear qos cos-cos-map
QoS cos-cos-map setting restored to default.
Console> (enable)
```

Mapping a CoS Value to a Host Destination MAC Address/VLAN Pair

Note QoS supports this command only with a Layer 2 Switching Engine.

To map a CoS value to all frames that are destined for a particular host destination MAC address and VLAN number value pair, perform this task in privileged mode:

	Task	Command
Step 1	Map a CoS value to a host destination MAC address/VLAN pair.	**set qos mac-cos** *dest_mac VLAN cos_value*
Step 2	Verify the configuration.	**show qos mac-cos** {*dest_mac* [*vlan*] | **all**}

This example shows how to map CoS 2 to a destination MAC address and VLAN 525:

```
Console> (enable) set qos mac-cos 00-40-0b-30-03-48 525 2
CoS 2 is assigned to 00-40-0b-30-03-48 vlan 525.
Console> (enable)
```

Deleting a CoS Value to a Host Destination MAC Address/VLAN Pair

Note QoS supports this command only with a Layer 2 Switching Engine.

To delete a host destination MAC address and VLAN number value pair CoS assignment, perform this task in privileged mode:

	Task	Command
Step 1	Delete a host destination MAC address and VLAN number value pair CoS assignment.	**clear qos mac-cos** {*dest_mac* [*vlan*] | **all**}
Step 2	Verify the configuration.	**show qos mac-cos** {*dest_mac* [*vlan*] | **all**}

This example shows how to delete all CoS assignments to the destination MAC addresses and VLANs:

```
Console> (enable) clear qos mac-cos all
All CoS to Mac/Vlan entries are cleared.
Console> (enable)
```

Enabling or Disabling Microflow Policing of Bridged Traffic

Note Supervisor Engine 1 with a Layer 2 Switching Engine does not support the commands in this section.

By default, the microflow policers affect only the Layer 3-switched traffic. To enable or disable microflow policing of the bridged traffic on the switch or on specified VLANs, perform one of these tasks in privileged mode:

Task	Command
Enable microflow policing of the bridged traffic on the switch or on the specified VLANs.	**set qos bridged-microflow-policing** {**enable** | **disable**} *vlan*
Disable microflow policing of the bridged traffic on the switch or on the specified VLANs.	**set qos bridged-microflow-policing** {**enable** | **disable**} *vlan*
Verify the configuration.	**show qos bridged-microflow-policing runtime** {**config** | **runtime**} *vlan*

Note With Layer 3 Switching Engine II, to do any microflow policing, you must enable microflow policing of the bridged traffic.

For more information, see the "Policers" section on page 52-24.

This example shows how to enable microflow policing of the traffic in VLANs 1–20:

```
Console> (enable) set qos bridged-microflow-policing enable 1-20
QoS microflow policing is enabled for bridged packets on vlans 1-20.
Console> (enable)
```

Configuring the Standard Receive-Queue Tail-Drop Thresholds

To configure the standard receive-queue tail-drop thresholds on the switch, perform this task in privileged mode:

Task	Command
Configure the standard receive-queue tail-drop thresholds.	**set qos drop-threshold** *port_type* **rx queue 1** *thr1 thr2 thr3 thr4*

For more information, see the "Receive Queues" section on page 52-13.

QoS maintains separate configurations for the **1q2t**, **1q4t**, and **1p1q4t** ports. This command configures the standard queue. Specify queue 1 (the threshold in the strict-priority queue, when present, is not separately configurable; it uses threshold 4 as specified for queue 1).

The thresholds are all specified as percentages ranging from 1–100. A value of 10 indicates a threshold when the buffer is 10 percent full.

This example shows how to configure the standard receive-queue tail-drop thresholds:

```
Console> (enable) set qos drop-threshold 1q4t rx queue 1 20 40 75 100
Receive drop thresholds for queue 1 set at 20% 40% 75% 100%
Console> (enable)
```

Note You cannot configure a drop threshold in a **1p1q0t** receive queue.

Configuring the 2q2t Port Standard Transmit-Queue Tail-Drop Thresholds

To configure the standard transmit-queue tail-drop thresholds on all **2q2t** ports, perform this task in privileged mode:

Task	Command
Configure the standard transmit-queue tail-drop thresholds on all **2q2t** ports.	**set qos drop-threshold** *port_type* **tx queue** *q# thr1 thr2*

Queue number 1 is the low-priority transmit queue, and queue number 2 is high priority. In each queue, the low-priority threshold number is 1 and the high-priority threshold number is 2.

The thresholds are all specified as percentages ranging from 1–100. A value of 10 indicates a threshold when the buffer is 10 percent full.

This example shows how to configure the low-priority transmit-queue tail-drop thresholds:

```
Console> (enable) set qos drop-threshold 2q2t tx queue 1 40 100
Transmit drop thresholds for queue 1 set at 40%  100%
Console> (enable)
```

Note You cannot configure the tail-drop thresholds in **1p3q1t** transmit queues.

Configuring the Standard Queue WRED-Drop Thresholds

The **1p1q8t** ports have weighted random early detection (WRED)-drop thresholds in their standard receive queue.

The **1p2q2t**, **1p3q1t**, **1p2q1t**, **1p3q8t**, and **1p7q8t** ports have WRED-drop thresholds in their standard transmit queues.

Note The **1p7q8t** (transmit), **1p3q1t** (transmit), **1p2q1t** (transmit), and **1p1q8t** (receive) ports also have nonconfigurable tail-drop thresholds.

To configure the standard queue WRED-drop thresholds on all ports of each type, perform this task in privileged mode:

Task	Command
Configure the standard queue WRED-drop thresholds on all ports of a given type.	**set qos wred 1p1q8t rx queue** *q#* [*thr1Lo*:]*thr1Hi* [*thr2Lo*:]*thr2Hi* [*thr_n_Lo*:]*thr_n_Hi*... [*thr8Lo*:]*thr8Hi* **set qos wred 1p7q8t tx queue** *q#* [*thr1Lo*:]*thr1Hi* [*thr2Lo*:]*thr2Hi* [*thr_n_Lo*:]*thr_n_Hi*... [*thr8Lo*:]*thr8Hi* **set qos wred 1p3q8t tx queue** *q#* [*thr1Lo*:]*thr1Hi* [*thr2Lo*:]*thr2Hi* [*thr_n_Lo*:]*thr_n_Hi*... [*thr8Lo*:]*thr8Hi* **set qos wred 1p2q2t tx queue** *q#* [*thr1Lo*:]*thr1Hi* [*thr2Lo*:]*thr2Hi* **set qos wred 1p3q1t tx queue** *q#* [*thr1Lo*:]*thr1Hi* **set qos wred 1p2q1t tx queue** *q#* [*thr1Lo*:]*thr1Hi*

When configuring the **1p1q8t** ports, note the following:

- Queue 1 is the single standard receive queue.
- When you configure the single standard receive queue, note the following:
  - The first percentage that you enter sets the lowest-priority threshold.
  - The second percentage that you enter sets the next highest-priority threshold.
  - The eighth percentage that you enter sets the highest-priority threshold.

When configuring the **1p7q8t** ports, note the following:

- Queue 1 is the lowest-priority standard transmit queue.
- Queue 7 is the highest-priority standard transmit queue.
- When configuring each standard transmit queue, note the following:
  - The first percentage that you enter sets the lowest-priority threshold.
  - The second percentage that you enter sets the next highest-priority threshold.
  - The eighth percentage that you enter sets the highest-priority threshold.

When configuring the **1p3q8t** ports, note the following:

- Queue 1 is the low-priority standard transmit queue.
- Queue 2 is the medium-priority standard transmit queue.
- Queue 3 is the high-priority standard transmit queue.
- When configuring each standard transmit queue, note the following:
  - The first percentage that you enter sets the lowest-priority threshold.
  - The second percentage that you enter sets the next highest-priority threshold.
  - The eighth percentage that you enter sets the highest-priority threshold.

When configuring the **1p2q2t** ports, note the following:

- Queue 1 is the low-priority standard transmit queue.
- Queue 2 is the high-priority standard transmit queue.
- When configuring each standard transmit queue, note the following:
  - The first percentage that you enter sets the low-priority threshold.
  - The second percentage that you enter sets the high-priority threshold.

When configuring the **1p3q1t** ports, note the following:

- Queue 1 is the low-priority standard transmit queue.
- Queue 2 is the medium-priority standard transmit queue.
- Queue 3 is the high-priority standard transmit queue.
- When you configure each standard transmit queue, the single percentage that you enter sets the threshold.

When configuring the **1p2q1t** ports, note the following:

- Queue 1 is the low-priority standard transmit queue.
- Queue 2 is the high-priority standard transmit queue.
- When you configure each standard transmit queue, the single percentage that you enter sets the threshold.

When configuring thresholds, note the following:

- The thresholds are all specified as percentages ranging from 0–100. A value of 10 indicates a threshold when the buffer is 10 percent full.
- You can configure both the low WRED threshold and the high WRED threshold. You must set the low threshold to a lower percentage than the high threshold.

- The low WRED threshold is the traffic level under which no traffic is dropped. The high WRED threshold is the traffic level above which all traffic is dropped. The traffic in the queue between the low and high WRED thresholds has an increasing chance of being dropped as the queue fills. The default low WRED threshold is zero (all traffic has some chance of being dropped).

This example shows how to configure the low-priority transmit-queue WRED-drop thresholds:

```
Console> (enable) set qos wred 1p2q2t queue 1 40:70 70:100
WRED thresholds for queue 1 set to 40:70 and 70:100 on all WRED-capable 1p2q2t ports.
Console> (enable)
```

Note The threshold in the strict-priority queue is not configurable.

Allocating Bandwidth Between the Standard Transmit Queues

The switch transmits frames from one standard queue at a time using one of these dequeuing algorithms, which use weights to allocate relative bandwidth to each queue as it is serviced in a round-robin fashion:

- Shaped round robin (SRR)—Supported as an option on Supervisor Engine 32 **1p3q8t** ports. If you do not enable SRR, DWRR is used. SRR only allows a queue to use the specific amount of bandwidth that the weight allocates.
- Deficit weighted round robin (DWRR)—Supported on **1p3q1t**, **1p2q1t**, **1p3q8t**, and **1p7q8t** ports. DWRR keeps track of any low-priority queue under-transmission and compensates in the next round.
- Weighted round robin (WRR)—Supported on all other ports. WRR allows a queue to use more than the allocated bandwidth if the other queues are not using any, up to the total bandwidth of the port.

The higher the weight that is assigned to a queue, the more transmit bandwidth is allocated to it. The ratio of the weights divides the total bandwidth of the queue. For example, for three queues on a Gigabit Ethernet port, weights of 25:25:50 provide this division:

- Queue 1—250 Mbps
- Queue 2—250 Mbps
- Queue 3—500 Mbps

Note The actual bandwidth division depends on the granularity that the port hardware applies to the configured weights.

To allocate the bandwidth between the standard transmit queues, perform this task in privileged mode:

Task	Command
Allocate the bandwidth between the standard transmit queues.	**set qos wrr** *port_type queue1-weight queue2-weight* [*queue3-weight*] [**srr**]

The valid values for the *port_type* parameter are **2q2t**, **1p2q2t**, **1p3q1t**, **1p2q1t**, **1p3q8t**, and **1p7q8t**.

QoS maintains separate configurations for each port type. This command configures only the standard queues; the strict-priority queue requires no configuration. The valid values for weight range from 1–255.

This example shows how to allocate the bandwidth for the **2q2t** ports:

```
Console> (enable) set qos wrr 2q2t 30 70
QoS wrr ratio is set successfully.
Console> (enable)
```

Configuring the Receive-Queue Size Ratio

For the **1p1q0t** and **1p1q8t** ports, estimate the mix of standard-priority and strict-priority traffic on your network (for example, 85 percent standard-priority traffic and 15 percent strict-priority traffic). Specify the queue ratios with the estimated percentages, which must range from 1– 99 and together add up to 100.

To configure the receive-queue size ratio, perform this task in privileged mode:

Task	Command
Configure the receive-queue size ratio between receive queue 1 (standard priority) and receive queue 2 (strict priority).	**set qos rxq-ratio** {**1p1q0t** | **1p1q8t**} *queue1-val queue2-val*

This example shows how to configure the receive-queue size ratio:

```
Console> (enable) set qos rxq-ratio 1p1q0t 80 20
QoS rxq-ratio is set successfully.
Console> (enable)
```

Configuring the Transmit-Queue Size Ratio

For the **2q2t**, **1p2q2t**, **1p2q1t**, **1p3q8t**, and **1p7q8t** ports, estimate the mix of the traffic of various priorities on your network (for example, 75 percent low-priority traffic, 15 percent high-priority traffic, and 10 percent strict-priority traffic). Specify the queue ratios with the estimated percentages, which must range from 1– 99 and together add up to 100.

To configure the transmit-queue size ratio for each port type, perform this task in privileged mode:

Task	Command
Configure the transmit-queue size ratio.	**set qos txq-ratio** {**2q2t** | **1p2q2t** | **1p2q1t** | **1p3q8t** | **1p7q8t**} *queue1-val queue2-val* [*queue3-val* [*queue4-val*]]

This example shows how to configure the transmit-queue size ratio:

```
Console> (enable) set qos txq-ratio 2q2t 80 20
QoS txq-ratio is set successfully.
Console> (enable)
```

Mapping the CoS Values to the Drop Thresholds

This command associates the CoS values with the receive- and transmit-queue drop thresholds. QoS maintains separate configurations for each port type.

These sections describe how to map the CoS values to the drop thresholds:

Associating the 1q4t/2q2t Ports

On the **1q4t/2q2t** ports, you configure the receive queues and the transmit queues with the same command.

To associate the CoS values to the drop thresholds on the **1q4t/2q2t** ports, perform this task in privileged mode:

	Task	Command
Step 1	Associate the CoS value to a drop threshold.	**set qos map 2q2t tx** *q# thr#* **cos** *coslist*
Step 2	Verify the configuration.	**show qos info config** {**1q4t rx** | **2q2t tx**}

The receive- and transmit-drop thresholds have this relationship:

- Receive queue 1 (standard) threshold 1 = transmit queue 1 (standard low priority) threshold 1
- Receive queue 1 (standard) threshold 2 = transmit queue 1 (standard low priority) threshold 2
- Receive queue 1 (standard) threshold 3 = transmit queue 2 (standard high priority) threshold 1
- Receive queue 1 (standard) threshold 4 = transmit queue 2 (standard high priority) threshold 2

Use the transmit queue and transmit-queue drop-threshold values in this command. This example shows how to associate the CoS values 0 and 1 to both the standard receive-queue 1/threshold 1 and the standard transmit-queue 1/threshold 1:

```
Console> (enable) set qos map 2q2t tx 1 1 cos 0,1
Qos tx priority queue and threshold mapped to cos successfully.
Console> (enable)
```

Associating the 1q8t, 1q2t/1p2q2t, and 1p1q4t/1p2q2t Ports

On the **1q8t**, **1q2t/1p2q2t** and **1p1q4t/1p2q2t** ports, you configure the receive queues and the transmit queues separately.

1q8t Receive Queues

To associate the CoS values to the **1q8t** receive-queue drop thresholds, perform this task in privileged mode:

	Task	Command
Step 1	Associate the CoS value to a receive-queue drop threshold.	**set qos map 1q8t rx 1** *thr#* **cos** *coslist*
Step 2	Verify the configuration.	**show qos info config 1q2t rx**

Threshold 1 is the lowest-priority threshold. The priority increases with the threshold number.

This example shows how to associate the CoS value 3 to threshold 2:

```
Console> (enable) set qos map 1q8t rx 1 2 cos 3
QoS rx priority queue and threshold mapped to cos successfully.
Console> (enable)
```

1q2t Receive Queues

To associate the CoS values to the **1q2t** receive-queue drop thresholds, perform this task in privileged mode:

	Task	Command
Step 1	Associate the CoS value to a receive-queue drop threshold.	**set qos map 1q2t rx 1 1 cos** *coslist*
Step 2	Verify the configuration.	**show qos info config 1q2t rx**

Threshold 1 is the low-priority threshold. Threshold 2, which is the high-priority threshold, is not configurable.

This example shows how to associate the CoS value 3 to threshold 1:

```
Console> (enable) set qos map 1q2t rx 1 1 cos 3
QoS rx priority queue and threshold mapped to cos successfully.
Console> (enable)
```

1p1q4t Receive Queues

To associate the CoS values to the **1p1q4t** receive-queue drop thresholds, perform this task in privileged mode:

	Task	Command
Step 1	Associate the CoS value to a receive-queue drop threshold.	**set qos map 1p1q4t rx** *q# thr#* **cos** *coslist*
Step 2	Verify the configuration.	**show qos info config 1p1q4t rx**

Queue 1 is the standard queue. Queue 2 is the strict-priority queue.

The threshold numbers range from 1 for low priority to 4 for high priority.

This example shows how to associate the CoS value 5 to the strict-priority receive-queue 2/threshold 1:

```
Console> (enable) set qos map 1p1q4t rx 2 1 cos 5
Qos rx strict queue and threshold mapped to cos successfully.
Console> (enable)
```

1p2q2t Transmit Queues

To associate the CoS values to the **1p2q2t** transmit-queue drop thresholds, perform this task in privileged mode:

	Task	Command
Step 1	Associate the CoS value to a transmit-queue drop threshold.	**set qos map 1p2q2t tx** *q# thr#* **cos** *coslist*
Step 2	Verify the configuration.	**show qos info config 1p2q2t tx**

Queue 1 is standard low priority, queue 2 is high priority, and queue 3 is strict priority.

Threshold 1 is low priority, and threshold 2 is high priority.

This example shows how to associate the CoS value 5 to the strict-priority transmit-queue 3/drop threshold 1:

```
Console> (enable) set qos map 1p2q2t tx 3 1 cos 5
Qos tx strict queue and threshold mapped to cos successfully.
Console> (enable)
```

Associating the 1p1q0t/1p3q1t Ports

On the **1p1q0t/1p3q1t** ports, you configure the receive queues and the transmit queues separately.

1p1q0t Receive Queues

To associate the CoS values to the **1p1q0t** receive queues, perform this task in privileged mode:

	Task	Command
Step 1	Associate the CoS value to a receive queue.	**set qos map 1p1q0t rx** *q#* **cos** *coslist*
Step 2	Verify the configuration.	**show qos info config 1p1q0t rx**

Queue 1 is the standard queue, and queue 2 is the strict-priority queue.

This example shows how to associate the CoS value 5 to the strict-priority receive-queue 2:

```
Console> (enable) set qos map 1p1q0t rx 2 cos 5
QoS queue mapped to cos successfully.
Console> (enable)
```

1p3q1t Transmit Queues

With the **1p3q1t** transmit queues, you can associate a CoS value with either the nonconfigurable tail-drop threshold or the configurable WRED-drop threshold as follows:

- To associate the CoS value with the tail-drop threshold, map the CoS value to the queue.
- To associate the CoS value with the WRED-drop threshold, map the CoS value to the queue and threshold.

To associate the CoS values to the **1p3q1t** transmit-queue drop thresholds, perform this task in privileged mode:

	Task	Command
Step 1	Associate the CoS value to a transmit-queue drop threshold.	**set qos map 1p3q1t tx** *q#* [*thr#*] **cos** *coslist*
Step 2	Verify the configuration.	**show qos info config 1p3q1t tx**

Queue 1 is the standard low priority, queue 2 is medium priority, queue 3 is high priority, and queue 4 is strict priority.

To map the CoS values to the tail-drop threshold, omit the threshold number or enter **0**.

The WRED-drop threshold number is 1.

This example shows how to associate the CoS value 0 to the transmit-queue 1/drop threshold 1:

```
Console> (enable) set qos map 1p3q1t tx 1 1 cos 0
Qos tx strict queue and threshold mapped to cos successfully.
Console> (enable)
```

Associating the 1p1q8t/1p2q1t, 1p3q8t, and 1p7q8t Ports

When you configure the **1p1q8t/1p2q1t** and **1p3q8t** ports, note the following:

- You configure the receive queues and the transmit queues separately.
- You can associate a CoS value with either the nonconfigurable tail-drop threshold or the configurable WRED-drop threshold:
  - To associate a CoS value with the tail-drop threshold, map the CoS value to the queue.
  - To associate a CoS value with the WRED-drop threshold, map the CoS value to the queue and threshold.

1p1q8t Receive Queues

To associate the CoS values to the **1p1q8t** receive queue drop thresholds, perform this task in privileged mode:

	Task	Command
Step 1	Associate the CoS value to a receive queue.	**set qos map 1p1q8t rx** *q#* [*thr#*] **cos** *coslist*
Step 2	Verify the configuration.	**show qos info config 1p1q8t rx**

On the **1p1q8t** ports, queue 1 is the standard receive queue, and queue 2 is the strict-priority receive queue.

To map the CoS values to a tail-drop threshold, omit the threshold number or enter **0**.

This example shows how to associate the CoS value 5 to the strict-priority receive-queue 2:

```
Console> (enable) set qos map 1p1q8t rx 2 cos 5
QoS queue mapped to cos successfully.
Console> (enable)
```

1p2q1t, 1p3q8t, and 1p7q8t Transmit Queues

To associate the CoS values to the **1p2q1t**, **1p3q8t**, or **1p7q8t** transmit-queue drop thresholds, perform this task in privileged mode:

	Task	Command
Step 1	Associate the CoS value to a transmit-queue drop threshold.	**set qos map** [**1p2q1t** | **1p3q8t** | **1p7q8t**] **tx** *q#* [*thr#*] **cos** *coslist*
Step 2	Verify the configuration.	**show qos info config 1p2q1t tx**

On the **1p2q1t** ports, queue 1 is the low-priority standard transmit queue, queue 2 is the high-priority standard transmit queue, and queue 3 is the strict-priority transmit queue.

On the **1p3q8t** ports, queue 1 is the low-priority standard transmit queue, queue 2 is the medium-priority standard transmit queue, queue 3 is the high-priority standard transmit queue, and queue 4 is the strict-priority transmit queue.

On the **1p7q8t** ports, queue 1 is the lowest-priority standard transmit queue, queue 7 is the highest-priority standard transmit queue, and queue 8 is the strict-priority transmit queue.

To map the CoS values to the tail-drop threshold, omit the threshold number or enter **0**.

This example shows how to associate the CoS value 0 to the transmit-queue 1/drop threshold 1:

```
Console> (enable) set qos map 1p2q1t tx 1 1 cos 0
Qos tx strict queue and threshold mapped to cos successfully.
Console> (enable)
```

Reverting to the CoS Map Default

To revert to the default CoS value/drop threshold mapping, perform this task in privileged mode:

	Task	Command
Step 1	Revert to the QoS map defaults.	**clear qos map** {**1p1q4t rx** | **1p1q0t rx** | **1p2q2t tx** | **2q2t tx** | **1p3q1t tx** | **1q8t rx** | **1p3q8t tx** | **1p7q8t tx**}
Step 2	Verify the configuration.	**show qos info config** {**1q4t rx** | **1p1q4t rx** | **1p1q0t rx** | **1p1q8t rx** | **1p2q2t tx** | **2q2t tx** | **1p3q1t tx** | **1p2q1t tx** | **1q8t rx** | **1p3q8t tx** | **1p7q8t tx**}

This example shows how to revert to the QoS map defaults:

```
Console> (enable) clear qos map 1p3q1t tx
Qos map setting cleared.
Console> (enable)
```

Configuring the DSCP Value Maps

Note Supervisor Engine 1 with a Layer 2 Switching Engine does not support the commands in this section.

These sections describe how the DSCP values are mapped to other values:

Mapping the Received CoS Values to the Internal DSCP Values

To map the received CoS values to the internal DSCP value (see the "Internal DSCP Values" section on page 52-16), perform this task in privileged mode:

	Task	Command
Step 1	Map the received CoS values to the internal DSCP values.	**set qos cos-dscp-map** *dscp1 dscp2 dscp3 dscp4 dscp5 dscp6 dscp7 dscp8*
Step 2	Verify the configuration.	**show qos maps** {**config** | **runtime**} [**cos-dscp-map** | **ipprec-dscp-map** | **dscp-cos-map** | **policed-dscp-map**]

Enter 8 DSCP values to which the QoS maps received CoS values 0–7. This example shows how to map the received CoS values to the internal DSCP values:

```
Console> (enable) set qos cos-dscp-map 20 30 1 43 63 12 13 8
QoS cos-dscp-map set successfully.
Console> (enable)
```

To revert to the default CoS to DSCP value mapping, perform this task in privileged mode:

	Task	Command
Step 1	Revert to the CoS value/DSCP value map defaults.	**clear qos cos-dscp-map**
Step 2	Verify the configuration.	**show qos maps** {**config** | **runtime**} [**cos-dscp-map** | **ipprec-dscp-map** | **dscp-cos-map** | **policed-dscp-map**]

This example shows how to revert to the CoS-DSCP map defaults:

```
Console> (enable) clear qos cos-dscp-map
QoS cos-dscp-map setting restored to default.
Console> (enable)
```

Mapping the Received IP Precedence Values to the Internal DSCP Values

To map the received IP precedence values to the internal DSCP value (see the "Internal DSCP Values" section on page 52-16), perform this task in privileged mode:

	Task	Command
Step 1	Map the received IP precedence values to the internal DSCP values.	**set qos ipprec-dscp-map** *dscp1 dscp2 dscp3 dscp4 dscp5 dscp6 dscp7 dscp8*
Step 2	Verify the configuration.	**show qos maps** {**config** | **runtime**} [**cos-dscp-map** | **ipprec-dscp-map** | **dscp-cos-map** | **policed-dscp-map**]

Enter 8 internal DSCP values to which QoS maps received IP precedence values 0–7. This example shows how to map the received IP precedence values to the internal DSCP values:

```
Console> (enable) set qos ipprec-dscp-map 20 30 1 43 63 12 13 8
QoS ipprec-dscp-map set successfully.
Console> (enable)
```

To revert to the default IP precedence-to-DSCP value mapping, perform this task in privileged mode:

	Task	Command
Step 1	Revert to the default IP precedence-to-DSCP value mapping.	**clear qos ipprec-dscp-map**
Step 2	Verify the configuration.	**show qos maps** {**config** | **runtime**} [**cos-dscp-map** | **ipprec-dscp-map** | **dscp-cos-map** | **policed-dscp-map**]

This example shows how to revert to the QoS map defaults:

```
Console> (enable) clear qos ipprec-dscp-map
QoS ipprec-dscp-map setting restored to default.
Console> (enable)
```

Mapping the Internal DSCP Values to the Egress CoS Values

To map the internal DSCP values to the egress CoS values that are used for egress port scheduling and congestion avoidance, perform this task in privileged mode:

	Task	Command
Step 1	Map the internal DSCP values to the egress CoS values.	**set qos dscp-cos-map** *dscp_list:cos ...*
Step 2	Verify the configuration.	**show qos maps** {**config** | **runtime**} [**cos-dscp-map** | **ipprec-dscp-map** | **dscp-cos-map** | **policed-dscp-map**]

For more information, see the "Internal DSCP Values" section on page 52-16 and the "Ethernet Egress Port Scheduling, Congestion Avoidance, and Marking" section on page 52-28.

Enter up to 64 internal DSCP value list/egress CoS value pairs. This example shows how to map the internal DSCP values to the egress CoS values:

```
Console> (enable) set qos dscp-cos-map 20-25:7 33-38:3
QoS dscp-cos-map set successfully.
Console> (enable)
```

To revert to the default CoS-to-DSCP value mapping, perform this task in privileged mode:

	Task	Command
Step 1	Revert to the DSCP value/CoS value map defaults.	**clear qos dscp-cos-map**
Step 2	Verify the configuration.	**show qos maps** {**config** | **runtime**} [**cos-dscp-map** | **ipprec-dscp-map** | **dscp-cos-map** | **policed-dscp-map**]

This example shows how to revert to the CoS-to-DSCP map defaults:

```
Console> (enable) clear qos dscp-cos-map
QoS dscp-cos-map setting restored to default.
Console> (enable)
```

Mapping the DSCP Markdown Values

To map the DSCP markdown values that are used by the policers, perform this task in privileged mode:

	Task	Command
Step 1	Map the DSCP values to mark down the DSCP values.	**set qos policed-dscp-map** *dscp_list:markdown_dscp ...*
Step 2	With PFC2, map the DSCP values to the markdown DSCP values.	**set qos policed-dscp-map** [**normal** | **excess**] *in_profile_dscp_list:policed_dscp ...*
Step 3	Verify the configuration.	**show qos maps** {**config** | **runtime**} [**cos-dscp-map** | **ipprec-dscp-map** | **dscp-cos-map** | **policed-dscp-map**]

For more information, see the "Policers" section on page 52-24.

Enter up to 64 DSCP-value-list/DSCP-value pairs.

This example shows how to map the DSCP markdown values:

```
Console> (enable) set qos policed-dscp-map 20-25:7 33-38:3
QoS dscp-dscp-map set successfully.
Console> (enable)
```

This example shows how to map the DSCP markdown values for the packets exceeding the excess rate:

```
Console> (enable) set qos policed-dscp-map 33:30
QoS normal-rate policed-dscp-map set successfully.
Console> (enable) set qos policed-dscp-map excess-rate 33:30
QoS excess-rate policed-dscp-map set successfully.
Console> (enable)
```

Note Configure the marked-down DSCP values that map to the CoS values that are consistent with the markdown penalty (see the "Mapping the Internal DSCP Values to the Egress CoS Values" section on page 52-74).

To revert to the default DSCP markdown value mapping, perform this task in privileged mode:

	Task	Command
Step 1	Revert to the DSCP markdown map defaults.	**clear qos policed-dscp-map** [**normal-rate** | **excess-rate**]
Step 2	Verify the configuration.	**show qos maps** {**config** | **runtime**} [**cos-dscp-map** | **ipprec-dscp-map** | **dscp-cos-map** | **policed-dscp-map**]

This example shows how to revert to the DSCP markdown map defaults:

```
Console> (enable) clear qos policed-dscp-map
QoS dscp-cos-map setting restored to default.
Console> (enable)
```

Note Without the **normal-rate** or the **excess-rate** keywords, the **clear qos policed-dscp-map** command clears only the normal policed-dscp map.

Displaying QoS Information

To display the QoS information, perform this task:

Task	Command
Display the QoS information.	**show qos info** [**runtime** | **config**]

This example shows how to display the QoS runtime information for port 2/1:

```
Console> show qos info config 2/1
QoS setting in NVRAM:
QoS is enabled
Port 2/1 has 2 transmit queue with 2 drop thresholds (2q2t).
Port 2/1 has 1 receive queue with 4 drop thresholds (1q4t).
Interface type:vlan-based
ACL attached:
The qos trust type is set to untrusted.
Default CoS = 0
Queue and Threshold Mapping:
Queue Threshold CoS
----- --------- ---------------
1     1         0 1
1     2         2 3
2     1         4 5
2     2         6 7
Rx drop thresholds:
Rx drop thresholds are disabled for untrusted ports.
Queue #  Thresholds - percentage (abs values )
```

```
-------  -----------------------------------
1        50% 60% 80% 100%
Tx drop thresholds:
Queue #  Thresholds - percentage (abs values )
-------  -----------------------------------
1        40% 100%
2        40% 100%
Tx WRED thresholds:
WRED feature is not supported for this port_type.
Queue Sizes:
Queue #  Sizes - percentage (abs values )
-------  -----------------------------------
1        80%
2        20%
WRR Configuration of ports with speed 1000Mbps:
Queue #  Ratios (abs values )
-------  -----------------------------------
1        100
2        255
Console> (enable)
```

Displaying the QoS Statistics

To display the QoS statistics, perform this task:

Task	Command
Display the QoS statistics.	**show qos statistics** {*mod*[*/port*] | **l3stats** | **aggregate-policer** [*policer_name*]}

This example shows how to display the QoS statistics for port 2/1:

```
Console> (enable) show qos statistics 5/1
Tx port type of port 5/1 : 2q2t
Q#   Threshold#      Packets            Average Packet      Peak Packet
                     dropped (pkts)     drop rate (pps)     drop rate (pps)
--   ----------      --------------     ---------------     ---------------
1         1           963646               2052                4369
1         2                0                  0                   0
2         1                0                  0                   0
2         2                0                  0                   0

Rx port type of port 5/1 : 1q4t
For untrusted ports all the packets are sent to the same queue,
Rx thresholds are disabled, tail drops are reported instead.
Q#   Threshold#      Packets            Average Packet      Peak Packet
                     dropped (pkts)     drop rate (pps)     drop rate (pps)
--   ----------      --------------     ---------------     ---------------
1    1                           0                  0                   0
1    2                           0                  0                   0
1    3                           0                  0                   0
1    4                           0                  0                   0
```

This example shows how to display the QoS Layer 3 statistics:

```
Console> (enable) show qos statistics l3stats

                                     Total packets Average      Peak Rate
                                                    Rage(pps)   (pps)
                                   ---------------   ----------    --------
Packets dropped due to policing:       621085       1289          3618
IP packets with ToS changed:           4495508      9937         18507
IP packets with CoS changed:           4495508      9937         18507
Non-IP packets with CoS changed:             0         0             0
Console> (enable)
```

This example shows how to display the QoS aggregate policer statistics:

```
Console> (enable) show qos statistics aggregate-policer ag1
QoS aggregate-policer statistics:
Aggregate policer              Allowed packet Packets exceed
                                   count          excess rate
--------------------             -------------- ------------
ag1                                2171253          411450

QoS aggregate-policer average rate statistics over 5 minutes:
Aggregate policer              Allowed packet   Traffic exceeding
                                   count          excess rate
--------------------             -------------- --------------
ag1                                5399          1024

QoS aggregate-policer peak rate statistics over 5 minutes:
(collected every 30 seconds)

Aggregate policer        Peak Allowed
                         rate (pps)
---------------       --------------
ag1                      20802
```

For PFC3B- or PFC3BXL-based switches, the peak allowed packet count rate will not be displayed. The peak byte count rate will be displayed as shown here:

```
Console> (enable) show qos statistics aggregate-policer ag1
QoS aggregate-policer statistics:
Aggregate policer              Allowed byte    Bytes exceed
                                   count         excess rate
--------------------           --------------   ------------
ag1                            9992929900        2819929466

QoS aggregate-policer average rate statistics over 5 minutes:

Aggregate policer               Allowed rate              Traffic exceeding
                                  (kbps)                 excess rate (kbps)
------------------------------ ----------------    ------------------------
ag1                              152080                          42911

QoS aggregate-policer peak rate statistics over 5 minutes:
(collected every 30 seconds)

Aggregate policer        Peak Allowed
                         rate (pps)
---------------       --------------
ag1                         218912
```

Reverting to the QoS Defaults

Note Reverting to the defaults disables QoS, because QoS is disabled by default.

To revert to the QoS defaults, perform this task in privileged mode:

Task	Command
Revert to the QoS defaults.	**clear qos config**

This example shows how to revert to the QoS defaults:

```
Console> (enable) clear qos config
This command will disable QoS and take values back to factory default.
Do you want to continue (y/n) [n]? y
QoS config cleared.
Console> (enable)
```

Disabling QoS

To disable QoS, perform this task in privileged mode:

Task	Command
Disable QoS on the switch.	**set qos** {**enable** | **disable**}

This example shows how to disable QoS:

```
Console> (enable) set qos disable
QoS is disabled.
Console> (enable)
```

Configuring COPS Support

Note

- Supervisor Engine 1 with a Layer 2 Switching Engine does not support the commands in this section.
- COPS can configure QoS only for the IP traffic. Use the CLI or SNMP to configure QoS for all the other traffic.
- Throughout this publication and all Catalyst 6500 series publications, the term *COPS* refers to COPS support as implemented on the Catalyst 6500 series switches.

These sections describe configuring COPS support:

- Port ASICs, page 52-80
- Understanding QoS Policy, page 52-80
- Selecting COPS as the QoS Policy Source, page 52-80

Port ASICs

Some COPS support features affect all ports that are controlled by a port ASIC. The following sections use the term *per-ASIC* to identify the features that configure all ports on the same port ASIC:

- The port ASICs on the Gigabit Ethernet switching modules control up to 4 ports each: 1–4, 5–8, 9–12, and 13–16.
- A port ASIC on the 10-Mbps, 10/100-Mbps, and 100-Mbps Ethernet switching modules controls all ports.
- On the 10-Mbps, 10/100-Mbps, and 100-Mbps Ethernet switching modules, another set of port ASICs control 12 ports each (1–12, 13–24, 25–36, and 37–48), but COPS cannot configure them.
- Changes to an EtherChannel port apply to all ports in the EtherChannel and to all ports that are controlled by the ASIC (or ASICs) that control the EtherChannel ports.

Understanding QoS Policy

The term *QoS policy* refers to the QoS values in effect, such as port trust state and which ACLs are applied to the ports and the VLANs.

Selecting COPS as the QoS Policy Source

QoS uses the locally configured QoS values as the default QoS policy source. To select COPS as the QoS policy source, perform this task in privileged mode:

	Task	Command
Step 1	Select COPS as the QoS policy source.	**set qos policy-source** {**local** | **cops**}
Step 2	Verify the QoS policy source.	**show qos policy-source**

This example shows how to select COPS as the QoS policy source:

```
Console> (enable) set qos policy-source cops
QoS policy source for the switch set to COPS.
Console> (enable) show qos policy-source
QoS policy source for the switch set to COPS.
Console> (enable)
```

Selecting COPS as the QoS policy source switches the following values from the locally configured values to the received COPS values:

- All DSCP maps
- Named and default ACL definitions
- Microflow and aggregate policers
- CoS-to-queue assignments
- Threshold configuration
- WRR weight and buffer configuration
- Default port CoS and ACL-to-interface attachments

Selecting the Locally Configured QoS Policy

To select the locally configured QoS policy, perform this task in privileged mode:

	Task	Command
Step 1	Select the locally configured QoS policy.	**set qos policy-source** {**local** | **cops**}
Step 2	Verify the QoS policy source.	**show qos policy-source**

This example shows how to select the locally configured QoS policy:

```
Console> (enable) set qos policy-source local
QoS policy source for the switch set to local.
Console> (enable) show qos policy-source
QoS policy source for the switch set to local.
Console> (enable)
```

Enabling Use of the Locally Configured QoS Policy

When enabled, COPS is the default QoS policy source for all ports. You can use a locally configured QoS policy on a per-ASIC basis. To enable use of the locally configured QoS policy on a port ASIC, perform this task in privileged mode:

	Task	Command
Step 1	Enable use of the locally configured QoS policy on a port.	**set port qos policy-source** {**local** | **cops**}
Step 2	Verify the QoS policy source for the port.	**show port qos**

This example shows how to enable use of the locally configured QoS policy:

```
Console> (enable) set port qos 1/1 policy-source local
QoS policy source set to local on port(s) 1/1-2.
Console> (enable)
```

Assigning a Port Role

COPS does not configure the ports using the slot number and port number parameters. COPS uses *roles* that you create and assign to the port ASICs.

A role is a name that describes the capability of the ports (for example, *access* or *mod2_1-4*). QoS supports 64 roles per switch. You can assign more than one role to a port ASIC (for example, *mod2ports1-12* and *access*), with the limitation that the combined length of the role names that are assigned to a port ASIC cannot exceed 255 characters.

The role name can be up to 31 characters, is not case sensitive but may include uppercase and lowercase characters, and may consist of a–z, A–Z, 0–9, the dash character (-), the underscore character (_), and the period character (.). The role names cannot start with the underscore character.

The first assignment of a new role to a port creates the role.

To assign the roles to a port ASIC, perform this task in privileged mode:

	Task	Command
Step 1	Assign the roles to a port ASIC.	**set port cops** {*mod/port*} **roles** *role1* [*role2*] ...
Step 2	Verify the roles for the port.	**show port cops** [*mod*[*/port*]]

This example shows how to assign two new roles to the ASIC controlling port 2/1:

```
Console> (enable) set port cops 2/1 roles mod2ports1-12 access
New role 'mod2ports1-12' created.
New role 'access' created.
Roles added for port 2/1-12.
Console> (enable)
```

Removing a Role from the Port ASICs

To remove a role from a port ASIC, perform this task in privileged mode:

	Task	Command
Step 1	Remove a role from a port ASIC.	**clear port cops** {*mod/port*} {**all-roles** | **roles** *role1* [*role2*] ...}
Step 2	Verify the roles for the port.	**show port cops** [*mod*[*/port*]]

This example shows how to remove a role from a port ASIC:

```
Console> (enable) clear port cops 3/1 roles backbone_port main_port
Roles cleared for port(s) 3/1-4.
Console> (enable)
```

Deleting a Role

To delete a role (which removes it from all ports), perform this task in privileged mode:

	Task	Command
Step 1	Delete a role.	**clear cops** {**all-roles** | **roles** *role1* [*role2*] ...}
Step 2	Verify the roles for the port.	**show port cops** [*mod*[*/port*]]

This example shows how to delete a role:

```
Console> (enable) clear cops roles backbone_port main_port
Roles cleared.
Console> (enable)
```

Configuring the Policy Decision Point Servers

Note COPS and RSVP can use the same policy decision point (PDP) server.

COPS obtains the QoS policy from a PDP server. Configure a primary PDP server and optionally, a backup PDP server.

To configure a PDP server, perform this task in privileged mode:

	Task	Command
Step 1	Configure a PDP server.	**set cops server** *ip_address* [*port*] [**primary**] [**diff-serv** | **rsvp**]
Step 2	Verify the PDP server configuration.	**show cops info**

The *ip_address* parameter can be the IP address or the name of the server.

The *port* variable is the PDP server TCP port number.

Use the **diff-serv** keyword to set the address only for COPS.

This example shows how to configure a PDP server:

```
Console> (enable) set cops server my_server1 primary
my_server1 added to the COPS diff-serv server table as primary server.
my_server1 added to the COPS rsvp server table as primary server.
Console> (enable)
```

Deleting the PDP Server Configuration

To delete the PDP server configuration, perform this task in privileged mode:

	Task	Command
Step 1	Delete the PDP server configuration.	**clear cops server** {**all** | *ip_address* [**diff-serv** | **rsvp**]}
Step 2	Verify the PDP server configuration.	**show cops info**

This example shows how to delete the PDP server configuration:

```
Console> (enable) clear cops server all
All COPS diff-serv servers cleared.
All COPS rsvp servers cleared.
Console> (enable)
```

Configuring the COPS Domain Name

The PDP servers use a COPS domain name to communicate with the policy enforcement point (PEP) devices such as the switches. To configure a COPS domain name for the switch, perform this task in privileged mode:

	Task	Command
Step 1	Configure the COPS domain name.	**set cops domain-name** *domain_name*
Step 2	Verify the COPS domain name.	**show cops info**

This example shows how to configure a COPS domain name:

```
Console> (enable) set cops domain-name my_domain
Domain name set to my_domain.
Console> (enable)
```

Deleting the COPS Domain Name

To delete the COPS domain name, perform this task in privileged mode:

	Task	Command
Step 1	Delete the COPS domain name.	**clear cops domain-name**
Step 2	Verify the configuration.	**show cops info**

This example shows how to delete the COPS domain name:

```
Console> (enable) clear cops domain-name
Domain name cleared.
Console> (enable)
```

Configuring the COPS Communications Parameters

To configure the parameters that COPS uses to communicate with the PDP server, perform this task in privileged mode:

	Task	Command
Step 1	Configure the parameters that COPS uses to communicate with the PDP server.	**set cops retry-interval** *initial increment maximum*
Step 2	Verify the configuration.	**show cops info**

Enter the parameters as a number of seconds in the range from 0–65535. The value of the *initial* parameter plus the value of the *increment* parameter must not exceed the value of the *maximum* parameter.

This example shows how to configure the parameters that COPS uses to communicate with the PDP server:

```
Console> (enable) set cops retry-interval 15 1 30
Connection retry intervals set.
Console> (enable)
```

Configuring RSVP Support

Note

- Supervisor Engine 1 with a Layer 2 Switching Engine does not support the commands in this section.
- Throughout this publication and all Catalyst 6500 series switch publications, the term *RSVP* refers to the RSVP null service template and receiver proxy functionality support as implemented on the Catalyst 6500 series switches.

These sections describe how to configure the RSVP null service template and receiver proxy functionality support:

Enabling RSVP Support

To enable RSVP support, perform this task in privileged mode:

	Task	Command
Step 1	Enable RSVP support on the switch.	**set qos rsvp** {**enable** | **disable**}
Step 2	Verify the configuration.	**show qos rsvp info**
Step 3	Display RSVP activity.	**show qos rsvp flow-info**

This example shows how to enable RSVP support:

```
Console> (enable) set qos rsvp enable
RSVP enabled on the switch.
Console> (enable)
```

Disabling RSVP Support

To disable RSVP support, perform this task in privileged mode:

	Task	Command
Step 1	Disable RSVP support on the switch.	**set qos rsvp** {**enable** | **disable**}
Step 2	Verify the configuration.	**show qos rsvp info**

This example shows how to disable RSVP support:

```
Console> (enable) set qos rsvp disable
RSVP disabled on the switch.
Console> (enable)
```

Enabling the Participation in the DSBM Election

Catalyst 6500 series switches can serve as the Designated Subnet Bandwidth Manager (DSBM). You can enable the participation in the election of the DSBM on a per-port basis.

Note The DSBM is not reelected when additional RSVP devices join the network. To control which device is the DSBM, disable election participation in all devices except the one that you want elected as the DSBM. After the DSBM is elected, reenable election participation in other devices, as appropriate for the network configuration.

To enable the participation of a port in the election of the DSBM, perform this task in privileged mode:

	Task	Command
Step 1	Enable the participation of a port in the election of the DSBM.	**set port rsvp** {*mod/port*} **dsbm-election** {**disable** | **enable** *priority*}
Step 2	Verify the configuration of the port.	**show port rsvp** [*mod* | *mod/port*]

The range for the *priority* parameter is from 128–255.

This example shows how to enable the participation of ports 2/1 and 3/2 in the election of the DSBM:

```
Console> (enable) set port rsvp 2/1,3/2 dsbm-election enable 232
DSBM enabled and priority set to 232 for ports 2/1,3/2.
Console> (enable)
```

Disabling the Participation in the DSBM Election

To disable the participation of a port in the election of the DSBM, perform this task in privileged mode:

	Task	Command
Step 1	Disable the participation of a port in the election of the DSBM.	**set port rsvp** {*mod/port*} **dsbm-election** {**disable** | **enable** *priority*}
Step 2	Verify the configuration.	**show port rsvp show port rsvp** [*mod*[*/port*]]

This example shows how to disable the participation of port 2/1 in the election of the DSBM:

```
Console> (enable) set port rsvp 2/1 dsbm-election disable
DSBM disabled for port 2/1.
Console> (enable)
```

Configuring the Policy Decision Point Servers

Note COPS and RSVP can use the same PDP server.

When the switch is the DSBM, RSVP communicates with a PDP server. Configure a primary PDP server and optionally, a backup PDP server.

To configure a PDP server, perform this task in privileged mode:

	Task	Command
Step 1	Configure a PDP server.	**set cops server** *ip_address* [*port*] [**primary**] [**diff-serv** | **rsvp**]
Step 2	Verify the PDP server configuration.	**show cops info**

The *ip_address* parameter can be the IP address or the name of the server.

The *port* variable is the PDP server TCP port number.

Use the **rsvp** keyword to set the address only for RSVP.

This example shows how to configure a PDP server:

```
Console> (enable) set cops server my_server1 primary rsvp
my_server1 added to the COPS rsvp server table as primary server.
Console> (enable)
```

Deleting the PDP Server Configuration

To delete the PDP server configuration, perform this task in privileged mode:

	Task	Command
Step 1	Delete the PDP server configuration.	**clear cops server** {**all** | *ip_address* [**diff-serv** | **rsvp**]}
Step 2	Verify the PDP server configuration.	**show cops info**

Use the **rsvp** keyword to delete only the RSVP address.

This example shows how to delete the PDP server configuration:

```
Console> (enable) clear cops server all
All COPS diff-serv servers cleared.
All COPS rsvp servers cleared.
Console> (enable)
```

Configuring the RSVP Policy Timeout

When the switch is the DSBM and communication with the PDP server is lost, the switch continues to function as the DSBM, using the cached values, for the period that is specified by the timeout value; the behavior for the new or modified RSVP **path** messages is determined by the RSVP local policy setting.

If communication with the PDP server is not reestablished before the timeout period expires, the switch reverts to the role of the Subnet Bandwidth Manager (SBM) client for all ports and forwards the RSVP messages to a newly elected DSBM on the segment. When there is no communication with the PDP server, the switch does not participate in election of the DSBM.

To configure the time that the switch continues to be the DSBM after communication with the PDP server is lost, perform this task in privileged mode:

	Task	Command
Step 1	Configure the RSVP policy timeout.	**set qos rsvp policy-timeout** *timeout*
Step 2	Verify the configuration.	**show qos rsvp info**

Enter the *timeout* parameter as a number of minutes in the range from 0–65535 (the default is 30).

This example shows how to configure the RSVP policy timeout:

```
Console> (enable) set qos rsvp policy-timeout 45
RSVP database policy timeout set to 45 minutes.
Console> (enable)
```

Configuring the RSVP Use of Local Policy

To configure how RSVP operates after communication with the PDP is lost, perform this task in privileged mode:

	Task	Command
Step 1	Configure how RSVP operates when there is no communication with the PDP server.	**set qos rsvp local-policy** {**forward** | **reject**}
Step 2	Verify the configuration.	**show qos rsvp info**

The **forward** keyword sets the local policy to forward all new or modified RSVP **path** messages. The **reject** keyword sets the local policy to reject all new or modified RSVP **path** messages. This example shows how to change the default local RSVP policy setting to reject all the new or modified RSVP **path** messages:

```
Console> (enable) set qos rsvp local-policy reject
RSVP local policy set to reject.
Console> (enable)
```

Note The RSVP local policy is used only until the RSVP policy timeout expires after the connection to the PDP is lost. After the RSVP policy timeout expires, the switch behaves as an SBM client. The RSVP messages pass through the switch unchanged regardless of the RSVP local policy setting. The RSVP local policy setting is not used if the switch never establishes a connection to the PDP.

Configuring QoS Statistics Data Export

These sections describe how to configure the QoS statistics data export feature:

Enabling QoS Statistics Data Export Globally

To export the QoS statistics data for the ports and the aggregate policers, you must first configure the feature globally.

To enable QoS statistics data export globally, perform this task in privileged mode:

	Task	Command
Step 1	Enable QoS statistics data export.	**set qos statistics export enable | disable**
Step 2	Verify the configuration.	**show qos statistics export info**

This example shows how to enable QoS statistics data export globally and verify the configuration:

```
Console> (enable) set qos statistics export enable
Export is enabled.
Export destination:172.20.52.3 SYSLOG facility LOG_LOCAL6 (176), severity LOG_DE
BUG (7)
Aggregate policer export is not supported
Console> (enable) show qos statistics export info
Statistics export status and configuration information
-----------------------------------------------------
Export status: enabled
Export time interval: 300
Export destination:172.20.52.3 SYSLOG facility LOG_LOCAL6 (176), severity LOG_DE
BUG (7)
Port    Export
------  --------
 1/1    disabled
 1/2    disabled
 3/1    disabled
 3/2    disabled
 5/1    disabled
 5/2    disabled
 5/3    disabled
 5/4    disabled
<...output truncated...>
Console> (enable)
```

Enabling Per-Port QoS Statistics Data Export

To enable QoS statistics data export on a per-port basis, perform this task in privileged mode:

	Task	Command
Step 1	Enable QoS statistics data export per port.	**set qos statistics export port** *mod/port* **enable** | **disable**
Step 2	Verify the configuration.	**show qos statistics export info**

Note You must enable QoS statistics data export globally in order for the per-port configuration to take effect.

This example shows how to enable the QoS statistics data export per port and verify the configuration:

```
Console> (enable) set qos statistics export port 5/1 enable
Port export enabled on 5/1.
Console> (enable) show qos statistics export info
Statistics export status and configuration information
-----------------------------------------------------
Export status: enabled
Export time interval: 300
Export destination:172.20.52.3 SYSLOG facility LOG_LOCAL6 (176), severity LOG_DE
BUG (7)
Port    Export
------  --------
 1/1    disabled
 1/2    disabled
 3/1    disabled
 3/2    disabled
 5/1    enabled
 5/2    disabled
<output truncated>
Console> (enable)
```

When enabled on a port, QoS statistics data export contains the following fields, separated by the delimiter character:

- Export type (“1” for a port)
- Slot/port
- Number of ingress packets
- Number of ingress bytes
- Number of egress packets
- Number of egress bytes
- Time stamp

Enabling Per-Aggregate Policer QoS Statistics Data Export

To enable QoS statistics data export on a per-aggregate policer basis, perform this task in privileged mode:

	Task	Command
Step 1	Enable QoS statistics data export per-aggregate policer.	**set qos statistics export aggregate** *name* {**enable** | **disable**}
Step 2	Verify the configuration.	**show qos statistics export info**

Note You must enable QoS statistics data export globally in order for the per-aggregate policer configuration to take effect.

This example shows how to enable QoS statistics data export for a specific aggregate policer and verify the configuration:

```
Console> (enable) set qos statistics export aggregate ipagg_3 enable
Statistics data export enabled for aggregate policer ipagg_3
Console> (enable) show qos statistics export info
Statistics export status and configuration information
-----------------------------------------------------
Export status: enabled
Export time interval: 300
Export destination:172.20.52.3 SYSLOG facility LOG_LOCAL6 (176), severity LOG_DE
BUG (7)
Port    Export
------  --------
 1/1    disabled
 1/2    disabled
 3/1    disabled
 3/2    disabled
 5/1    enabled
 5/2    disabled
<output truncated>

Aggregate name  Export
--------------  --------
ipagg_3         enabled
Console> (enable)
```

When enabled for a named aggregate policer, QoS statistics data export contains the following fields, separated by the delimiter character:

- Export type (“2” for an aggregate policer)
- Aggregate policer name
- Direction (“in”)
- Number of in-profile packets
- Number of packets that exceed the CIR
- Number of packets that exceed the PIR
- Time stamp

Clearing the Aggregate Policer QoS Statistics

To clear the aggregate policer QoS statistics, perform this task in privileged mode:

	Task	Command
Step 1	Clear the aggregate policer QoS statistics.	**clear qos statistics aggregate-policer** [*policer_name*]
Step 2	Verify the configuration.	**show qos statistics export info**

This example shows how to clear the QoS aggregate policer statistics for a specific aggregate policer:

```
Console> (enable) clear qos statistics aggregate-policer aggr_1
Aggregate policer 'aggr_1' statistical counters cleared.
```

If you do not specify the aggregate policer, all aggregate policer statistics are cleared:

```
Console> (enable) clear qos statistics aggregate-policer
QoS aggregate policers statistical counters cleared.
```

Setting the QoS Statistics Data Export Time Interval

The default interval at which the QoS statistics is exported is 30 seconds. To set the time interval for the QoS statistics data export, perform this task in privileged mode:

	Task	Command
Step 1	Set the time interval for the QoS statistics data export.	**set qos statistics export interval** *interval_seconds*
Step 2	Verify the configuration.	**show qos statistics export info**

This example shows how to set the QoS statistics data export interval and verify the configuration:

```
Console> (enable) set qos statistics export interval 500
Time interval set to 500
Console> (enable) show qos statistics export info
Statistics export status and configuration information
-----------------------------------------------------
Export status: enabled
Export time interval: 500
Export destination:172.20.52.3 SYSLOG facility LOG_LOCAL6 (176), severity LOG_DE
BUG (7)
Port    Export
------  --------
 1/1    disabled
 1/2    disabled
 3/1    disabled
 3/2    disabled
 5/1    enabled
 5/2    disabled
<output truncated>

Aggregate name  Export
--------------  --------
ipagg_3         enabled
Console> (enable)
```

Configuring the QoS Statistics Data Export Destination Host and UDP Port

To configure the QoS statistics data export destination host and UDP port number, perform this task in privileged mode:

	Task	Command
Step 1	Configure the QoS statistics data export destination host and UDP port number.	**set qos statistics export destination** {*host_name* | *ip_address*} [**syslog** [*facility* | *severity*] *port*]
Step 2	Verify the configuration.	**show qos statistics export info**

This example shows how to configure the QoS statistics data export destination host and UDP port number and verify the configuration:

```
Console> (enable) set qos statistics export destination stargate 9996
Statistics data export destination set to stargate port 9996.
Console> (enable) show qos statistics export info
Statistics export status and configuration information
-----------------------------------------------------
Export status: enabled
Export time interval: 500
Export destination:Stargate, UDP port 9996
Port    Export
------  --------
 1/1    disabled
 1/2    disabled
 3/1    disabled
 3/2    disabled
 5/1    enabled
 5/2    disabled
<output truncated>

Aggregate name  Export
--------------  --------
ipagg_3         enabled
Console> (enable)
```

Displaying the QoS Statistics

To display the QoS statistics per-aggregate policer packet and byte rates, perform this task in privileged mode:

Task	Command
Display the QoS statistics per-aggregate policer packet and byte rates.	**show qos statistics aggregate-policer** [*policer_name*]

This example shows how to display the QoS statistics per-aggregate policer packet and byte rates:

```
Console> show qos statistics aggregate-policer
QoS aggregate-policer statistics:
Aggregate Policer                Packet Count Packets exceed Packets exceed
                                              normal rate excess rate
-------------------------------- ------------ -------------- -----------------
test                                     1000             20                 5
Console>
```

CHAPTER 53

Using Automatic QoS

This chapter describes how to use the automatic quality of service (QoS) configuration features on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

Note Automatic QoS is not supported on Supervisor Engine 720 in software release 8.1(1).

Note For information on using automatic voice configuration, see the "Using SmartPorts" section on page 55-38.

This chapter consists of these sections:

Understanding How Automatic QoS Works

Automatic QoS consists of a macro that simplifies the QoS configuration on the Catalyst 6500 series switches. The automatic QoS macro covers all the QoS configuration tasks that are required for implementing the recommended Architecture for Voice, Video, and Integrated Data (AVVID) settings for a voice port.

Automatic QoS focuses on the voice networks that are built using the Cisco IP Phone 79xx series and the Cisco SoftPhone. However, other phones can equally benefit from the automatically configured QoS settings. With automatic QoS, you use keywords, such as **ciscoipphone** or **ciscosoftphone**, or other AVVID types to allow you to specify the type of QoS parameters that you desire on a particular port. With automatic QoS, all appropriate QoS settings (Internet Engineering Task Force (IETF)-recommended values and proven AVVID settings) are applied to the port.

QoS Overview

These sections provide an overview of QoS:

Typical CoS and DSCP Values for Voice and Video Networks

The IETF recommends that you use several values for the different traffic types that are found in voice and video networks. Automatic QoS uses these values to configure such QoS parameters as CoS-to-queue maps, differentiated services code point (DSCP)-to-CoS maps, and so on.

Catalyst 6500 series switches use the differentiated services (DIFFSERV) model for QoS. This model outlines three traffic types:

- EF (Expedited Forwarding)
- AF (Assured Forwarding)
- BE (Best Effort)

Four traffic classes exist within the AF class. The classes are denoted by AF*XY* where *X* is the class number and *Y* is the drop precedence number. *X* corresponds to a queue, and *Y* corresponds to a drop precedence value within the queue (either WRED or tail drop). EF has the highest priority, BE has the lowest priority, and the priority for AF is somewhere in between.

See Table 53-1 for the recommended CoS and DSCP values for the voice networks and other traffic types. The values listed are assumed when configuring the CoS-to-queue maps and other CoS/DSCP value-dependent configurations with the automatic QoS macro.

Table 53-1 Typical CoS and DSCP values in Cisco Voice and Video Networks

CoS Value[1]	DSCP	Significance
0	0	Default traffic (BE class)
3	26 (IETF recommended)	Voice/video call control/signaling (TCP) AF31 class
5	46 (IETF recommended)	Voice-bearer stream (RTP/UDP) EF class
4	34 (IETF recommended)	Video-bearer stream AF41 class
2	18	Mission critical/transactional traffic AF21 class
1	10	Streaming video (not interactive) AF11
6	48	Routing protocols (as default)
7		Spanning Tree Protocol

1. Some values differ from the current QoS default values for Catalyst software (such as CoS-to-DSCP maps).

The priorities for these CoS/DSCP values are as follows:

- CoS 5 (voice data)—Highest priority (priority queue if present, otherwise high queue)
- CoS 6, 7 (routing protocols)—Second priority (high queue)
- CoS 3, 4 (call signal and video stream)—Third priority (high queue)
- CoS 1, 2 (streaming and mission critical)—Fourth priority (high queue)
- CoS 0— Low priority (low queue)

For the ports that do not implement a priority queue, the WRED and tail-drop mechanisms are used to attain traffic prioritization within the queue. See the "Global Automatic QoS Detail Settings" section on page 53-7 for specific scheduling settings.

QoS Scenario—Cisco IP Phone

In most configurations, you can connect the Cisco IP Phone 79xx directly to the Catalyst switch port. Optionally, you can attach a PC to the phone and use the phone as a hop to the switch.

Typically, the traffic that comes from the phone and enters the switch is marked with a tag using the 802.1Q/p header. The header contains the VLAN information and the CoS 3-bit field. The CoS determines the priority of the packet. The switch uses the CoS field to distinguish the PC traffic from the phone traffic. The switch can also use the DSCP field for the same purpose.

In most Cisco IP Phone 79xx configurations, the traffic that comes from the phone and enters the switch is trusted. You set the port trust to trust-cos to prioritize the voice traffic over other types of traffic in the network.

The Cisco IP Phone 79xx has a built-in switch that mixes the traffic that comes from the PC, the phone, and the switch port. The Cisco IP Phone 79xx has the trust and classification capabilities that you need to configure. For more information, see the "Port-Specific Automatic QoS Settings—ciscosoftphone" section on page 53-10.

QoS Scenario—Cisco SoftPhone

The Cisco SoftPhone is a software product that runs on a standard PC and emulates an IP phone. The main difference between the Cisco SoftPhone and the Cisco IP Phone 79xx is that the Cisco SoftPhone marks its voice traffic through a DSCP, while the Cisco IP Phone 79xx marks its traffic through a CoS. The QoS settings on the switch accommodate this behavior by trusting the Layer 3 marking of the traffic entering the port. All other behavior is similar to the Cisco IP Phone 79xx.

Using the Automatic QoS Macro on the Switch

These sections describe the automatic QoS macro:

Automatic QoS Overview

The automatic QoS macro is divided into these two separate components:

- Global automatic QoS command (**set qos auto**)—Deals with all switch-wide related QoS settings that are not specific to any given interface. These settings include CoS-to-queue maps, CoS-to-DSCP maps, and WRED settings for specific port types and global mappings.
- Port-specific automatic QoS command (**set port qos** *mod/port* **autoqos**)—Configures all inbound QoS parameters for a particular port to reflect the desired traffic type (voice, video, and applications).

Tip To ensure that automatic QoS works properly, you should execute both components.

Automatic QoS Configuration Guidelines and Restrictions

These sections provide the configuration guidelines and restrictions for automatic QoS:

Configuration Files

Creating the commands (macros) that implement other commands can lead to conflicting commands. For example, if you configure a CoS-to-queue map with a certain setting and then enable the automatic QoS macro feature, the macro that you enabled will alter the CoS-to-queue map.

To avoid conflicting commands, the configuration file includes all the legacy commands that are included in the macro. The actual macro command does not appear in the configuration file; instead, all the existing configuration commands that result from executing the macro are included in the configuration file. For example, when you enter the **set qos autoqos** command and then enter the **write config** command, all existing QoS-related CLI commands display, excluding the actual macro command itself.

Supported Phones

When you use automatic QoS with the **ciscoipphone** keyword, some of the QoS configuration requires phone-specific configuration (trust-ext, ext-cos) which is supported only on the following phones: Cisco IP Phone 7910, Cisco IP Phone 7940, Cisco IP Phone 7960, and Cisco IP Phone 7935. However, the **ciscoipphone** keyword is not exclusive to these models only; any phone can benefit from all the other QoS settings that are configured on the switch.

Cisco SoftPhone is supported through the **ciscoipsoftphone** keyword.

CDP Dependencies

To configure the QoS settings and trusted boundary on the Cisco IP Phone, you must enable Cisco Discovery Protocol (CDP) version 2 or later on the port. If you enable trusted boundary, a syslog warning message displays if CDP is not enabled or if CDP is running version 1.

You need to enable CDP only for the **ciscoipphone** QoS configuration; CDP does not affect the other components of the automatic QoS features. When you use the **ciscoipphone** keyword with the port-specific automatic QoS feature, a warning displays if the port does not have CDP enabled. See the "CDP Warning" section on page 53-24.

COPS Considerations

You can configure a port for the local policy or the Common Open Policy Service (COPS) policy. This setting specifies whether the port should get its QoS configuration information from a local configuration or through a COPS server. If you enable COPS on the port as well as globally enable COPS, the policy that is specified by the COPS server applies. If you disable COPS and/or the configured policy is local, the local configuration QoS policy applies.

Automatic QoS affects only the local policy on a port. If you execute automatic QoS on a port that has a configured policy that is currently set to COPS, the policy reverts to the local policy. The global QoS policy reverts to the local policy (through the global automatic QoS command), and the port-based policy reverts to the local policy (through the port-based automatic QoS command). A warning displays if the policy of a port or global policy has been changed from COPS to local. For more information, see the "COPS Warning Message" section on page 53-24. Any existing COPS roles that are already associated with the port are not changed.

RSVP Considerations

All global and port-based Resource Reservation Protocol (RSVP)-related settings (including the RSVP [Designated Subnet Bandwidth Manager] DBSM election settings) are not changed by the automatic QoS macros.

Current QoS Default Settings

All current QoS settings are applied as described in the "Detailed Automatic QoS Configuration Statements" section on page 53-18. Some of these QoS settings reflect the current QoS defaults. After automatic QoS has been applied, *all* QoS settings, regardless of whether or not they were defaults, are applied on the port/switch.

EtherChannel Considerations

The global automatic QoS command supports channeling. All outbound QoS is configured for all channeling or nonchanneling interfaces. Channeling is not supported with the per-port automatic QoS commands.

Video Traffic Considerations

The CoS and DSCP values that are associated with the video traffic are prioritized for the global QoS settings. For more information, see the "Typical CoS and DSCP Values for Voice and Video Networks" section on page 53-2.

Clearing the QoS Configuration

Clearing the QoS configuration resets the configuration to the default QoS values. The automatic QoS features do not alter the default values.

PFC/PFC2 Support

No PFC or PFC2 is required for the **ciscoipphone** and **trust cos** keywords. A PFC or PFC2 is required for the **ciscosoftphone** and **trust dscp** keywords.

1p1q0t/1p3q1t Port Support

All 1p1q0t/1p3q1t ports must either be in port-based mode or VLAN-based mode. If a change is required (for example, if the port was configured for VLAN-based mode before you executed automatic QoS), a syslog message displays. The message indicates that a change to an interface type was needed that affected all ports in the module. For more information, see the "Interface Change for All Ports Required—Warning Level" section on page 53-26.

Global Automatic QoS Macro

These sections describe the global automatic QoS macro:

Overview

You must configure both egress and ingress QoS for QoS to function properly. Because any traffic type can egress on any given port, the egress QoS settings must have global QoS settings. The settings take into account *all* the possible traffic types that are listed in the "Typical CoS and DSCP Values for Voice and Video Networks" section on page 53-2. The egress QoS settings are applied to all the ports in the switch. The global QoS settings cover the *ingress* scheduling settings, because the granularity CoS-to-queue mapping is *port-type* specific and not port specific. The port-specific QoS settings, such as QoS ACLs, port trust, and default CoS, are not altered.

Global Automatic QoS Detail Settings

Table 53-2 through Table 53-6 list the values of all the QoS parameters that are configured through the global automatic QoS command.

Note The 1p1q8t default WRED settings are not changed from the current QoS defaults; only the CoS-to-threshold map is changed.

Table 53-2 Switch-Wide Settings (Global QoS Settings)

QoS Parameter	Setting
CoS-to-DSCP map	0 **10 18 26 34 46** 48 56 (bold indicates nondefault values)
IP-precedence-to-DSCP map	0 **10 18 26 34 46** 48 56 (bold indicates nondefault values)
DSCP-to-CoS map	{0-7}, {8-15}, {16-23}, {24-31}, {32-39}, {40-47}, {48-55}, {56-63} (as per default)
Policed-DSCP map	As per default with 46:0 and 26:0 (see the "Global Automatic QoS Macro" section on page 53-6)
Policed-DSCP map excess rate	As per default (see the "Global Automatic QoS Macro" section on page 53-6)
Default QoS IP ACL	ip dscp 0 (as per default)

Table 53-3 Scheduling Specific Settings (Global QoS Settings)

Field	Value
1p1q0t rxq-ratio	80% : 20% (q1 : p1)
1p3q1t wrr	20 100 200 (q1 q2 q3)
2q2t txq-ratio	80% : 20% (q1 : q2)
2q2t wrr	100 255 (q1 q2)

Table 53-4 CoS-to-Queue Maps and Tail/WRED Settings (Global QoS Settings)

	2q2t	Tail (2q2t)	1q2t	Tail (1q2t)	1q4t	Tail (1q4t)	1p3q1t	WRED (1p3q1t)	1p1q0t
Q1t1	0	(100%)	0, 1, 2, 3, 4	(80%)	0	(50%)	0	(70% : 100%)	0, 1, 2, 3, 4
Q1t2		(100%)	5, 6, 7	(100%)		(60%)			
Q1t3					1, 2, 3, 4	(80%)			
Q1t4					5, 6, 7	(100%)			
Q2t1	1, 2, 3, 4	(80%)					1, 2	(70% : 100%)	5, 6, 7
Q2t2	5, 6, 7	(100%)							
Q3t1							3, 4	(70% : 90%)	
Q3							6, 7	WRED disabled	
Q4t1							5		

Table 53-5 Scheduling Specific Settings (Global QoS Settings)

Field	Value
1p2q2t txq-ratio	70% : 15% : 15% (q1 q2 1p)
1p2q2t wrr	50 255 (q1 q2)
1p1q8t rxq-ratio	80 20 (q1 1p)
1p2q1t txq-ratio	70% : 15% : 15% (q1 q2 1p)
1p2q1t wrr	50 255 (q1 q2)

Table 53-6 CoS-to-Queue Maps and Tail/WRED Settings (Global QoS Settings)

	1p2q2t	WRED	1p1q4t	Tail	1p2q1t	WRED	1p1q8t	WRED
Q1t1	0	(70% : 100%)	0	(50%)	0	(70% : 100%)	0	(40% : 70%)
Q1t2		(70% : 100%)		(60%)			1, 2	(60% : 90%) (*threshold 5*)
Q1t3			1,2,3,4	(80%)			3, 4	(70% : 100%) (*threshold 8*)
Q1t4			6,7	(100%)				
Q2t1	1, 2, 3, 4	(70% : 90%)	5		1, 2, 3, 4	(70% : 90%)	5, 6, 7	
Q2t2	6, 7	(100% : 100%)						
Q2					6, 7	WRED disabled		
Q3t1	5				5			

Port-Specific Automatic QoS Macro

The port-specific automatic QoS macro handles all inbound QoS configuration that is specific to a particular traffic type. The support is implemented for **ciscoipphone**, **ciscosoftphone**, and **trust**. See the "CLI Interface for Automatic QoS" section on page 53-13 for the associated CLI commands.

The QoS ingress port-specific settings include port trust, default CoS, classification, and policing but do not include scheduling. The input scheduling is programmed through the global automatic QoS macro. Together with the global automatic QoS macro command, all QoS settings are configured properly for a specific QoS traffic type.

The existing QoS ACLs that are already associated with a port are removed when the ACL mappings change. The ACL names and instances are not changed.

These sections describe the port-specific automatic QoS macro:

Port-Specific Automatic QoS Settings—ciscoipphone

Use the **ciscoipphone** keyword to set the port to trust-cos and to enable trusted boundary. Combined with the global automatic QoS command, all settings are configured on the switch to properly handle the signaling, voice bearer, and PC data entering and leaving the port.

In addition to the switch-side QoS settings that are covered by the global automatic QoS command, the phone has a few QoS features that you need to configure for proper labeling to occur. The QoS configuration information is sent to the phone through CDP from the switch. The QoS values that need to be configured are the trust setting of the "PC port" on the phone (trust or untrusted) and the CoS value that is used by the phone to remark the packets in case the port is untrusted (ext-cos).

AVVID recommends an untrusted and cos-ext value of 0. The PC traffic that enters the switch is marked with CoS 0 by the phone, the voice bearer traffic that is generated by the phone is always labeled with CoS 5, and the signaling is labeled with CoS 3.

Table 53-7 lists the port-specific settings that are implemented after executing the automatic QoS **ciscoipphone** macro on a port. See the "Port-Specific Automatic QoS—voip ciscoipphone" section on page 53-21 for detailed configuration examples.

Note You must enable CDP version 2 for trusted boundary to work. If CDP version 2 is not enabled, a syslog message displays. See the "CDP Warning" section on page 53-24.

Table 53-7 Port-Specific Settings for Voice (ciscoipphone Keyword)

Item	Value
Interface type	Port-based
Policy source—config	Local
Policy source—runtime	Local (as per default)
Trust type—config	Trust-cos
Trust type—runtime	Trust-cos
Default CoS—config	0 (as per default)
Default CoS—runtime	0 (as per default)
Trust-device	Ciscoipphone
QoS ACL attached to port	trust-cos any (if 1q4t/2q2t port, otherwise none)
QoS ACL name	ACL_IP-PHONES (if 1q4t/2q2t port, otherwise none)[1, 2, 3]
Trust-ext	Untrusted
Cos-ext	0

1. Only the IP QoS ACLs are applied (not IPX).
2. If the ACL_IP-PHONES name is already in use, the name ACL_IP-PHONES*x*, where *x* is a value from 1 to 99, will be tried sequentially. If all these names are taken, a syslog message displays.
3. ACL_IP-PHONES acl will not be created on WS-X6148-RJ-45 and WS-X6148-RJ-21 modules.

Port-Specific Automatic QoS Settings—ciscosoftphone

On the ports that connect to a Cisco SoftPhone, the QoS settings must be configured to trust the Layer 3 markings of the traffic that enters the port. Trusting all Layer 3 markings is a security risk because the PC users could send the nonpriority traffic with DSCP 46 and gain unauthorized performance benefits. Policing on all inbound traffic prevents the malicious users from obtaining unauthorized bandwidth from the network. Policing is accomplished by rate limiting the DSCP 46 (EF) inbound traffic to the codec rate that is used by the Cisco SoftPhone application (worst case G.722). Any traffic that exceeds this rate is marked down to the default traffic rate (DSCP 0 - BE). Signaling traffic (DSCP 24) is also policed and marked down to zero if excess signaling traffic is detected. All the other inbound traffic types are reclassified to default traffic (DSCP 0 – BE).

Caution You must disable trusted boundary for the Cisco SoftPhone ports.

Table 53-8 lists the port-specific settings that are implemented after executing the automatic QoS **voip ciscosoftphone** macro on a port. See the "Port-Specific Automatic QoS—voip ciscosoftphone" section on page 53-22 for detailed configuration examples.

Table 53-8 Port-Specific Settings for Voice (ciscosoftphone Keyword)

Item	Value
Interface type	Port-based
Policy source—config	Local
Policy source—runtime	Local

Table 53-8 Port-Specific Settings for Voice (ciscosoftphone Keyword) (continued)

Trust type—config	untrusted
Item	**Value**
Trust type—runtime	untrusted
Default CoS—config	0
Default CoS—runtime	0
Trust-device	none
Trust-ext	Untrusted
Cos-ext	0
QoS ACL attached to port	trust-dscp aggregate POLICE_SOFTPHONE-DSCP46-x-y any dscp-field 46[1, 2] trust-dscp aggregate POLICE_SOFTPHONE-DSCP24-x-y any dscp-field 24 *
QoS ACL name	ACL_IP-SOFTPHONES-x-y[3, 4]
QoS policers	aggregate POLICE_SOFTPHONE-DSCP46-3-1 rate 320 burst 20 policed-dscp aggregate POLICE_SOFTPHONE-DSCP24-3-1 rate 32 burst 8 policed-dscp
QoS policer names	POLICE_SOFTPHONE-DSCP46-x-y POLICE_SOFTPHONE-DSCP24-x-y

1. x = module number (interface on which the port-based automatic QoS macro is applied).
2. y = port number (if a range is specified, use the first number in the range).
3. Only the IP QoS ACLs are applied (not IPX).
4. If the ACL_IP-SOFTPHONE-x-y name is already in use, the name ACL_IP-SOFTPHONE-x-y-z, where z is a value from 1 to 99, will be tried sequentially. If all these names are taken, an error message displays. A similar action is taken with the policer name (see the "Out of Policer Names" section on page 53-24).

Policing Configuration for ciscosoftphone

Two rate limiters are associated with the interface on which the **ciscosoftphone** port-based automatic QoS macro is executed. The two rate limiters ensure that all inbound traffic on a Cisco SoftPhone port has the following characteristics:

1. The rate of DCSP 46 is at or less than that of the expected SoftPhone application rate (G.722 – worst case).
2. The rate of DSCP 24 is at or less than the expected signaling rate.
3. All other traffic is remarked to DSCP 0 (default traffic).

Action 3 is accomplished by the default QoS ACL. Any traffic that exceeds actions (1) or (2) is policed-dscp back to zero (remarked back to DSCP 0 - BE).

DSCP 46 is policed at the rate of 320 kbps with a burst of 20 kb. DSCP 24 is policed at 32 kbps with a burst of 8 kb. The burst and rate values are based on worst-case G.722 codec with a 256-kbps maximum packet length of 256 bytes and minor signaling with a maximum packet length of 1000 bytes. Signaling is transmitted with DSCP 24 and the bearer channel of the SoftPhone stream with DSCP 46.

The port is set to untrusted for all port types to prevent ingress QoS scheduling. The global automatic QoS macro configures the policed-dscp-map to make sure that DSCP 46 is marked down to DSCP 0 and that DSCP 24 is marked down to DSCP 0. The global automatic QoS macro configures the default QoS IP ACL that is used to remark all the other traffic to DSCP 0.

Limitations for ciscosoftphone

Because there is a limit on the total number of policers and QoS ACLs that are supported on the Catalyst 6500 series switches, similar limitations are associated with the **ciscosoftphone** automatic QoS macro. Up to 1023 aggregate policers are supported. Approximately 500 Cisco SoftPhone interfaces are supported (less interfaces are supported when other QoS ACLs and security ACLs are configured).

With a large number of Cisco SoftPhone interfaces, both the bootup time and NVRAM space could be affected. The bootup time increases with a large number of Cisco SoftPhone instances. It is possible to run out of NVRAM space with a high number of Cisco SoftPhone instances. To avoid running out of NVRAM space, you might need to use the text configuration mode. For more information, see the "Out of TCAM Space" section on page 53-23.

Port-Specific Automatic QoS Settings—trust cos

Use the **trust cos** automatic QoS keyword for the ports that require a "trust all" solution. Use the keyword only on the ports that connect other switches or known servers because the port trusts all inbound traffic marking in Layer 2 (CoS). Trusted boundary is disabled, and no QoS policing is configured on these types of ports.

Table 53-9 outlines the details of the configuration after executing the automatic QoS trust macro on a port. See the "Port-Specific Automatic QoS—trust cos" section on page 53-22 for configuration examples.

Table 53-9 Port-Specific Settings for Trust (trust cos Keyword)

Item	Value
Interface type	Port-based
Policy source—config	Local
Policy source—runtime	Local (as per default)
Trust type—config	Trust-cos
Trust type—runtime	Trust-cos
Default CoS—config	0 (as per default)
Default CoS—runtime	0 (as per default)
Trust-device	None
QoS ACL attached to port	trust-cos any (if 1q4t/2q2t port, otherwise none)
QoS ACL name	ACL_IP-TRUSTCOS (if 1q4t/2q2t port, otherwise none)[1, 2]
Trust-ext	Untrusted
Cos-ext	0

1. Only the IP QoS ACLs are applied (not IPX).
2. If the ACL_IP- TRUSTCOS name is already in use, the name ACL_IP- TRUSTCOS*x* , where *x* is a value from 1 to 99, will be tried sequentially. If all these names are taken, a syslog message is displayed.

Port-Specific Automatic QoS Settings—trust dscp

Use the **trust dscp** automatic QoS keyword for the ports that require a "trust all" type of solution. Use this keyword only on the ports that connect to the other switches or known servers because the port will be trusting all inbound traffic marking Layer 3 (DSCP). Trusted boundary is disabled, and no QoS policing is configured on these types of ports.

Table 53-10 outlines the details of the configuration after executing the automatic QoS trust macro on a port. See the "Port-Specific Automatic QoS Settings—trust dscp" section on page 53-13 for configuration examples.

Table 53-10 Port Specific Settings for Trusts (trust dscp Keyword)

Item	Value
Interface type	Port-based
Policy source—config	Local
Policy source—runtime	Local (as per default)
Trust type—config	Trust-dscp (all except 1q4t/2q2t ports) Untrusted (1q4t/2q2t ports)
Trust type—runtime	Trust-dscp (all except 1q4t/2q2t ports) Untrusted (1q4t/2q2t ports)
Default CoS—config	0 (as per default)
Default CoS—runtime	0 (as per default)
Trust-device	None
QoS ACL attached to port	trust-dscp any (if 1q4t/2q2t port, otherwise none)
QoS ACL name	ACL_IP-TRUSTDSCP (if 1q4t/2q2t port, otherwise none)[1, 2]
Trust-ext	Untrusted
Cos-ext	0

1. Only the IP QoS ACLs are applied (not IPX).
2. If the ACL_IP-TRUSTDSCP name is already in use, the name ACL_IP-TRUSTDSCP*x*, where *x* is a value from 1 to 99, will be tried sequentially. If all these names are taken, a syslog message is displayed.

CLI Interface for Automatic QoS

These sections describe the CLI interface for automatic QoS:

- Global Automatic QoS Macro—set qos autoqos, page 53-14
- Port-Specific Automatic QoS Macro—set port qos autoqos, page 53-14
- Displaying the QoS Settings, page 53-14
- Clearing the Automatic QoS Settings, page 53-15
- Tracking the QoS Configuration, page 53-17

Global Automatic QoS Macro—set qos autoqos

When you execute the global automatic QoS macro, all the global QoS settings are applied to all ports in the switch. After completion, a prompt displays showing the CLI for the port-based automatic QoS commands that are currently supported.

```
Console> (enable) set qos autoqos ?
Usage: set qos autoqos
Console> (enable) set qos autoqos
QoS is enabled.
........
All ingress and egress QoS scheduling parameters configured on all ports.
CoS to DSCP, DSCP to COS, IP Precedence to DSCP and policed dscp maps configured.
Global QoS configured, port specific autoqos recommended:
    set port qos <mod/port> autoqos trust <cos|dscp>
    set port qos <mod/port> autoqos voip <ciscoipphone|ciscosoftphone>
Console> (enable)
```

Port-Specific Automatic QoS Macro—set port qos autoqos

The port-specific automatic QoS macro accepts a *mod/port* combination and must include an AVVID-type keyword. The **ciscoipphone**, **ciscosoftphone**, and **trust** keywords are supported.

This example shows how to use the **ciscoipphone** keyword:

```
Console> (enable) set port qos 3/1 autoqos help
Usage: set port qos <mod/port> autoqos trust <cos|dscp>
       set port qos <mod/port> autoqos voip <ciscoipphone|ciscosoftphone>
Console> (enable) set port qos 3/1 autoqos voip ciscoipphone
Port 3/1 ingress QoS configured for Cisco IP Phone.
It is recommended to execute the "set qos autoqos" global command if not executed
previously.
Console> (enable)
```

This example shows how to use the **ciscosoftphone** keyword:

```
Console> (enable) set port qos 3/1 autoqos voip ciscosoftphone
Port 3/1 ingress QoS configured for Cisco Softphone.
It is recommended to execute the "set qos autoqos" global command if not executed
previously.
Console> (enable)
```

This example shows how to use the **trust cos** keyword:

```
Console> (enable) set port qos 3/1 autoqos trust cos
Port 3/1 QoS configured to trust all incoming CoS marking.
It is recommended to execute the "set qos autoqos" global command if not executed
previously.
Console> (enable)
```

This example shows how to use the **trust dscp** keyword:

```
Console> (enable) set port qos 3/1 autoqos trust dscp
Port 3/1 QoS configured to trust all incoming DSCP marking.
It is recommended to execute the "set qos autoqos" global command if not executed
previously.
Console> (enable)
```

Displaying the QoS Settings

Enter the existing QoS **show** commands to display the QoS settings. These commands include the **show port qos** and **show qos info runtime** commands.

Clearing the Automatic QoS Settings

You can clear the automatic QoS configuration by entering a port-based **clear** command and a global **clear** command. To clear the automatic QoS configuration, clear each interface on which automatic QoS has run with the port-based **clear** command and then enter the global **clear** command as described in the following sections:

- Clearing the Automatic QoS Port-Based Settings, page 53-15
- Clearing the Automatic QoS Global Settings, page 53-15

Clearing the Automatic QoS Port-Based Settings

All automatic QoS settings that are configured through the port-based automatic QoS command can be configured back to the factory-default settings by entering the **clear port qos** *mod/port* **autoqos** command, as follows:

```
Console> (enable) set port qos 3/1 autoqos voip ciscosoftphone
Port 3/1 ingress QoS configured for Cisco Softphone.
It is recommended to execute the "set qos autoqos" global command if not executed
previously.
Console> (enable) clear port qos ?
  <mod/port>                Module number and Port number(s)
Console> (enable) clear port qos 3/1 ?
  autoqos                   Clear port based autoqos settings
  cos                       Clear QoS default CoS value on ports
  cos-ext                   Clear QoS default CoS extension on ports
Console> (enable) clear port qos 3/1 autoqos
Port based QoS settings will be restored back to factory defaults for port 3/1.
Do you want to continue (y/n) [n]? y
Port 3/1 autoqos settings have been cleared.
It is recommended to execute the "clear qos autoqos" global command if
not executed previously to clear global autoqos settings.
Console> (enable)
```

The port-based **clear** command is supported on all ports that support the port-based automatic QoS **set** commands. All QoS settings that are configured through the automatic QoS port-based command revert back to the factory-default settings (with the exception of automatic QoS ACLs). All QoS ACLs that are mapped to the port are unmapped from the port, even if the QoS ACL is not related to automatic QoS. The QoS ACLs that are created for automatic QoS purposes are cleared when you enter the global **clear** command.

Clearing the Automatic QoS Global Settings

All QoS settings that are configured through the global automatic QoS command can be configured back to the factory-default settings by entering the **clear qos autoqos** command, as follows:

```
Console> (enable) clear qos autoqos
Its highly recommended to execute clear port autoqos commands prior
to the global clear command:
    clear port qos <mod/port> autoqos

Do you want to continue (y/n) [n]? y
.........................
Autoqos ACL 'ACL_IP-SOFTPHONE-3-1' successfully deleted.
Cleared Autoqos policer 'POLICE_SOFTPHONE-DSCP46-3-1'
Cleared Autoqos policer 'POLICE_SOFTPHONE-DSCP24-3-1'
```

```
All ingress and egress QoS scheduling parameters set to factory default.
CoS to DSCP, DSCP to COS, IP Precedence to DSCP and policed dscp maps
configured.  Global Autoqos QoS cleared.
Console> (enable)
```

The QoS ACLs that are created through the **set port autoqos** commands are cleared when you enter the global automatic QoS **clear** command. In addition, any policers that are used by the automatic QoS ACLs are cleared.

The global automatic QoS **clear** command searches for the automatic QoS ACL names. The search algorithm looks for names that begin with these strings:

- ACL_IP-PHONES (for ciscoipphone)
- ACL_IP-SOFTPHONE (for ciscosoftphone)
- ACL_IP-TRUSTCOS (for trust cos)
- ACL_IP-TRUSTDSCP (for trust dscp)

Any QoS ACL that starts with the above strings is considered an automatic QoS ACL and is cleared. If one is found and the QoS ACL is committed and not mapped to a port or a VLAN, the automatic QoS ACL is deleted.

Similarly, the search algorithm looks for the aggregate QoS policers starting with the name: POLICE_SOFTPHONE-DSCP (for ciscosoftphone).

The global **clear** command searches for the aggregate policer names that begin with POLICE_SOFTPHONE-DSCP. If a policer is found, and there is no QoS ACL that is associated with it, it is deleted. If a policer is found, and there is a QoS ACL that is associated with it, a warning is displayed indicating that the policer is still in use.

Various error conditions can occur when you use the global **clear** command. If you have properly executed the port-based **clear** commands before entering the global **clear** command, no error conditions should occur. However, if you execute the global **clear** command first or modify the automatic QoS configuration, these error conditions could occur:

- The automatic QoS ACLs are still mapped to a port or VLAN.

  The global **clear** command does not clear the automatic QoS ACLs that are still mapped to a VLAN or port. Instead, the command displays a warning indicating the name of the QoS ACL that is still mapped to a port/VLAN.

- The aggregate policers are still in use.

  If the automatic QoS policers are still in use (referenced by a QoS ACL), the global **clear** command does not remove them. Instead, it displays the name of the aggregate policer.

- The automatic QoS ACLs are uncommitted.

  The global **clear** command removes only the committed automatic QoS ACLs but ignores the uncommitted automatic QoS ACLs.

This example shows what is displayed under these various error conditions:

```
Console> (enable) clear qos autoqos
Its highly recommended to execute clear port autoqos commands prior
to the global clear command:
    clear port qos <mod/port> autoqos
```

```
Do you want to continue (y/n) [n]? y
.....................
Autoqos ACL 'ACL_IP-SOFTPHONE-3-2' successfully deleted.
Autoqos ACL 'ACL_IP-SOFTPHONE-3-3' successfully deleted.
Autoqos ACL 'ACL_IP-SOFTPHONE-3-4' still mapped to port or vlan.
Autoqos ACL 'ACL_IP-SOFTPHONE-3-5' still mapped to port or vlan.
Autoqos ACL 'ACL_IP-SOFTPHONE-3-6' still mapped to port or vlan.
Cleared Autoqos policer 'POLICE_SOFTPHONE-DSCP46-3-2'
Cleared Autoqos policer 'POLICE_SOFTPHONE-DSCP24-3-2'
Cleared Autoqos policer 'POLICE_SOFTPHONE-DSCP46-3-3'
Cleared Autoqos policer 'POLICE_SOFTPHONE-DSCP24-3-3'
Could not clear Autoqos policer ''POLICE_SOFTPHONE-DSCP46-3-4', still in use.
QoS is disabled.

All ingress and egress QoS scheduling parameters set to factory default.
CoS to DSCP, DSCP to COS, IP Precedence to DSCP and policed dscp maps
configured.  Global Autoqos QoS cleared.
Console> (enable)
```

Tracking the QoS Configuration

A configuration "comment" appears in the configuration file to help you determine where the QoS configuration originated: Traditional QoS or automatic QoS. The comment is created after you enter the global **set qos autoqos** command and remains in the configuration file until you enter either the **clear global autoqos** command or the **clear qos config** command. An example is as follows:

```
Console> (enable) set qos autoqos
..............
All ingress and egress QoS scheduling parameters configured on all ports.
CoS to DSCP, DSCP to COS, IP Precedence to DSCP and policed dscp maps
configured.  Global QoS configured, port specific autoqos recommended:
    set port qos <mod/port> autoqos trust <cos|dscp>
    set port qos <mod/port> autoqos voip <ciscoipphone|ciscosoftphone>
Console> (enable) show config
This command shows non-default configurations only.
Use 'show config all' to show both default and non-default configurations.
..................


.............................

.........................

..

begin
<snip>
#qos - qos configuration via autoqos
set qos enable
set qos map 2q2t tx 2 1 cos 1
set qos map 2q2t tx 2 1 cos 2
<snip>
Console> (enable) clear qos autoqos
Its highly recommended to execute clear port autoqos commands prior
to the global clear command:
    clear port qos <mod/port> autoqos

Do you want to continue (y/n) [n]? y
.....................

No Autoqos ACLs found.
No Autoqos aggregate policer(s) found.
```

```
QoS is disabled.

All ingress and egress QoS scheduling parameters set to factory default.
CoS to DSCP, DSCP to COS, IP Precedence to DSCP and policed dscp maps
configured.  Global Autoqos QoS cleared.
Console> (enable) show config
This command shows non-default configurations only.
Use 'show config all' to show both default and non-default configurations.
.................

<snip>
#qos
<snip>
Console> (enable)
```

Detailed Automatic QoS Configuration Statements

These sections provide the detailed automatic QoS configuration statements:

Global Automatic QoS Macro

Entering the global automatic QoS command results in the following configuration:

```
set qos autoqos
---------------
set qos enable

set qos policy-source local
set qos ipprec-dscp-map 0 10 18 26 34 46 48 56
set qos cos-dscp-map 0 10 18 26 34 46 48 56
set qos dscp-cos-map 0-7:0 8-15:1 16-23:2 24-31:3 32-39:4 40-47:5 48-55:6 56-63:7
set qos acl default-action ip dscp 0
set qos map 2q2t tx queue 2 2 cos 5,6,7
set qos map 2q2t tx queue 2 1 cos 1,2,3,4
set qos map 2q2t tx queue 1 1 cos 0
set qos drop-threshold 2q2t tx queue 1 100 100
set qos drop-threshold 2q2t tx queue 2 80 100
set qos drop-threshold 1q4t rx queue 1 50 60 80 100
set qos txq-ratio 2q2t 80 20
set qos wrr 2q2t 100 255

set qos map 1p3q1t tx 1 1 cos 0
set qos map 1p3q1t tx 2 1 cos 1,2
set qos map 1p3q1t tx 3 1 cos 3,4
set qos map 1p3q1t tx 3 0 cos 6,7
set qos map 1p3q1t tx 4 cos 5
set qos wrr 1p3q1t 20 100 200
set qos wred 1p3q1t queue 1 70:100
set qos wred 1p3q1t queue 2 70:100
set qos wred 1p3q1t queue 3 70:90
set qos map 1p1q0t rx 1 cos 0,1,2,3,4
set qos map 1p1q0t rx 2 cos 5,6,7
```

```
set qos rxq-ratio 1p1q0t 80 20
set qos map 1p2q2t tx 1 2 cos 0
set qos map 1p2q2t tx 2 1 cos 1,2,3,4
set qos map 1p2q2t tx 2 2 cos 6,7
set qos map 1p2q2t tx 3 cos 5
set qos txq-ratio 1p2q2t 75 15 15
set qos wrr 1p2q2t 50 255
set qos wred 1p2q2t queue 1 1 40:70
set qos wred 1p2q2t queue 1 2 70:100
set qos wred 1p2q2t queue 2 1 40:70
set qos wred 1p2q2t queue 2 2 70:100
set qos map 1p1q4t rx 1 1 cos 0
set qos map 1p1q4t rx 1 3 cos 1,2,3,4
set qos map 1p1q4t rx 1 4 cos 6,7
set qos map 1p1q4t rx 2 cos 5
set qos drop-threshold 1p1q4t rx queue 1 50 60 80 100

set qos map 1p2q1t tx 1 1 cos 0
set qos map 1p2q1t tx 2 1 cos 1,2,3,4
set qos map 1p2q1t tx 2 cos 6,7
set qos map 1p2q1t tx 3 cos 5
set qos txq-ratio 1p2q1t 75 15 15
set qos wrr 1p2q1t 50 255
set qos wred 1p2q1t queue 1 70:100
set qos wred 1p2q1t queue 2 70:100
set qos map 1p1q8t rx 1 1 cos 0
set qos map 1p1q8t rx 1 5 cos 1,2
set qos map 1p1q8t rx 1 8 cos 3,4
set qos map 1p1q8t rx 2 cos 5,6,7
set qos wred 1p1q8t queue 1 1 40:70
set qos wred 1p1q8t queue 1 5 60:90
set qos wred 1p1q8t queue 1 8 70:100
set qos rxq-ratio 1p1q8t 80 20
set qos policed-dscp-map 0:0
set qos policed-dscp-map 1:1
set qos policed-dscp-map 2:2
set qos policed-dscp-map 3:3
set qos policed-dscp-map 4:4
set qos policed-dscp-map 5:5
set qos policed-dscp-map 6:6
set qos policed-dscp-map 7:7
set qos policed-dscp-map 8:8
set qos policed-dscp-map 9:9
set qos policed-dscp-map 10:10
set qos policed-dscp-map 11:11
set qos policed-dscp-map 12:12
set qos policed-dscp-map 13:13
set qos policed-dscp-map 14:14
set qos policed-dscp-map 15:15
set qos policed-dscp-map 16:16
set qos policed-dscp-map 17:17
set qos policed-dscp-map 18:18
set qos policed-dscp-map 19:19
set qos policed-dscp-map 20:20
set qos policed-dscp-map 21:21
set qos policed-dscp-map 22:22
set qos policed-dscp-map 23:23
set qos policed-dscp-map 24:24
set qos policed-dscp-map 25:25
set qos policed-dscp-map 26:0
set qos policed-dscp-map 27:27
set qos policed-dscp-map 28:28
set qos policed-dscp-map 29:29
set qos policed-dscp-map 30:30
```

```
set qos policed-dscp-map 31:31
set qos policed-dscp-map 32:32
set qos policed-dscp-map 33:33
set qos policed-dscp-map 34:34
set qos policed-dscp-map 35:35
set qos policed-dscp-map 36:36
set qos policed-dscp-map 37:37
set qos policed-dscp-map 38:38
set qos policed-dscp-map 39:39
set qos policed-dscp-map 40:40
set qos policed-dscp-map 41:41
set qos policed-dscp-map 42:42
set qos policed-dscp-map 43:43
set qos policed-dscp-map 44:44
set qos policed-dscp-map 45:45
set qos policed-dscp-map 46:0
set qos policed-dscp-map 47:47
set qos policed-dscp-map 48:48
set qos policed-dscp-map 49:49
set qos policed-dscp-map 50:50
set qos policed-dscp-map 51:51
set qos policed-dscp-map 52:52
set qos policed-dscp-map 53:53
set qos policed-dscp-map 54:54
set qos policed-dscp-map 55:55
set qos policed-dscp-map 56:56
set qos policed-dscp-map 57:57
set qos policed-dscp-map 58:58
set qos policed-dscp-map 59:59
set qos policed-dscp-map 60:60
set qos policed-dscp-map 61:61
set qos policed-dscp-map 62:62
set qos policed-dscp-map 63:63
set qos policed-dscp-map excess-rate 0:0
set qos policed-dscp-map excess-rate 1:1
set qos policed-dscp-map excess-rate 2:2
set qos policed-dscp-map excess-rate 3:3
set qos policed-dscp-map excess-rate 4:4
set qos policed-dscp-map excess-rate 5:5
set qos policed-dscp-map excess-rate 6:6
set qos policed-dscp-map excess-rate 7:7
set qos policed-dscp-map excess-rate 8:8
set qos policed-dscp-map excess-rate 9:9
set qos policed-dscp-map excess-rate 10:10
set qos policed-dscp-map excess-rate 11:11
set qos policed-dscp-map excess-rate 12:12
set qos policed-dscp-map excess-rate 13:13
set qos policed-dscp-map excess-rate 14:14
set qos policed-dscp-map excess-rate 15:15
set qos policed-dscp-map excess-rate 16:16
set qos policed-dscp-map excess-rate 17:17
set qos policed-dscp-map excess-rate 18:18
set qos policed-dscp-map excess-rate 19:19
set qos policed-dscp-map excess-rate 20:20
set qos policed-dscp-map excess-rate 21:21
set qos policed-dscp-map excess-rate 22:22
set qos policed-dscp-map excess-rate 23:23
set qos policed-dscp-map excess-rate 24:24
set qos policed-dscp-map excess-rate 25:25
set qos policed-dscp-map excess-rate 26:26
set qos policed-dscp-map excess-rate 27:27
set qos policed-dscp-map excess-rate 28:28
set qos policed-dscp-map excess-rate 29:29
set qos policed-dscp-map excess-rate 30:30
```

```
set qos policed-dscp-map excess-rate 31:31
set qos policed-dscp-map excess-rate 32:32
set qos policed-dscp-map excess-rate 33:33
set qos policed-dscp-map excess-rate 34:34
set qos policed-dscp-map excess-rate 35:35
set qos policed-dscp-map excess-rate 36:36
set qos policed-dscp-map excess-rate 37:37
set qos policed-dscp-map excess-rate 38:38
set qos policed-dscp-map excess-rate 39:39
set qos policed-dscp-map excess-rate 40:40
set qos policed-dscp-map excess-rate 41:41
set qos policed-dscp-map excess-rate 42:42
set qos policed-dscp-map excess-rate 43:43
set qos policed-dscp-map excess-rate 44:44
set qos policed-dscp-map excess-rate 45:45
set qos policed-dscp-map excess-rate 46:46
set qos policed-dscp-map excess-rate 47:47
set qos policed-dscp-map excess-rate 48:48
set qos policed-dscp-map excess-rate 49:49
set qos policed-dscp-map excess-rate 50:50
set qos policed-dscp-map excess-rate 51:51
set qos policed-dscp-map excess-rate 52:52
set qos policed-dscp-map excess-rate 53:53
set qos policed-dscp-map excess-rate 54:54
set qos policed-dscp-map excess-rate 55:55
set qos policed-dscp-map excess-rate 56:56
set qos policed-dscp-map excess-rate 57:57
set qos policed-dscp-map excess-rate 58:58
set qos policed-dscp-map excess-rate 59:59
set qos policed-dscp-map excess-rate 60:60
set qos policed-dscp-map excess-rate 61:61
set qos policed-dscp-map excess-rate 62:62
set qos policed-dscp-map excess-rate 63:63
```

Port-Specific Automatic QoS—voip ciscoipphone

Entering the port-specific automatic QoS command results in the following configuration:

set port qos *mod/port* **autoqos voip ciscoipphone**

```
---------------
set port qos mod/port policy-source local
set port qos mod/port port-based
set port qos mod/port cos 0
set port qos mod/port cos-ext 0
set port qos mod/port trust-ext untrusted
set port qos mod/port trust-device ciscoipphone
```

If the port type is 1q4t/2q2t, the configuration is as follows:

```
set qos acl ip ACL_IP-PHONES trust-cos any
commit qos acl ACL_IP-PHONES
set qos acl map ACL_IP-PHONES mode/port
set port qos mod/port trust trust-cos
```

If the port type is another port type, the configuration is as follows:

```
set port qos mod/port trust trust-cos
```

Note If the ACL_IP-PHONES name is in use, automatic QoS checks if the existing ACL is the same as the one that is trying to be created. If the existing QoS ACL is the same, automatic QoS reuses it. If the existing QoS ACL is not the same, automatic QoS attempts other names.

Port-Specific Automatic QoS—voip ciscosoftphone

Entering the port-specific automatic QoS command results in the following configuration:

set port qos *mod/port* **autoqos voip ciscosoftphone**

```
---------------
set port qos mod/port policy-source local
set port qos mod/port port-based
set port qos mod/port cos 0
set port qos mod/port cos-ext 0
set port qos mod/port trust-ext untrusted
set port qos mod/port trust-device none
set port qos mod/port trust untrusted
set qos policer aggregate POLICE_SOFTPHONE-DSCP46-mod-port rate 320 burst 20 policed-dscp
set qos policer aggregate POLICE_SOFTPHONE-DSCP26-mod-port rate 32  burst 8 policed-dscp
set qos acl ip ACL_IP-SOFTPHONE-mod-port trust-dscp aggregate
POLICE_SOFTPHONE-DSCP46-mod-port any dscp-field 46
set qos acl ip ACL_IP-SOFTPHONE-mod-port trust-dscp aggregate
POLICE_SOFTPHONE-DSCP26-mod-port any dscp-field 26
commit qos acl ACL_IP-SOFTPHONE-mod-port
set qos acl map ACL_IP-SOFTPHONE-mod-port mod/port
```

Port-Specific Automatic QoS—trust cos

Entering the port-specific automatic QoS command results in the following configuration:

set port qos *mod/port* **autoqos trust cos**

```
---------------
set port qos mod/port policy-source local
set port qos mod/port port-based
set port qos mod/port cos 0
set port qos mod/port cos-ext 0
set port qos mod/port trust-ext untrusted
set port qos mod/port trust-device none
```

If the port type is 1q4t/2q2t, the configuration is as follows:

```
set qos acl ip ACL_IP-TRUSTCOS trust-cos any
commit qos acl ACL_IP-TRUSTCOS
set qos acl map ACL_IP-TRUSTCOS mode/port
set port qos mod/port trust trust-cos
```

If the port type is another port type, the configuration is as follows:

```
set port qos mod/port trust trust-cos
```

Port-Specific Automatic QoS—trust dscp

Entering the port-specific automatic QoS command results in the following configuration:

set port qos *mod/port* **autoqos trust dscp**

```
---------------
set port qos mod/port policy-source local
set port qos mod/port port-based
set port qos mod/port cos 0
set port qos mod/port cos-ext 0
set port qos mod/port trust-ext untrusted
set port qos mod/port trust-device none
```

If the port type is 1q4t/2q2t, the configuration is as follows:

```
set qos acl ip ACL_IP-TRUSTDSCP trust-dscp any
commit qos acl ACL_IP-TRUSTDSCP
set qos acl map ACL_IP-TRUSTDSCP mode/port
set port qos mod/port trust untrusted
```

If the port type is another port type, the configuration is as follows:

```
set port qos mod/port trust trust-dscp
```

Warning and Error Conditions

These sections describe the warnings and error conditions for automatic QoS:

Out of ACL Names

When creating a QoS ACL for a 1q4t/2q2t type port to fix the trust problem, you may note that the following QoS ACL names are already in use (where x=1 to 99):

- ACL_IP-PHONESx (for **ciscoipphone**)
- ACL_IP-SOFTPHONE-m-p-x (for **ciscosoftphone**)
- ACL_IP-TRUSTCOSx (for **trust cos**)
- ACL_IP-TRUSTDSCPx (for **trust dscp**)

This example shows the display when the system is out of ACL names:

```
Console> (enable) set port qos 4/1 autoqos voip ciscoipphone
ERROR:  IP QoS ACL name in use, could not configure QoS ACL.
Rename existing QoS ACL ACL_IP-PHONES.
Autoqos did not complete.  Settings remain unchanged.
Console> (enable)
```

Out of TCAM Space

When configuring the ACLs using the port-based automatic QoS command, it is possible to have a full TCAM. In this event, an error message displays and the port-based automatic QoS command fails, leaving all QoS settings unchanged.

This example shows the display when the system is out of TCAM space:

```
Console> (enable) set port qos 4/1 autoqos voip ciscoipphone
Error:  Please remove QoS or security ACLs to make space for new QoS ACL.
Autoqos did not complete.  Settings remain unchanged.
Console> (enable)
```

COPS Warning Message

If COPS has been enabled globally or enabled on a port, executing the global automatic QoS command or the port-specific automatic QoS command changes the policy source to local and a warning message displays.

This example shows that if the port-based command is successful, the port-based policy setting is changed to local as follows:

```
Console> (enable) set port qos 4/1 autoqos voip ciscosoftphone
Warning:  QoS policy changed to local for port 4/1.
Port 4/1 ingress QoS configured for ciscosoftphone.
It is recommended to execute the "set qos autoqos" global command if not executed
previously.
```

This example for the global command shows that if the global QoS policy is COPS before the global automatic QoS command is executed, a warning message displays as follows:

```
Console> (enable) set qos autoqos
........
Warning: QoS policy source changed to local.
All ingress and egress QoS scheduling parameters configured on all ports.
CoS to DSCP, DSCP to COS and IP Precedence to DSCP maps configured.
Global QoS configured, port specific autoqos recommended:
   set port qos <mod/port> autoqos trust [cos|dscp]
   set port qos <mod/port> autoqos voip [ciscoipphone|ciscosoftphone]
Console> (enable)
```

CDP Warning

When executing the port-specific automatic QoS command with the **ciscoipphone** keyword without the trust option, the trust-device feature is enabled. The trust-device feature is dependent on CDP. If CDP is not enabled or not running version 2, a warning message displays as follows:

```
Console> (enable) set port qos 4/1 autoqos voip ciscoipphone
Warning:  CDP is disabled or CDP version 1 is in use.  Ensure that CDP version 2 is
enabled globally, and also ensure that CDP is enabled on the port(s) you wish to configure
autoqos on.
Port 4/1 ingress QoS configured for ciscoipphone.
It is recommended to execute the "set qos autoqos" global command if not executed
previously.
Console> (enable)
```

Out of Policer Names

When executing the port-specific automatic QoS command with the **ciscosoftphone** keyword, two policer instances are created and named with the following strings:

- POLICE_SOFTPHONE-DSCP46-x-y
- POLICE_SOFTPHONE-DSCP26-x-y

where x is the module number and y is the port number of the *mod/port* combination that is specified with the **ciscosoftphone** keyword.

If the above policer names are already in use, the macro attempts the following names:

- POLICE_SOFTPHONE-DSCP46-x-y-z
- POLICE_SOFTPHONE-DSCP26-x-y-z

where z = 1 to 99 starting from 1. Both names must pass with the same z value or the macro attempts the next z value until both names are valid with the same z value. If the z = 99 attempt fails, this error message displays and all settings remain unchanged:

```
Console> (enable) set port qos 4/1 autoqos voip ciscosoftphone
ERROR:  QoS policer name in use, could not configure QoS policer.
Rename existing QoS policer POLICE_SOFTPHONE-DSCP46-4-1 and/or
POLICE_SOFTPHONE-DSCP26-4-1.
Autoqos did not complete.  Settings remain unchanged.
Console> (enable)
```

QoS Disabled

When executing any port-based automatic QoS command on an interface where QoS is disabled, a notification message appears in the CLI as follows:

```
Console> (enable) set port qos 4/1 autoqos voip ciscosoftphone
Port 4/1 ingress QoS configured for ciscosoftphone.  Policing configured on 4/1.
QoS is disabled, changes will take effect after QoS is enabled.
It is recommended to execute the "set qos autoqos" global command if not executed
previously.
Console> (enable)
```

syslog Additions

Set the switch logging level to 4 or 5 for the QoS facility (**set logging level qos 5**) as follows:

- Log level 4 = Warnings
- Log level 5 = Notices

These sections describe the syslog additions for the automatic QoS features:

- CDP Warning —Warning Level, page 53-25
- Interface Change for All Ports Required—Warning Level, page 53-26

CDP Warning —Warning Level

When executing the port-based automatic QoS **voip ciscoipphone** keyword on a port where either CDP is disabled on the port or globally, or is running in version 1 mode, a warning message displays as follows:

```
Console> (enable)
2001 Jun 02 09:20:42 %QOS-4-DEVICE_CDP_DIS:Trust-Device feature enabled with CDP disabled
or running in v1 mode.
Console> (enable)
```

Interface Change for All Ports Required—Warning Level

For the 1p1q0t/1p3q1t ports, if a change in an interface type is needed (if VLAN-based mode is configured before the automatic QoS macro is executed), a syslog message displays indicating that all ports in the module had the interface type changed to port-based QoS, as follows:

```
Console> (enable)
2001 Jun 02 09:20:42 %QOS-3-INTERFACE-CHANGED:All ports in module 3 have been configured
to port-based QoS.
Console> (enable)
```

Other Relevant syslog Messages

These sections describe the other relevant syslog messages that relate to the automatic QoS configuration:

- Device No Longer Detected on the Port—Notice Level (Trusted Boundary), page 53-26
- Device Detected on the Port—Notice Level, page 53-26
- CDP Disabled with Trust-Dev Configured—Warning Level, page 53-26

Device No Longer Detected on the Port—Notice Level (Trusted Boundary)

After enabling trusted boundary on a port (using the **ciscoipphone** keyword), if the phone is detected to have left the port, a syslog message displays stating that the device has left and the port trust state has been changed, as follows:

```
Console> (enable)
2001 Jun 02 09:20:42 %QOS-5-DEVICE_LOST:ciscoipphone not detected on port 4/1, port set to
untrusted.
Console> (enable)
```

Device Detected on the Port—Notice Level

If the trusted device joins the port, a syslog message displays indicating the change in the port trust status. The heading contains the new trust type of the port as specified in the configuration. This example shows that the port trust for port 4/1 is set to “trust-cos” in the configuration:

```
Console> (enable)
2001 Jun 02 09:20:42 %QOS-5-DEVICE_DETECTED:ciscoipphone detected on port 4/1, port set to
trust-cos.
Console> (enable)
```

CDP Disabled with Trust-Dev Configured—Warning Level

When executing the port-based automatic QoS **ciscoipphone** keyword on a port, the trust-device is configured to “ciscoipphone” which activates trusted boundary. After trusted boundary is enabled, if either CDP is disabled on that port or CDP is running in version 1 mode, or CDP is globally disabled, a syslog message displays as follows:

```
Console> (enable)
2001 Jun 02 09:20:42 %QOS-4-DEVICE_CDP_DIS:Trust-Device feature enabled with CDP disabled
or running in v1 mode.
Console> (enable)
```

This message is displayed only once when a problem is detected. When the problem is fixed, the message can appear again if the configuration is broken again. There is a maximum time of 15 seconds for detecting a misconfiguration.

Summary of Automatic QoS Features

These sections summarize the automatic QoS features:

- Global Automatic QoS Features (set qos autoqos), page 53-27
- Port-Based Automatic QoS Features, page 53-27

Global Automatic QoS Features (set qos autoqos)

The global automatic QoS feature is summarized as follows:

- Configures all switch-wide QoS parameters to accommodate and properly prioritize all traffic types that are listed in the "Typical CoS and DSCP Values for Voice and Video Networks" section on page 53-2.
- Overwrites any old or misconfigured settings that were previously applied.
- Works with the port-based automatic QoS command.

Port-Based Automatic QoS Features

The port-based automatic QoS features are summarized as follows:

- voip ciscoipphone
  - Changes the port to port-based QoS.
  - For the 1p1q0t/1p3q1t ports, changes all ports to port-based mode.
  - Creates a trust-cos QoS ACL for the ports that need it (1q4t/2q2t ports).
  - Applies the trust-cos ACL to the port (1q4t/2q2t ports).
  - Enables trusted boundary on the port.
  - Sets the port trust to trust-cos.
  - Supports the ports with or without an auxiliary VLAN.
  - Supported only on the 10/100 ports and the 10/100/1000 ports.
  - PFC or PFC2 not required (PFC and PFC2 are supported).
- voip ciscosoftphone
  - Changes the port to port-based QoS.
  - Changes trust to untrusted.
  - For the 1p1q0t/1p3q1t ports, changes all ports to port-based mode.
  - Disables trusted boundary on the port.
  - Applies two rate limiters, one for DSCP 46 and one for DSCP 26 inbound traffic, and trusts only inbound DSCP 46 and DSCP 26 traffic.
  - Results in traffic that is marked down to DSCP 0 for violations of either rate limiter.
  - Remarks all other (non-DSCP 26 and 46) inbound traffic to DSCP 0.

  - Supports the ports with or without an auxiliary VLAN.
  - Supported on all ports.
  - Requires the PFC or the PFC2.
- trust cos
  - Changes the port to port-based QoS.
  - For the 1p1q0t/1p3q1t ports, changes all ports to port-based mode.
  - Creates a trust-cos QoS ACL for the ports that need it (1q4t/2q2t ports).
  - Applies the trust-cos ACL to the port (1q4t/2q2t ports).
  - Disables trusted boundary on the port.
  - Sets port trust to trust-cos.
  - Supports the ports with or without an auxiliary VLAN.
  - Supported on all ports.
  - PFC not required (PFC and PFC2 are supported).
- trust dscp
  - Changes the port to port-based QoS.
  - For the 1p1q0t/1p3q1t ports, changes all ports to port-based mode.
  - Creates a trust-dscp QoS ACL for the ports that need it (1q4t/2q2t ports).
  - Applies the trust-dscp ACL to the port (1q4t/2q2t ports).
  - Disables trusted boundary on the port.
  - Sets port trust to untrusted (1q4t/2q2t ports) or trust-dscp (not on 1q4t/2q2t ports).
  - Supports the ports with or without an auxiliary VLAN.
  - Supported on all ports.
  - Requires the PFC or the PFC2.

Using Automatic QoS in Your Network

Tip To ensure that automatic QoS works properly, you should execute the global automatic QoS macro and, for each interface, you should execute the interface-specific automatic QoS macro.

Depending on the interface and what is connected to it, you will need to execute different automatic QoS macros. To execute the global automatic QoS macro, and then for each interface, execute the interface-specific automatic QoS macro with the appropriate keyword, perform these steps:

Step 1 Execute the **set qos autoqos** command to enable QoS and configure all the outbound QoS settings.

Step 2 For each port, execute the port-based automatic QoS commands as shown in Table 53-11.

Table 53-11 Using Automatic QoS Keywords

Keyword	Port Type
ciscoipphone	Ports that connect only a Cisco IP Phone 79xx.
ciscoipphone	Ports that connect a Cisco IP Phone 79xx with a PC connected to the 79xx.
ciscoipphone	Ports that connect a Cisco IP Phone 79xx with a PC connected to the 79xx running Cisco SoftPhone[1].
ciscosoftphone	Ports that connect a PC running Cisco SoftPhone without a Cisco IP Phone 79xx.
trust	Ports that connect to other places in the network where all automatic QoS traffic types exist[2].

1. For cases where ports connect a Cisco IP Phone 79xx with a PC running Cisco SoftPhone, the control traffic through CTI communication with the Cisco CallManager is tagged but is remarked to DSCP 0.
2. For ports connecting to other networks or Cisco CallManagers, we recommend that you use the **trust** keyword. Currently, Cisco CallManager and gateways correctly mark skinny, H.323, and MGCP signaling traffic. However, some versions of Cisco CallManager do not explicitly mark H.323 and MGCP traffic. We recommend QoS ACLs for these situations.

CHAPTER 54

Configuring ASLB

This chapter describes how to configure accelerated server load balancing (ASLB) on the Catalyst 6500 series switches.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

Note The information and procedures in this chapter apply only to Supervisor Engine 1 with the Policy Feature Card. ASLB is not supported on Supervisor Engine 2 with PFC2, Supervisor Engine 720 with PFC3A/PFC3B/PFC3BXL, or Supervisor Engine 32 with PFC3B/PFC3BXL.

This chapter consists of these sections:

Hardware and Software Requirements

The hardware and software requirements for your ASLB configuration are as follows:

- The LocalDirector requirements are as follows:
  - Hardware platforms—LocalDirector models 410, 415, 416, 420, or 430
  - Interface Modules—The ASLB configuration requires two 10/100BASE-X Ethernet interfaces or two 1000BASE-X Gigabit Ethernet interfaces

Note The 1000BASE-X interfaces are supported only on the LocalDirector 420 and 430; they are not supported on the LocalDirector 410, 415, or 416.

  - Software—Cisco configuration version 3.2.x
- The Catalyst 6500 series switch requirements are as follows:
  - Supervisor Engine 1 (or 1A) with the PFC
  - Supervisor engine software release 5.3(1)CSX or later releases
- The participating routers are as follows:
  - Multilayer Switch Feature Card (MSFC)—With supervisor engine software release 5.4(1)CSX or later releases, an MSFC in the Catalyst 6500 series switch can be used as a participating router for ASLB. With earlier supervisor engine software releases, an internal MSFC *cannot* be a participating router.
  - External MSFC—An MSFC in an externally attached Catalyst 6500 series switch can be used as a participating router.
  - Multilayer Switch Module (MSM)—If the Catalyst 6500 series switch that you are using for ASLB has an MSM, it can be used as a participating router for ASLB. The MSM in an externally attached Catalyst 6500 series switch can also be used as a participating router.
  - Other Cisco routers can also be used as participating routers for ASLB.

Understanding How ASLB Works

Note Refer to the *Cisco LocalDirector Installation and Configuration Guide,* Version 3.2, for an overview on load balancing TCP/IP traffic.

These sections describe ASLB:

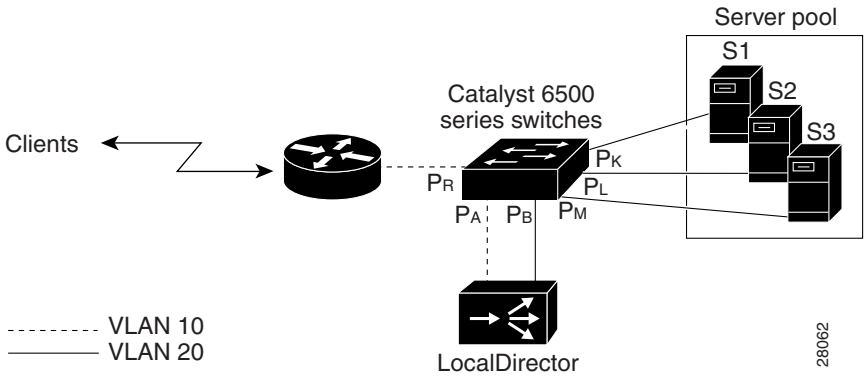
The LocalDirector is a secure, real-time, embedded operating system that intelligently load balances the TCP/IP traffic across multiple servers. ASLB enables Catalyst 6500 series switches to cache the Cisco LocalDirector load-balancing flows, accelerating the performance of the LocalDirector.

Note The accelerated performance of the LocalDirector is achieved through the Catalyst 6500 series Layer 3 switching technology.

Figure 54-1 shows a network that uses the ASLB feature. You must connect the LocalDirector to the switch with two links; one link connects to the same VLAN that the router is on and the other link connects to the VLAN that the servers are on. In Figure 54-1, one LocalDirector link is connected to VLAN 10, the router VLAN; the other link is connected to VLAN 20, the server VLAN.

The LocalDirector supports directed mode and dispatched mode. Only the dispatched mode can be supported for ASLB feature implementation on Catalyst 6500 series switches.

Figure 54-1 ASLB Functional Description



Layer 3 Operations for ASLB

You can specify up to 1024 server virtual-IP addresses and TCP port pairs for acceleration by the switch. All the traffic for the virtual-IP/port pairs specified is accelerated except for the SYN, FIN, RST, and fragment packets with a nonzero offset. These packets are redirected to both the active and standby LocalDirectors (if a backup LocalDirector is configured).

Layer 2 Operations for ASLB

The Catalyst 6500 series switch content-addressable memory (CAM) table contains entries for the router VLAN and the server VLAN. In the CAM table, the router VLAN has an entry for the MAC address of the LocalDirector that is associated with a port index, and the server VLAN has entries for the router MAC addresses that are associated with the port indexes. In these port indexes, the ports appear as 0/0. You can display system CAM entries by entering the **show cam system** command.

Table 54-1 shows the entries in the CAM table (the ASLB configuration is shown in Figure 54-1). The first entry identifies the MAC address of the LocalDirector on VLAN 10. The CAM table shows that the MAC address has an Xtag value of 14. This value indicates that the MAC address requires a Layer 3 lookup. The second entry identifies the MAC address of the router and also requires a Layer 3 lookup.

Table 54-1 Layer 2 Table Entries

VLAN	MAC Address	Index	Xtag[1]
10	LocalDirector MAC	0/0	14
20	Router MAC[2]	0/0	14

1. Xtag = The identifier field in the Layer 2 table that identifies the router to which the MAC address belongs.
2. Note that the router MAC address is added on the server VLAN (VLAN 20), not on the router VLAN (VLAN 10).

Client-to-Server Data Forwarding

Figure 54-2 shows how the data is forwarded from the router to the servers. Table 54-2 lists the sequence of events, and Table 54-3 lists the Layer 3 table entries.

These sections describe the client-to-server data-forwarding paths:

Path 1

The first packet from the router has a destination MAC address of the LocalDirector and is on VLAN 10. The MAC address has an Xtag value of 14 in the Layer 2 table. This value indicates that it requires a Layer 3 lookup, and the SYN flag is set so that the frame goes to port P_A.

In addition to forwarding the frame to port P_A, the switch hardware creates a "candidate" entry in the Layer 3 forwarding table. This entry is updated later by an "enabler" frame to become a full ASLB Multilayer Switching (MLS) entry.

Path 2

After receiving the frame from port P_A, the LocalDirector makes its standard load-balancing decision and forwards the frame to port P_B. The LocalDirector changes the destination MAC address to that of the appropriate server. When this frame enters the switch, it is considered an "enabler" frame. The switch hardware does a lookup in the Layer 3 table and searches for the entry that is created by the previous candidate packet (the packet that is forwarded through the LocalDirector). If the search was successful, a "hit" occurs in the Layer 3 table.

Path 3—N

The ASLB MLS entry has been created and the next and subsequent frames from the router with a destination MAC address of the LocalDirector MAC will be Layer 3 switched unless the packet has SYN, FIN, or RST flags set or the packet is fragmented.

Path N + 1, N + 2...

On the last frame of a connection, either the FIN or RST flags will be set in the TCP header causing the packet to go to the LocalDirector. The LocalDirector must then forward the frame back to the switch after modifying the destination MAC address to be that of the appropriate server. This redirected frame takes the same path as the first frame of the flow. The FIN packet is used by the LocalDirector to indicate that the connection with the server has been terminated, and by the ASLB to purge the affected ASLB MLS entry.

Figure 54-2 Client-to-Server ASLB Packet Flow

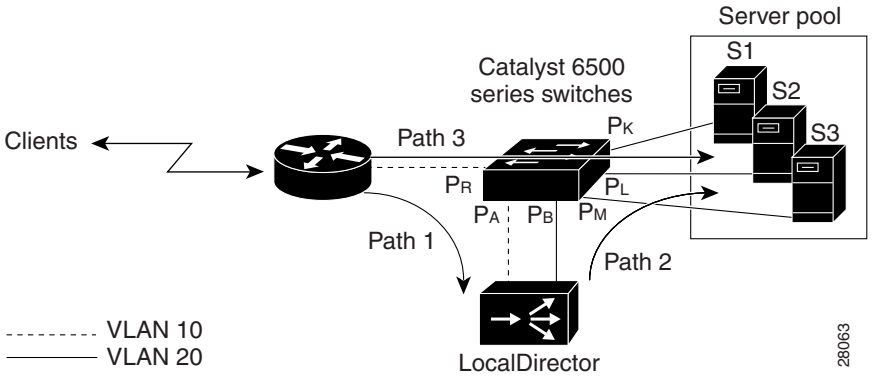


Table 54-2 Client-to-Server ASLB Packet Flow

Path Number	VLAN	MAC Destination Address	MAC Source Address	IP Destination Address	IP Source Address	Flags	Action
1	10	LocalDirector MAC[1]	Router MAC	VIP[2]	CIP[3]	SYN	Candidate entry in Layer 3 table
2	20	Server MAC[4]	Router MAC[1]	VIP	CIP	-	Enabler frame
3—N	10	LocalDirector MAC[1]	Router MAC	VIP	CIP	-	Full ASLB MLS entry created
N + 1	10	LocalDirector MAC[1]	Router MAC	VIP	CIP	FIN/RST	Path 1 redirect
N + 2...	20	Server MAC	Router MAC[1]	VIP	CIP	FIN/RST	Path 2

1. This MAC address has an Xtag value of 14 in the Layer 2 table for this packet's VLAN.
2. VIP = virtual-IP address.
3. CIP = client's IP address.
4. The MAC address of the server that the LocalDirector selected.

Table 54-3 Client-to-Server ASLB Layer 3 Table Entries

IP Destination Address	IP Source Address	Protocol	Ports	VLAN	MAC Destination Address	MAC Source Address
VIP[1]	CIP[2]	TCP	80/YZ	20	Server MAC[3]	Router MAC

1. VIP = virtual-IP address.
2. CIP = client's IP address.
3. MAC address of the server that the LocalDirector selected.

Server-to-Client Data Forwarding

Figure 54-3 shows how data is forwarded from the servers to the clients. Table 54-4 lists the sequence of events, and Table 54-5 lists the Layer 3 table entries.

The traffic from the servers to the router or client devices works in the same manner, but in the reverse direction, as described in the "Client-to-Server Data Forwarding" section on page 54-4. The exception is that the LocalDirector put its own MAC address as the source of the packet for all the packets that are going to the router. For the traffic in the client-to-server direction, the source MAC address of the packet was unmodified.

Figure 54-3 Server-to-Client ASLB Packet Flow

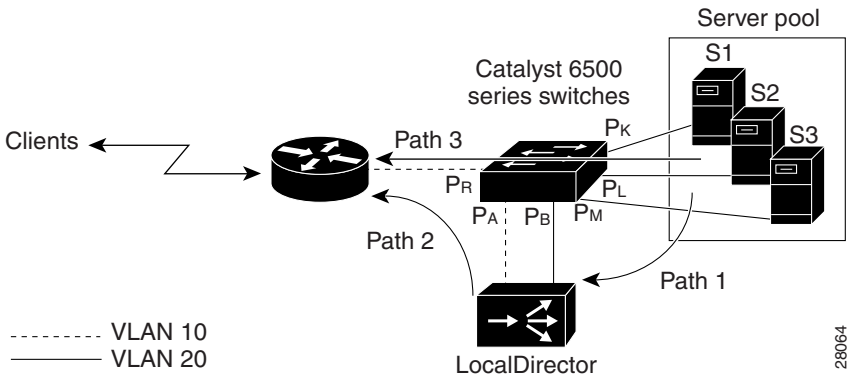


Table 54-4 Server-to-Client ASLB Packet Flow

Path Number	VLAN	MAC Destination Address	MAC Source Address	IP Destination Address	IP Source Address	Flags	Action
1	20	Router MAC[1]	Server MAC[2]	CIP[3]	VIP[4]	SYN	Candidate entry in Layer 3 table
2	10	Router MAC	LocalDirector MAC[1]	CIP	VIP	-	Enabler packet
3—N	20	Router MAC[1]	Server MAC	CIP	VIP	-	Full ASLB MLS entry created
N + 1	20	Router MAC[1]	Server MAC	CIP	VIP	FIN/RST	Path 1 redirect
N +2...	10	Router MAC	LocalDirector MAC[1]	CIP	VIP	FIN/RST	Path 2

1. This MAC address has an Xtag value of 14 in the Layer 2 table for this packet's VLAN.
2. The MAC address of the server that the LocalDirector selected.
3. CIP = client's IP address.
4. VIP = virtual-IP address.

Table 54-5 Server-to-Client ASLB Layer 3 Table Entries

IP Destination Address	IP Source Address	Protocol	Ports	VLAN	MAC Destination Address	MAC Source Address
VIP[1]	CIP[2]	TCP	80/YZ	20	Server MAC[3]	Router MAC
CIP	VIP	TCP	YZ/80	10	Router MAC	LocalDirector MAC

1. VIP = virtual-IP address.
2. CIP = client's IP address.
3. MAC address of the server that the LocalDirector selected.

Cabling Guidelines

This section describes the cabling guidelines for your ASLB configuration:

- Check that your connections to the servers are attached to the switch. The servers must be either directly attached to the switch or within the same bridging domain as the LocalDirector port in the server VLAN.
- Use two Category 5 unshielded twisted-pair cables to connect two 10/100 or two 1000BASE-X switch ports to two comparable LocalDirector interfaces.

Caution Connect the LocalDirector directly to the Catalyst 6500 series switch.

See the "Configuring the LocalDirector Interfaces" section on page 54-7 to configure the LocalDirector interfaces. See the "Configuring ASLB from the CLI" section on page 54-11 to configure the switch.

Configuring ASLB on the Switch

This section lists the tasks to configure ASLB:

Configuring the LocalDirector Interfaces

Refer to the *Cisco LocalDirector Installation and Configuration Guide,* Version 3.2, for detailed information on configuring the LocalDirector interfaces for ASLB.

ASLB Configuration Guidelines

This section lists the usage guidelines and restrictions for configuring ASLB:

For configuration examples, see the "ASLB Configuration Example" section on page 54-18. If you run into problems during your configuration, see the "Troubleshooting the ASLB Configuration" section on page 54-25.

Routers

The router configuration guidelines are as follows:

- The router must be the default gateway for the servers that are being load balanced, and its MAC address must be known.
- Multiple routers must be on the same router VLAN. Specify all the participating router MAC addresses by entering the **set lda mac router** command.
- When ASLB is configured, a VLAN access control list (VACL) is created to redirect the TCP traffic on the two VLANs to which the LocalDirector is connected; no security Cisco IOS access control lists (ACLs) or VACLs can be configured on these VLANs.

Servers

The server configuration guidelines are as follows:

- The servers must be either directly attached to the switch or within the same bridging domain as the LocalDirector port in the server VLAN.
- Configure the server default route as the aliased address of the router that is on the same subnet as the real IP address of the server.
- Configure the servers to ignore the ARP requests for the virtual-IP address. On some server operating systems, you cannot disable the responses to the ARP requests on the alias (secondary) IP addresses. Use the static ARP entries at the routers as a workaround for the servers that respond to the ARP requests for the virtual-IP address.

Caution To accelerate the client-to-server traffic, you must configure the servers to ignore the ARP requests for the virtual-IP address. If you fail to do this step, traffic acceleration does not start, and fully redundant topologies in your network take a long time to recover from a LocalDirector failure.

IP Addresses

The IP address configuration guidelines are as follows:

Note You can specify an IP address for the virtual-IP address other than the server IP network addresses.

- Ensure that the LocalDirectors and servers are on the same subnet to allow the LocalDirector to ARP the real IP address of each server.
- Ensure that the routers are on the same subnet as the virtual-IP address to allow the router to ARP the virtual-IP address.

Configure the network for ASLB as follows (the virtual-IP address in this example is 171.1.1.200):

Router	LocalDirector	Servers[1]
171.1.1.1	171.1.1.2	171.1.1.x

1. The default router on each server is 171.1.1.1.

If the servers in your ASLB configuration need to follow RFC 1918 for privacy, use the following as a guideline (the virtual-IP address in this example is 171.1.1.200):

Routers	LocalDirector	Servers[1]
171.1.1.1	171.1.1.2	10.1.1.x (real IP address)
Alias 10.1.1.1	Alias 10.1.1.2	Loopback alias to 171.1.1.200

1. The default router on each server is 10.1.1.1.

Supervisor Engine

The supervisor engine configuration guidelines are as follows:

- Up to 32 router MAC addresses are supported.
- Up to 1024 virtual-IP/TCP port pairs are supported.

Backup LocalDirector Configuration (Optional)

Connect the ports on the backup LocalDirector to the switch and specify the server and router configuration by entering the **set lda server** and **set lda router** commands. Connect the active and backup LocalDirectors to their specified ports or the ASLB feature will not work.

MSFC and Multilayer Switching

The MSFC and Multilayer Switching (MLS) configuration guidelines are as follows:

- With supervisor engine software release 5.4(1)CSX or later releases, an MSFC can be the participating router for ASLB.

  Note Traffic is Layer 3 switched when an MSFC routes the traffic from the clients. This process creates the MLS entries that exist separately from the ASLB MLS entries for the same traffic.

- The aging task that removes the terminated ASLB flows also purges the MLS terminated flows. The ASLB MLS entries share the Layer 3 MLS cache with the MLS shortcut entries.

  The MLS commands (**set mls**, **clear mls**, and **show mls**) do not interoperate with the ASLB (**set lda**, **clear lda**, **show lda**, and **commit lda**) commands. ASLB uses separate commands to view the LocalDirector MLS entries.

- When you enable ASLB, the ASLB MLS entries are established using one flow mask, full-flow mode (ip-flow).

NetFlow Data Export

You cannot use NetFlow Data Export (NDE) if you enable ASLB, and you cannot use ASLB if you enable NDE.

VLANs

The VLAN configuration guidelines are as follows:

- When you configure ASLB, a VACL is created to redirect the TCP traffic on the two VLANs to which the LocalDirector is connected (router VLAN and server VLAN). You cannot configure any security Cisco IOS ACLs or VACLs on these VLANs.
- Dedicate the router VLAN and server VLAN for ASLB use only. Do not connect the other network devices (such as end stations and clients) to these two VLANs.
- The VLANs that are created for ASLB propagate to the other switches through VLAN Trunking Protocol (VTP) when VTP is in the server mode. Spanning Tree Protocol runs over these ASLB VLANs on all VTP switches in the network, introducing additional overhead over the entire network. To avoid the spanning-tree propagation delays, do the following:
  - Configure the switch as VTP transparent so it does not populate the VLANs.
  - Remove the ASLB VLANs from all trunks on all switches (enter the **clear trunk** command).

Switch Port Configuration

The switch port configuration guidelines are as follows:

- Disable CDP on the ports that are connected to the LocalDirectors (both active and standby LocalDirectors if a backup is configured).
- If you specify a port that is part of an EtherChannel, the traffic is automatically redirected among all ports in the EtherChannel.

Configuring ASLB from the CLI

This section describes how to configure ASLB using the Catalyst 6500 series switch **lda** command set:

Configuring the Switch Ports Connected to the LocalDirector

To configure the 10/100-Ethernet switch ports that are connected to the LocalDirector, perform these steps:

Step 1 Enter the **set vlan** *vlan_num mod_ports* command to add the switch ports to the correct VLANs (router VLAN and server VLAN).

Step 2 Note that the port speed and duplex type for the switch ports do not need to be set as all 10/100-switch ports are set to autonegotiate as the default. If you have a problem with autonegotiation, configure the port speed and duplex type as follows:

Enter the **set port speed** *mod/port* {**10** | **100** | **auto**} command to set the port speed.

Enter the **set port duplex** *mod/port* {**full** | **half** | **auto**} command to set the type of duplex.

Enabling and Disabling ASLB

Note ASLB is disabled by default. When ASLB is disabled, you cannot enter the **set lda** commands to perform configuration tasks; to enter the **set lda** commands, you must enable ASLB.

To enable or disable ASLB, perform this task in privileged mode:

Task	Command
Enable or disable ASLB.	**set lda enable** | **disable**

This example shows how to enable ASLB on the switch:

```
Console> (enable) set lda enable
Successfully enabled Local Director Accelerator.
Console> (enable)
```

This example shows how to disable ASLB on the switch:

```
Console> (enable) set lda disable
Successfully disabled Local Director Accelerator.
Console> (enable)
```

Specifying the Server Virtual-IP Addresses and TCP Ports for Acceleration

Note You can specify up to 1024 virtual-IP addresses and TCP port pairs for acceleration by the Catalyst 6500 series switch. Newly specified pairs do not replace the previously specified pairs. To cancel a previously entered pair, enter the **clear lda vip** command.

Note You can use a zero (0) as a wildcard (don't care) digit for the *destination_tcp_port*.

To specify the server virtual-IP addresses and TCP ports for acceleration, perform this task in privileged mode:

Task	Command
Specify the server virtual-IP addresses and TCP ports for acceleration.	**set lda vip** {*server_virtual_ip*} {*destination_tcp_port*} [{*server_virtual_ip*} {*destination_tcp_port*}...]

This example shows how to specify a server virtual-IP address and TCP port for acceleration:

```
Console> (enable) set lda vip 10.0.0.8 8
Successfully set server virtual ip and port information.
Use commit lda command to save settings to hardware.
Console> (enable)
```

Specifying the MAC Addresses for Participating Routers

Note You can specify up to 32 router MAC addresses.

To specify the MAC addresses for the participating routers, perform this task in privileged mode:

Task	Command
Specify the MAC addresses for the participating routers.	**set lda mac router** {*mac-address*}...

This example shows how to specify the MAC addresses for the participating routers:

```
Console> (enable) set lda mac router 00-23-45-67-ee-7f
Successfully set mac address.
Use commit lda command to save settings to hardware.
Console> (enable)
```

Specifying a MAC Address for the LocalDirector

To specify a MAC address for the LocalDirector, perform this task in privileged mode:

:

Task	Command
Specify a MAC address for the LocalDirector.	**set lda mac ld** {*ld_mac-address*}

This example shows how to specify a MAC address for the LocalDirector:

```
Console> (enable) set lda mac ld 00-11-22-33-55-66
Successfully set mac address.
Use commit lda command to save settings to hardware.
Console> (enable)
```

Specifying the Router VLAN and the LocalDirector Port on the VLAN

Note After entering the **set lda router** command, if you change the switch port(s) to which the LocalDirector is connected, you must enter the **set lda router** command again to specify the new configuration.

Note Specifying a backup LocalDirector port is optional unless you are setting up a failover configuration of LocalDirectors. If you are setting up a failover configuration, you must specify the ports for the backup LocalDirector. If this is not done, failover does not work because the supervisor engine does not send any traffic to the intended backup LocalDirector.

To specify the router VLAN and the LocalDirector port on the VLAN, perform this task in privileged mode:

Task	Command
Specify the router VLAN and the LocalDirector port on the VLAN.	**set lda router** {*router_vlan*} {*ld_mod/port*} [*backup_ld_mod/port*]

This example shows how to specify the router VLAN and the LocalDirector port on the VLAN:

```
Console> (enable) set lda router 110 4/26
Successfully set router vlan and LD port.
Use commit lda command to save settings to hardware.
Console> (enable)
```

Specifying the Server VLAN and the LocalDirector Port on the VLAN

Note After entering the **set lda server** command, if you change the switch port(s) to which the LocalDirector is connected, you must enter the **set lda server** command again to specify the new configuration.

Note Specifying a backup LocalDirector port is optional unless you are setting up a failover configuration of LocalDirectors. If you are setting up a failover configuration, you must specify the ports for the backup LocalDirector. If this is not done, failover does not work because the supervisor engine does not send any traffic to the intended backup LocalDirector.

To specify the server VLAN and the LocalDirector port on the VLAN, perform this task in privileged mode:

Task	Command
Specify the server VLAN and the LocalDirector port on the VLAN.	**set lda server** {*server_vlan*} {*ld_mod/port*} [*backup_ld_mod/port*]

This example shows how to specify the server VLAN and the LocalDirector port on the VLAN:

```
Console> (enable) set lda server 105 4/40
Successfully set server vlan and LD port.
Use commit lda command to save settings to hardware.
Console> (enable)
```

Configuring the UDP Aging

To configure the User Datagram Protocol (UDP) aging, perform this task in privileged mode:

Task	Command
Configure the UDP aging.	**set lda udpage** *time_in_ms*

You can set the aging from 1–2024000 milliseconds (ms). Enter a value of zero to disable UDP aging.

This example shows how to configure the UDP aging to 500 ms:

```
Console> (enable) set lda udpage 500
Successfully set LDA UDP aging time to 500ms.
Console> (enable)
```

Committing the ASLB Configuration

Note The ASLB configuration settings are temporarily stored in an edit buffer. The settings are saved in NVRAM, but for the settings to take effect, you must enter the **commit lda** command. This command verifies your configuration settings. If the information is entered correctly and passes a consistency check, the settings are programmed into the hardware. Once the ASLB configuration is successfully committed, the mapping is saved in NVRAM and restored at the system bootup.

To commit your ASLB configuration settings, perform this task in privileged mode:

Task	Command
Commit your ASLB configuration settings.	**commit lda**

This example shows how to commit the ASLB configuration settings:

```
Console> (enable) commit lda
Commit operation in progress...
Successfully committed Local Director Accelerator.
Console> (enable)
```

Displaying the ASLB Configuration

Note Entering the **show lda** command without a keyword (**committed** | **uncommitted**) displays the committed configuration settings.

To display the committed or uncommitted ASLB configuration settings, perform this task in privileged mode:

Task	Command
Display the committed or uncommitted ASLB configuration settings.	**show lda** [**committed**	**uncommitted**]

This example shows how to display the committed ASLB configuration settings:

```
Console> (enable) show lda committed
Status:Committed

Virtual IP addresses:
Local Director Flow:10.0.0.8/ (TCP port 8)

Router MAC:
00-23-45-67-ee-7f


LD MAC:00-11-22-33-55-66

LD Router Side:
---------------
Router and LD are on VLAN 110
LD is connected to switch port 4/26 on VLAN 110

LD Server Side:
---------------
Server(s) and LD are on VLAN 105
LD is connected to switch port 4/40 on VLAN 105
Console> (enable)
```

If the configuration is modified and the changes are not committed, entering the **show lda** command again gives an indication that the configuration has been modified since the last commit, but the new modifications are not shown, only the committed modifications are displayed. To view the new modifications, enter the **show lda uncommitted** command.

Displaying the ASLB MLS Entries

Note The **short | long** keyword options give the flexibility to display the output in regular (80 characters in width) or wide-screen format.

To display the ASLB MLS entries, perform this task in privileged mode:

Task	Command
Display the ASLB MLS entries.	**show lda mls entry** **show lda mls entry** [**destination** *ip_addr_spec*] [**source** *ip_addr_spec*] [**protocol** *protocol*] [**src-port** *port*] [**dst-port** *port*] [**short | long**]

This example shows how to display all the ASLB MLS entries in short format:

```
Console> (enable) show lda mls entry short
Destination-IP  Source-IP       Prot  DstPrt SrcPrt Destination-Mac   Vlan
--------------- --------------- ----- ------ ------ ----------------- ----
EDst ESrc DPort  SPort  Stat-Pkts  Stat-Bytes  Uptime   Age
---- ---- ------ ------ ---------- ----------- -------- --------
10.0.0.8        172.20.20.10    TCP  8      64     00-33-66-99-22-44 105
ARPA ARPA -      4/25   0          0           00:00:02 00:00:05

10.0.0.8        172.20.20.11    TCP  8      64     00-33-66-99-22-44 105
ARPA ARPA -      4/25   0          0           00:00:05 00:00:08
Console> (enable)
```

This example shows how to display the ASLB information for the source IP address in short format:

```
Console> (enable) show lda mls entry source 172.20.20.11 short
Destination-IP  Source-IP       Prot  DstPrt SrcPrt Destination-Mac   Vlan
--------------- --------------- ----- ------ ------ ----------------- ----
EDst ESrc DPort  SPort  Stat-Pkts  Stat-Bytes  Uptime   Age
---- ---- ------ ------ ---------- ----------- -------- --------
10.0.0.8        172.20.20.11    TCP  8      64     00-33-66-99-22-44 105
ARPA ARPA -      4/25   0          0           00:00:05 00:00:08
Console> (enable)
```

Displaying the ASLB MLS Statistics

To display the ASLB MLS statistics, perform this task in privileged mode:

Task	Command
Display the ASLB MLS entry statistics.	**show lda mls statistics entry** **show lda mls statistics count** **show lda mls statistics entry** [**destination** *ip_addr_spec*] [**source** *ip_addr_spec*] [**protocol** *protocol*] [**src-port** *port*] [**dst-port** *port*]

This example shows how to display all the ASLB MLS entry statistics:

```
Console> (enable) show lda mls statistics entry
                                  Last    Used
Destination IP  Source IP       Prot DstPrt SrcPrt Stat-Pkts  Stat-Bytes
--------------- --------------- ---- ------ ------ ---------- ---------------
10.0.0.8        172.20.20.10    TCP  WWW    64     636        29256
10.0.0.8        172.20.22.10    TCP  WWW    64     0          0
Console> (enable)
```

This example shows how to display the number of ASLB active MLS entries:

```
Console> (enable) show lda mls statistics count
LDA active shortcuts: 20
Console> (enable)
```

This example shows how to display the statistics for a specific destination IP address:

```
Console> (enable) show lda mls statistics entry destination 172.20.22.14
                Last Used         Last    Used
Destination IP  Source IP       Prot DstPrt SrcPrt Stat-Pkts  Stat-Bytes
--------------- --------------- ---- ------ ------ ---------- ---------------
172.20.22.14    172.20.25.10    6    50648  80     3152       347854
Console> (enable)
```

Clearing the ASLB Configuration

Caution If you do not enter any keywords with the **clear lda** command, the *entire* ASLB configuration (including the MLS entries) is removed from the hardware and NVRAM. If you do not enter any keywords with the **clear lda mls** command, all the MLS entries are cleared.

To clear the ASLB entries or router MAC addresses, perform this task in privileged mode:

Task	Command
Clear the ASLB configuration settings.	**clear lda mls** **clear lda mls** [**destination** *ip_addr_spec*] [**source** *ip_addr_spec*] [**protocol** *protocol* **src-port** *src_port* **dst-port** *dst_port*] **clear lda vip** {**all** | *vip* | *vip tcp_port*} **clear lda mac** {**all** | *router_mac_address*}

This example shows how to clear the MLS entry at a specific destination address:

```
Console> (enable) clear lda mls destination 172.20.26.22
MLS IP entry cleared.
Console> (enable)
```

This example shows how to delete a virtual-IP address and port pair (10.0.0.8, port 8):

```
Console> (enable) clear lda vip 10.0.0.8 8
Successfully deleted vip/port pairs.
Console> (enable)
```

This example shows how to clear all the ASLB router MAC addresses:

```
Console> (enable) clear lda mac all
Successfully cleared Router MAC address.
Console> (enable)
```

This example shows how to clear a specific ASLB router MAC address:

```
Console> (enable) clear lda mac 1-2-3-4-5-6
Successfully cleared Router MAC address.
Console> (enable)
```

ASLB Configuration Example

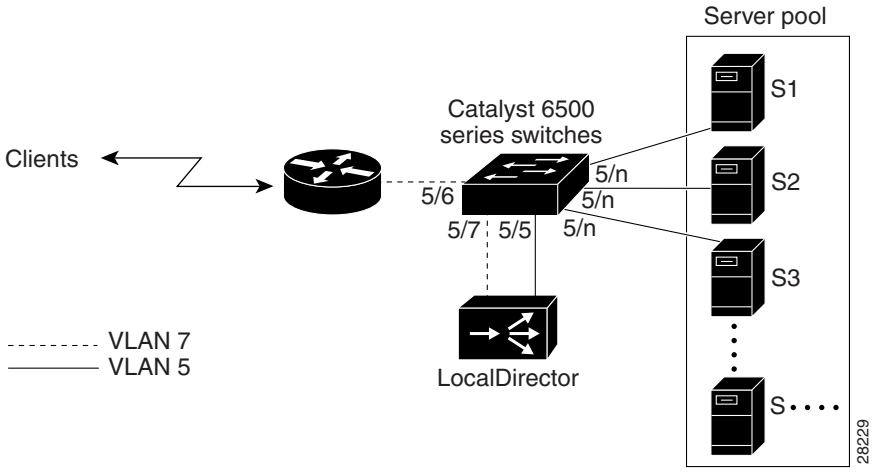
This section provides an example of a typical ASLB network configuration. Figure 54-4 shows the example network. The configuration specifications are as follows:

- The virtual-IP address is 192.255.201.55.
- The router interface MAC address is 00-d0-bc-e9-fb-47, and its IP address is 192.255.201.1.
- The LocalDirector IP address is 192.255.201.2.
- The LocalDirector MAC address is 00-e0-b6-00-4b-04.
- The server farm IP addresses are 192.255.201.3 through 192.255.201.11.
- The servers have been configured to ignore the ARP requests for the virtual-IP address 192.255.201.55.

The example in Figure 54-4 shows how to do the following:

- Load balance the HTTP connections in a round-robin sequence among servers 192.255.201.3 through 192.255.201.10.
- Forward the connections to port 8001 to server 192.255.201.11.
- Load balance the FTP connections to servers 192.255.201.3 through 192.255.201.8 in a "leastconns" sequence (which is the default for the LocalDirector).

Figure 54-4 ASLB Configuration Example



The router configuration is as follows (MSM is used in this example):

```
!
interface Port-channel1.7
encapsulation isl 7
ip address 192.255.201.1 255.255.255.0
no ip redirects
no ip directed-broadcast
!
```

The Catalyst 6500 series switch configuration is as follows:

```
Console (enable) show lda
Status:Committed

Virtual IP addresses:
Local Director Flow:192.255.201.55/www (TCP port 80)
Local Director Flow:192.255.201.55/ (TCP port 8001)
Local Director Flow:192.255.201.55/ftp (TCP port 21)

Router MAC:
00-d0-bc-e9-fb-47

LD MAC:  00-e0-b6-00-4b-04

LD Router Side:
---------------
Router and LD are on VLAN 7
LD is connected to switch port 5/7 on VLAN 7
```

```
LD Server Side:
---------------
Server(s) and LD are on VLAN 5
LD is connected to switch port 5/5 on VLAN 5
Console (enable)
```

The LocalDirector configuration is as follows:

```
LD430# show configuration
:Saved
:LocalDirector 430 Version 3.1.3.105
syslog output 20.3
no syslog console
hostname LD430
no shutdown ethernet 0
no shutdown ethernet 1
shutdown ethernet 2
shutdown ethernet 3
interface ethernet 0 100full
interface ethernet 1 100full
interface ethernet 2 auto
interface ethernet 3 auto
mtu 0 1500
mtu 1 1500
mtu 2 1500
mtu 3 1500
no multiring all
no secure  0
no secure  1
no secure  2
no secure  3
ping-allow 0
ping-allow 1
no ping-allow 2
no ping-allow 3

ip address 192.255.201.2 255.255.255.0
route 0.0.0.0 0.0.0.0 192.255.201.1 1
no rip passive
rip version 1
failover ip address 0.0.0.0
no failover
snmp-server enable traps
no snmp-server contact
no snmp-server location
virtual 192.255.201.55:80:0:tcp is
virtual 192.255.201.55:8001:0:tcp is
virtual 192.255.201.55:21:0:tcp is
predictor 192.255.201.55:80:0:tcp roundrobin
redirection 192.255.201.55:80:0:tcp dispatched assisted wildcard-ttl 60
fixed-ttl 60 igmp 224.0.1.2 port 1637
redirection 192.255.201.55:8001:0:tcp dispatched assisted wildcard-ttl 60
fixed-ttl 60 igmp 224.0.1.2 port 1637
redirection 192.255.201.55:21:0:tcp dispatched assisted wildcard-ttl 60
fixed-ttl 60 igmp 224.0.1.2 port 1637
real 192.255.201.5:80:0:tcp is
real 192.255.201.3:80:0:tcp is
real 192.255.201.4:80:0:tcp is
real 192.255.201.6:80:0:tcp is
real 192.255.201.7:80:0:tcp is
real 192.255.201.8:80:0:tcp is
real 192.255.201.9:80:0:tcp oos
real 192.255.201.10:80:0:tcp oos
real 192.255.201.11:8001:0:tcp oos
```

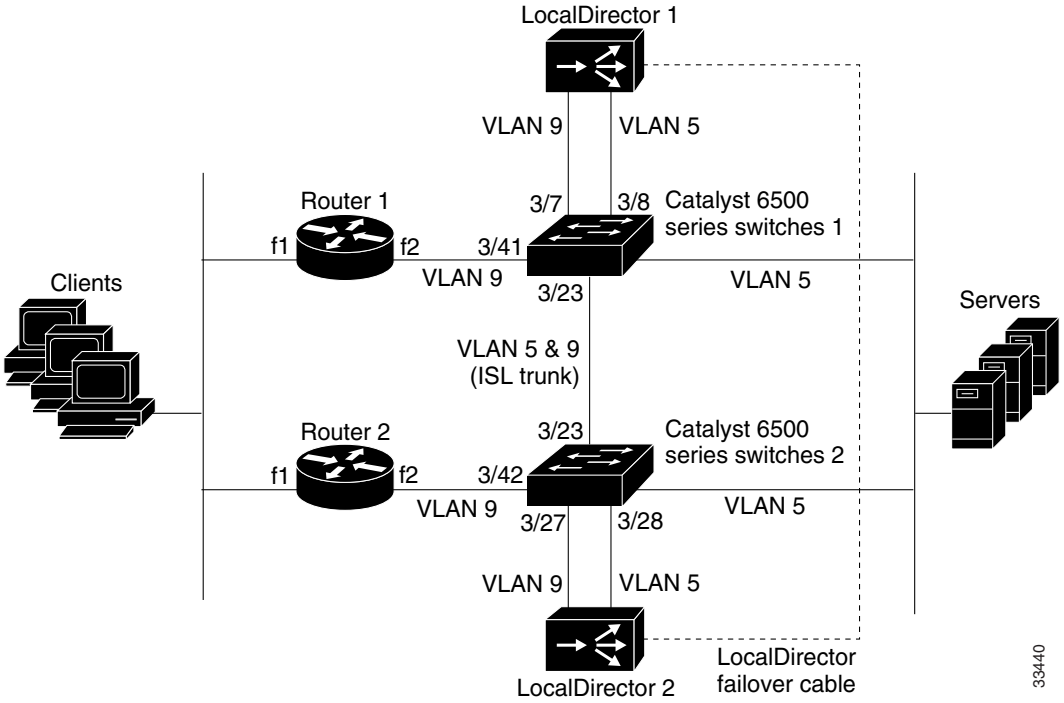
```
real 192.255.201.3:21:0:tcp is
real 192.255.201.4:21:0:tcp is
real 192.255.201.5:21:0:tcp is
real 192.255.201.6:21:0:tcp is
real 192.255.201.7:21:0:tcp is
real 192.255.201.8:21:0:tcp is
bind 192.255.201.55:80:0:tcp 192.255.201.3:80:0:tcp
bind 192.255.201.55:80:0:tcp 192.255.201.4:80:0:tcp
bind 192.255.201.55:80:0:tcp 192.255.201.5:80:0:tcp
bind 192.255.201.55:80:0:tcp 192.255.201.6:80:0:tcp
bind 192.255.201.55:80:0:tcp 192.255.201.7:80:0:tcp
bind 192.255.201.55:80:0:tcp 192.255.201.8:80:0:tcp
bind 192.255.201.55:80:0:tcp 192.255.201.9:80:0:tcp
bind 192.255.201.55:80:0:tcp 192.255.201.10:80:0:tcp
bind 192.255.201.55:8001:0:tcp 192.255.201.11:8001:0:tcp
bind 192.255.201.55:21:0:tcp 192.255.201.3:21:0:tcp
bind 192.255.201.55:21:0:tcp 192.255.201.4:21:0:tcp
bind 192.255.201.55:21:0:tcp 192.255.201.5:21:0:tcp
bind 192.255.201.55:21:0:tcp 192.255.201.6:21:0:tcp
bind 192.255.201.55:21:0:tcp 192.255.201.7:21:0:tcp
bind 192.255.201.55:21:0:tcp 192.255.201.8:21:0:tcp
```

ASLB Redundant Configuration Example

This section provides an example of a typical ASLB redundant network configuration. Figure 54-5 shows the example redundant network. The LocalDirectors and Catalyst 6500 series switches are configured to accelerate HTTP and Telnet for server VIP address 13.13.13.13.

Caution Router 1 and router 2 are running Hot Standby Router Protocol (HSRP) on both interfaces, f1 and f2, in Figure 54-5. Interface f1 must be active on the same router where f2 is active; otherwise, the traffic reaches interface f1 on one router and is not forwarded to interface f2 which is active on the other router. Use the HSRP **track** command to track the opposite side interface of each router.

Figure 54-5 ASLB Redundant Configuration Example



IP Addresses

The IP addresses are as follows:

- Router 1, f1 IP address: 7.0.0.100 (network 7)
- Router 2, f1 IP address: 7.0.0.101 (network 7)
- HSRP IP address: 7.0.0.1 for network 7
- Router 1, f2 IP address: 5.0.0.100 (network 5)
- Router 2, f2 IP address: 5.0.0.101 (network 5)
- HSRP IP address: 5.0.0.2 for network 5
- LocalDirector IP address: 5.0.0.1
- Server IP address: 5.100.100.100
- VIP address for servers: 13.13.13.13

MAC Addresses

The MAC addresses are as follows:

- HSRP MAC address for network 7: 00-00-0c-07-ac-00
- HSRP MAC address for network 5: 00-00-0c-07-ac-01
- Router 1, f2 MAC address: 00-d0-79-7b-20-88
- Router 2, f2 MAC address: 00-d0-79-7b-18-88
- LocalDirector MAC address: 00-e0-b6-00-47-ec

Catalyst 6500 Series Switch 1 Configuration

The switch 1 configuration is as follows:

```
set trunk 3/23 on isl 1,5,9
set lda enable
clear lda vip all
set lda vip 13.13.13.13 80 13.13.13.13 23
clear lda mac all
set lda mac router 00-00-0c-07-ac-01
set lda mac router 00-d0-79-7b-20-88
set lda mac router 00-d0-79-7b-18-88
set lda mac ld 00-e0-b6-00-47-ec
set lda router 9 3/7 3/23
set lda server 5 3/8 3/23
commit lda
```

Catalyst 6500 Series Switch 2 Configuration

The switch 2 configuration is as follows:

```
set trunk 3/23 on isl 1,5,9
set lda enable
clear lda vip all
set lda vip 13.13.13.13 80 13.13.13.13 23
clear lda mac all
set lda mac router 00-00-0c-07-ac-01
set lda mac router 00-d0-79-7b-20-88
set lda mac router 00-d0-79-7b-18-88
set lda mac ld 00-e0-b6-00-47-ec
set lda router 9 3/27 3/23
set lda server 5 3/28 3/23
commit lda
```

Router 1 Configuration

The router 1 configuration is as follows:

```
interface FastEthernet1
 ip address 7.0.0.100 255.0.0.0
 no ip redirects
 no ip directed-broadcast
 no ip route-cache distributed
 load-interval 30
 no keepalive
```

```
 full-duplex
 standby 1 ip 7.0.0.1
 standby 1 track FastEthernet2
!
interface FastEthernet2
 ip address 5.0.0.100 255.0.0.0
 no ip redirects
 no ip directed-broadcast
 no ip route-cache distributed
 no keepalive
 full-duplex
 standby priority 250
 standby 2 ip 5.0.0.2
 standby 2 track FastEthernet1
!
ip route 13.13.13.13 255.255.255.255 5.0.0.1
```

Router 2 Configuration

The router 2 configuration is as follows:

```
interface FastEthernet1
 ip address 7.0.0.101 255.0.0.0
 no ip redirects
 no ip directed-broadcast
 no ip route-cache distributed
 load-interval 30
 no keepalive
 full-duplex
 standby 1 ip 7.0.0.1
 standby 1 track FastEthernet2
!
interface FastEthernet2
 ip address 5.0.0.101 255.0.0.0
 no ip redirects
 no ip directed-broadcast
 no ip route-cache distributed
 no keepalive
 full-duplex
 standby priority 250
 standby 2 ip 5.0.0.2
 standby 2 track FastEthernet1
!
ip route 13.13.13.13 255.255.255.255 5.0.0.1
```

LocalDirector Configuration

The LocalDirector 1 and LocalDirector 2 configuration is as follows (the configuration is the same for both LocalDirectors):

```
no shutdown ethernet 0
no shutdown ethernet 4
interface ethernet 0 100full
interface ethernet 4 100full
ip address 5.0.0.1 255.0.0.0
failover ip address 5.0.0.5
virtual 13.13.13.13:80:0:tcp is
virtual 13.13.13.13:23:0:tcp is
predictor 13.13.13.13:80:0:tcp roundrobin
predictor 13.13.13.13:23:0:tcp roundrobin
```

```
redirection 13.13.13.13:80:0:tcp dispatched assisted
redirection 13.13.13.13:23:0:tcp dispatched assisted
real 5.100.100.100:80:0:tcp is
real 5.100.100.100:23:0:tcp is
bind 13.13.13.13:80:0:tcp 5.100.100.100:80:0:tcp
bind 13.13.13.13:23:0:tcp 5.100.100.100:23:0:tcp
```

Troubleshooting the ASLB Configuration

Table 54-6 lists the possible problem symptoms and recommended actions to troubleshoot the ASLB configuration.

Table 54-6 Troubleshooting the ASLB Configuration

Symptom	Recommended Action
LocalDirector does not receive any traffic.	Ensure that the LocalDirector is connected to the ports that you specified by entering the **set lda server** and **set lda router** commands.
LocalDirector connection entries are not purged.	Ensure that you configured all the virtual-IP/port pairs by entering the **set lda vip** command.
ASLB MLS entries are created in only one direction.	Ensure that you configured all the virtual-IP/port pairs on both the supervisor engine (**set lda vip** command) and the LocalDirector. Ensure that the LocalDirector is in the "dispatched assisted" mode. Ensure that you configured the IP addresses of the routers, LocalDirector, and servers following the guidelines in the "IP Addresses" section on page 54-9. Ensure that the router knows how to reach the LocalDirector when the traffic goes to the virtual-IP address (if the virtual-IP address is on a different subnet than the router interface). Ensure that the router MAC address is the same as specified by entering the **set lda mac router** command. Ensure that the LocalDirector MAC address is the same as specified by entering the **set lda mac ld** command.
Backup LocalDirector does not receive any traffic.	Ensure that you configured the backup LocalDirector ports by entering the **set lda router** and **set lda server** commands; for example, enter **set lda router** {*router_vlan*} **3/7 3/9** and **set lda server** {*server_vlan*} **3/8 3/10**.
You can ping the servers from the router, but the ASLB MLS entries are not created when you send the data traffic.	Ensure that the servers were configured to ignore ARP requests for the virtual-IP address.
You see the message: %CDP-4-NVLANMISMATCH: Native vlan mismatch detected on port ...	Disable CDP[1] on the ports that are connected to the LocalDirector (enter the **set cdp disable** command).

Table 54-6 Troubleshooting the ASLB Configuration (continued)

Symptom	Recommended Action
LocalDirector **set** commands did not take effect.	The **set lda** commands do not take effect until you enter the **commit lda** command. You can verify which **set lda** commands are in effect by entering the **show lda commit** command. You can determine which **set lda** commands are set but not committed or what changes will occur if the current **set lda** commands are committed by entering the **show lda uncommitted** command.
You see “collisions” or “port disabled” on the Catalyst 6500 series switch port.	Ensure that the port speed and duplex settings are compatible on both ends of the link between the LocalDirector and the switch. For example, if port 3/7 on the switch is connected to interface ethernet 0 on the LocalDirector, make sure that port 3/7 is set to 100full and that interface ethernet 0 on the LocalDirector is also set to 100full.

1. CDP = Cisco Discovery Protocol

CHAPTER 55

Configuring a VoIP Network

This chapter describes how to configure a Voice-over-IP (VoIP) network on the Catalyst 6500 series switches.

Note While this chapter introduces a number of Cisco networking products that are related to VoIP, the primary focus of the chapter is to provide configuration information for integrating the Catalyst 6500 series products into your VoIP network.

Note For complete syntax and usage information for the commands that are used in this chapter, refer to the *Catalyst 6500 Series Switch Command Reference* publication.

This chapter consists of these sections:

Hardware and Software Requirements

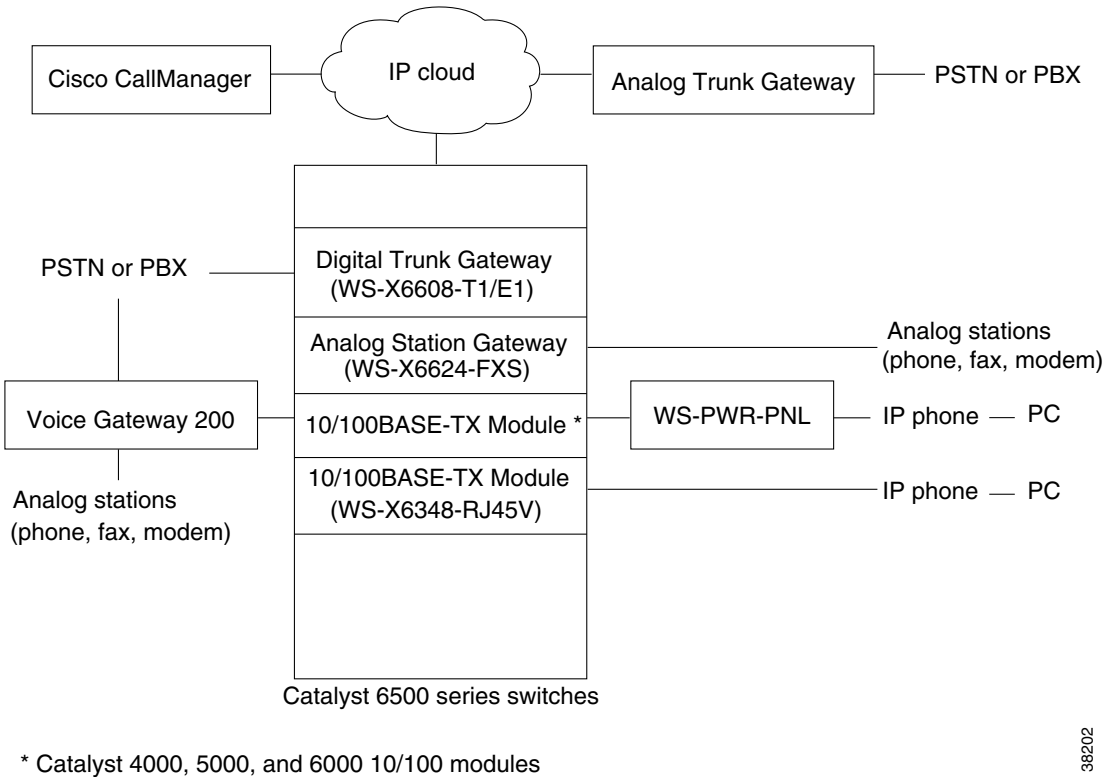
The hardware and software requirements for the Catalyst 6500 series switches and Cisco CallManager are as follows:

- Catalyst 4500 series, 5000 family, and Catalyst 6500 series switches running supervisor engine software release 6.1(1) or later releases
- Catalyst 4500 series and Catalyst 6500 series switches running supervisor engine software release 8.2(1) or later releases for IEEE 802.3af compliance
- Cisco CallManager release 3.0 or later releases

Understanding How a VoIP Network Works

A telephony system built on an IP network instead of the traditional circuit-switched private branch exchange (PBX) network is called an IP PBX system. (See Figure 55-1.) The system's components are described in these sections:

Figure 55-1 IP PBX System



* Catalyst 4000, 5000, and 6000 10/100 modules

Cisco IP Phone 7960

The Cisco IP Phone 7960 provides the connectivity to the IP PBX system. The IP phone has two RJ-45 jacks for connecting to the external devices: a LAN-to-phone jack and a PC-to-phone jack. The jacks use either Category 3 or Category 5 unshielded twisted-pair (UTP) cable. The LAN-to-phone jack is used to connect the phone to the LAN using a crossover cable; a workstation or a PC can be connected to the PC-to-phone jack using a straight-through cable.

The inline power is designed to work in cables from Category 3, Category 4, Category 5, and later up to 100 meters. The inline power works with IBM Token Ring STP cable of 100 meters when used with a Token Ring to Fast Ethernet adapter (LanTel Silver Bullet SB-LN/VIP-DATA adapter).

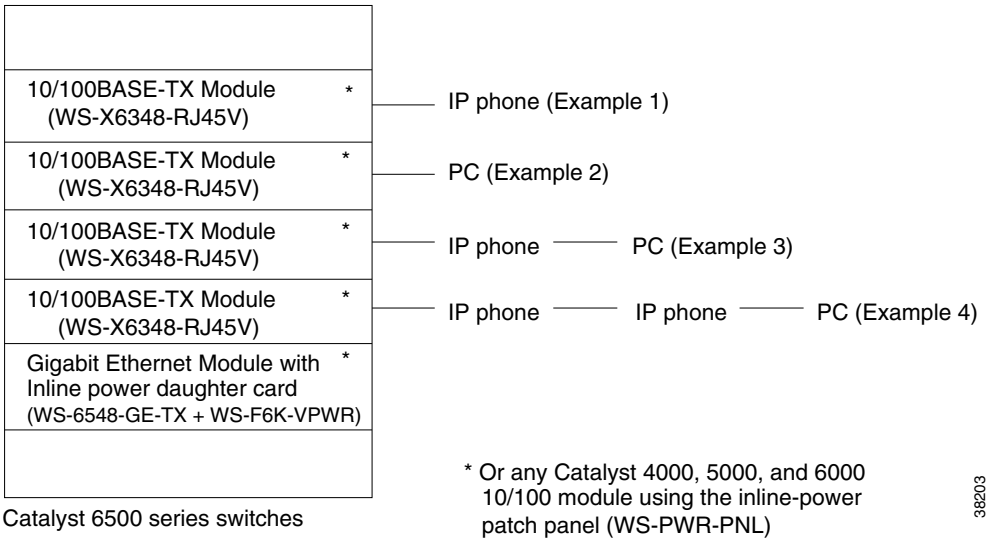
The IP phone is Dynamic Host Configuration Protocol (DHCP) capable. Optionally, you can program the IP phone with a static IP address.

The IP phone can be powered by the following sources:

- External power source—Optional transformer and power cord for connecting to a standard wall receptacle.
- Ethernet switching modules with the voice daughter card installed—Provides the inline power to the IP phone.
- WS-PWR-PNL (inline-power patch panel)—Provides the inline power to the IP phone. The inline patch panel allows the IP phone to connect to existing Catalyst 4500 series, 5000 family, and 6500 series 10/100BASE-TX switching modules.
- WS-PWR-PNL (inline-power patch panel)—Provides the inline power to the IP phone. The inline patch panel allows the IP phone to connect to existing Catalyst 4500 series, 5000 family, and 6500 series 10/100BASE-TX switching modules.
- WS-X6148-RJ-45 10/100 switching module with either the WS-F6K-VPWR inline-power field-upgrade module or the WS-F6K-FE48-AF inline-power field-upgrade module—Provides the inline power to the IP phone.
- WS-X6148-RJ-21 10/100 switching module with either the WS-F6K-VPWR inline-power field-upgrade module or the WS-F6K-FE48-AF inline-power field-upgrade module—Provides the inline power to the IP phone.
- WS-X6148X2-RJ-45 10/100 switching module with the WS-F6K-FE96-AF inline-power field-upgrade module—Provides the inline power to the IP phone.
- WS-X6148X2-RJ-21 10/100 switching module with the WS-F6K-FE96-AF inline-power field-upgrade module—Provides the inline power to the IP phone.
- WS-6548-GE-TX Gigabit Ethernet switching module with either the WS-F6K-VPWR-GE inline-power field-upgrade module or the WS-F6K-GE48-AF inline-power field-upgrade module—Provides the inline power to the IP phone.
- WS-6148-GE-TX Gigabit Ethernet switching module with either the WS-F6K-VPWR-GE inline-power field-upgrade module or the WS-F6K-GE48-AF inline-power field-upgrade module—Provides the inline power to the IP phone.

Figure 55-2 shows how to connect the Cisco IP Phone 7960 and PCs to the Catalyst 6500 series switch.

Figure 55-2 Connecting the Cisco IP Phone 7960 to the Catalyst 6500 Series Switch



The examples shown in Figure 55-2 are described in detail as follows:

- Example 1: Single Cisco IP Phone 7960

  Example 1 shows one IP phone that is connected to the 10/100 port on the Catalyst 6500 series switch. The PC-to-phone jack on the phone is not used. The phone can be powered through the 10/100 port or wall powered.

- Example 2: Single PC

  Example 2 shows one PC that is connected to the 10/100 port on the Catalyst 6500 series switch. The PC is wall powered.

- Example 3: One Cisco IP Phone 7960 and One PC

  Example 3 shows one IP phone that is connected to the 10/100 port on the Catalyst 6500 series switch and one PC that is connected to the PC-to-phone jack on the phone. The PC behaves as if it is connected directly to the 10/100 port on the Catalyst 6500 series switch. The phone can be powered through the 10/100 port or wall powered. The PC must be wall powered.

- Example 4: Two Cisco IP Phone 7960s and One PC

  Example 4 shows two IP phones that are connected to the 10/100 port on the Catalyst 6500 series switch and one PC that is connected to the PC-to-phone jack on the phone. The PC behaves as if it is connected directly to the 10/100 port on the Catalyst 6500 series switch. The first phone can be powered through the 10/100 port or wall powered. The second phone and the PC must be wall powered.

Note For more information on configuring the Cisco IP phones and third-party vendor phones, refer to the documentation that shipped with the phone.

Cisco CallManager

Cisco CallManager is an open and industry-standard call processing system; its software runs on a Windows NT server and sets up and tears down the calls between the phones, integrating traditional PBX functionality with the corporate IP network. Cisco CallManager manages the components of the IP PBX system, the phones, the access gateways, and the resources for such features as call conferencing and media mixing. Each Cisco CallManager manages the devices within its *zone* and exchanges information with the Cisco CallManager in charge of another zone to make the calls possible across multiple zones. Cisco CallManager can work with the existing PBX systems to route a call over the Public Switched Telephone Network (PSTN).

Note For information on configuring Cisco CallManager to work with the IP devices that are described in this chapter, refer to the *Cisco CallManager Administration Guide,* the *Configuration Notes for Cisco CallManager,* and the *Cisco CallManager Remote Serviceability Users Guide* publications.

Access Gateways

The access gateways allow the IP PBX system to talk to the existing PSTN or PBX systems. The access gateways consist of analog station gateways, analog trunk gateways, digital trunk gateways, and a *converged* voice gateway.

These sections describe the gateways:

- Analog Station Gateway, page 55-5
- Analog Trunk Gateway, page 55-6
- Digital Trunk Gateway, page 55-6
- Converged Voice Gateway, page 55-7

Analog Station Gateway

The Catalyst 6500 series 24-port Foreign Exchange Station (FXS) analog interface module allows the plain old telephone service (POTS) phones and fax machines to connect to the IP PBX network. The analog station gateway behaves like the PSTN side for the POTS equipment. It requires an IP address, is registered with Cisco CallManager in its domain, and is managed by Cisco CallManager.

To configure the analog station interfaces, see the "Configuring VoIP on a Switch" section on page 55-10. The module features are listed in Table 55-1.

Table 55-1 24-Port FXS Analog Interface Module Features

Digital Signal Processing Per Port
G.711 and G.729 voice encoding
Silence suppression; voice activity detection
Comfort noise generation
Ringer, software programmable frequency and cadence, based on country
DTMF[1] detection
Signaling, loop start

Table 55-1 24-Port FXS Analog Interface Module Features (continued)

Digital Signal Processing Per Port
Line echo cancellation (32 ms)
Impedance (600 ohms)
Programmable analog gain, signaling timers
Fax pass-through
SPAN[2] or port mirroring support
FXS Interface Features
Address signaling formats: In-band DTMF
Signaling formats: Loop start
Ringing tone: Programmable
Ringing voltage: Programmable, based on country
Ringing frequency: Programmable, based on country
Distance: 500-ohms maximum loop

1. DTMF = dual tone multifrequency
2. SPAN = Switched Port Analyzer

Analog Trunk Gateway

The Cisco access analog trunk gateways allow the IP PBX to connect to the PSTN or PBX. The gateway supports up to eight trunks to the PSTN and appears like a phone to the trunk lines coming from the PSTN. Using this gateway, the IP PBX places an IP call through the PSTN. Similar to the analog station gateway, the analog trunk gateway provides line echo cancellation and dual tone multifrequency (DTMF) tone generation and detection. The analog trunk gateway does not provide the ring voltage as it is not connected to the POTS end devices such as the POTS phones or fax machines. The analog trunk gateway requires an IP address, is registered with Cisco CallManager in its domain, and is managed by Cisco CallManager.

To configure the analog trunk gateways, refer to the documentation that shipped with the gateway.

Digital Trunk Gateway

The Catalyst 6500 series 8-port T1/E1 PSTN interface module can support both digital T1/E1 connectivity to the PSTN or transcoding and conferencing. The module requires an IP address, is registered with Cisco CallManager in its domain, and is managed by Cisco CallManager.

The module software is downloaded from a TFTP server. Depending upon which software you download, the ports can serve as the T1/E1 interfaces or the ports support transcoding and conferencing. The transcoding and conferencing functions are mutually exclusive. For every transcoding port in use, one less conferencing port is available and vice versa.

To configure the 8-port T1/E1 PSTN interfaces, see the "Configuring VoIP on a Switch" section on page 55-10. The module features are listed in Table 55-2.

Table 55-2 8-Port T1/E1 PSTN Interface Module Features

Digital Signal Processing Per T1/E1 Port
G.711 to G.723 and G.729a transcoding (maximum of 8 x 32 channels of transcoding)
Conference bridging, meet-me, and ad-hoc conference modes (maximum of 8 x 16 channels of conferencing)
Comfort noise generation
Fax pass-through
Silence suppression, voice activity detection
Line echo cancellation
Common channel signaling For T1: 23 DS0 channels for voice traffic; 24th channel is used for signaling For E1: 29 DS0 channels for voice traffic; 16th channel is reserved for signaling Any channel can be configured for common channel signaling
ISDN Primary Rate Interface signaling: Each interface supports 23 channels for T1 and 30 channels for E1. The default mode is for the 24th T1 channel or 16th E1 channel to be reserved for signaling. Both network side and user side operation modes are supported.
T1 binary 8-zero substitution/alternate mark inversion (B8ZS/AMI) line coding, u-law or a-law coding E1 HDB3 line coding
T1 line bit rate: 1.544 Mbps E1 line bit rate: 2.048 Mbps
T1 line code: AMI, B8ZS E1 line code: HDB3
Framing format: D4 superframe and extended superframe
Link Management
FDL[1] is a link management protocol that is used to help diagnose problems and gather statistics on T1 lines

1. FDL = Facilities Data Link

Converged Voice Gateway

The Cisco Voice Gateway 200 (VG200) allows you to connect the standard POTS phones (connected directly to the gateway or anywhere on the PSTN) with Cisco IP or any H.323-compliant telephony devices. When used with Cisco CallManager, the VG200 functions as a Media Gateway Control Protocol (MGCP) gateway. The Cisco VG200 provides a 10/100BASE-T Ethernet port for connection to the data network. The following telephony connections are also available:

- One to four Foreign Exchange Office (FXO) ports for connecting to a central office or PBX
- One to four FXS ports for connecting to POTS telephony devices

- One or two T1 digital ports for connecting to the following:
  - PSTN using FXO emulation
  - T1 channel bank using FXS emulation
  - PBX through a trunk (tie) line using ear and mouth (E&M) emulation

These ports can be used to integrate a VoIP network with POTS devices, PBXs, or the PSTN.

To configure the Cisco VG200, refer to the documentation that shipped with the gateway.

How a Call Is Made

An IP phone connects to a LAN either through a hub port or a switch port. The IP phone boots up and uses DHCP to get its IP address and the IP address of its TFTP file server. The IP phone uses its IP address to talk to the TFTP server and gets its configuration file. The configuration file includes the IP address of the phone's Cisco CallManager(s). The phone then talks with Cisco CallManager and registers itself. Each time a phone boots up, it might get a different IP address. Cisco CallManager knows how to associate a consistent user phone number to a particular phone by using the MAC address of the phone. Cisco CallManager always maintains a table mapping the phone MAC address and phone number. Each time a phone registers, the table is updated with the new IP address. During the registration, Cisco CallManager downloads the key pad template and the feature capability for the phone. It tells the phone which run-time image it should use. The phone then goes to the TFTP server to get its run-time image. Each phone has a dedicated TCP connection to Cisco CallManager called the control channel. All control information, such as key pressing, goes from the phone to Cisco CallManager through this channel. Instructions to generate ring tone, busy tone, and so on comes from Cisco CallManager to the phone through this channel.

Cisco CallManager stores the IP-address-to-phone-number mapping (and vice versa) in its tables. When a user wants to call another user, the user keys in the called party's phone number. Cisco CallManager translates the phone number to an IP address and generates an IP packet version of the ring tone to the called IP phone through the TCP connection. When the called IP phone receives the packet, it generates a ring tone. When the user picks up the phone, Cisco CallManager instructs the called IP phone to start talking with the calling party and removes itself from the loop. From this point on, the call goes between the two IP phones through the Real-Time Transport Protocol (RTP) which runs over the User Datagram Protocol (UDP). Because the voice packets are sensitive to delays, TCP is not suitable for voice transmission because the timeouts and retries increase the delay between the packets. When any change occurs during the call due to a feature being pressed on one of the phones, or one of the users hanging up or pressing the flash button, the information goes to Cisco CallManager through the control channel.

If a call is made to a number outside of the IP PBX network, Cisco CallManager routes the call to an analog or digital trunk gateway which routes it to the PSTN.

Understanding How VLANs Work

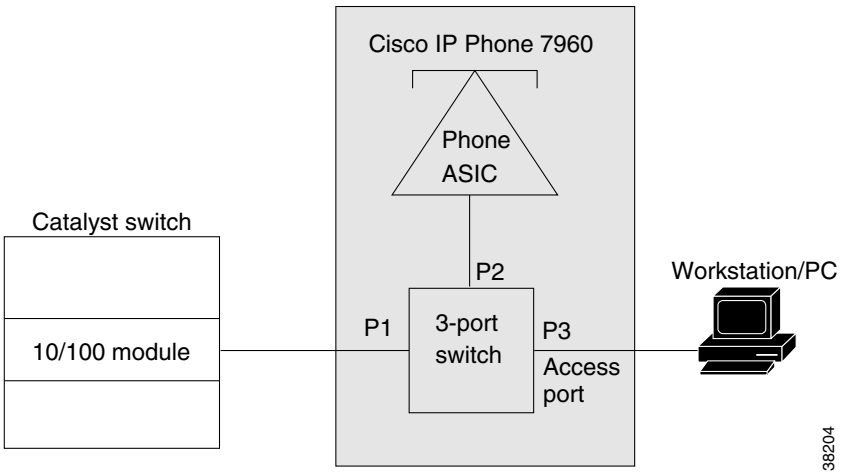
This section describes the native VLANs and the auxiliary VLANs. This section uses the following terminology:

- Auxiliary VLAN—Separate VLAN for IP phones
- Native VLAN—Traditional VLAN for data
- Auxiliary VLAN ID—VLAN ID of an auxiliary VLAN
- Native VLAN ID—VLAN ID of a native VLAN

Note For more information about the VLANs, see Chapter 11, "Configuring VLANs."

Figure 55-3 shows how to connect a Cisco IP Phone 7960 to a Catalyst 6500 series switch.

Figure 55-3 Switch-to-Phone Connections



When the IP phone connects to a 10/100 port on the Catalyst 6500 series switch, the *access port* (PC-to-phone jack) of the IP phone can be used to connect a PC.

The packets to and from the PC and to and from the phone share the same physical link to the switch and the same port of the switch. The various configurations are shown in the "Cisco IP Phone 7960" section on page 55-2).

Introducing the IP-based phones into the existing switch-based networks raises the following issues:

- The current VLANs might be configured on an IP subnet basis, and additional IP addresses might not be available to assign the phone to a port so that it belongs to the same subnet as other devices (PC) that are connected to the same port.
- The data traffic present on the VLAN supporting phones might reduce the quality of the VoIP traffic.

You can resolve these issues by isolating the voice traffic onto a separate VLAN on each of the ports that are connected to a phone. The switch port that is configured for connecting a phone would have separate VLANs that are configured for carrying the following:

- Voice traffic to and from the IP phone (auxiliary VLAN)
- Data traffic to and from the PC that is connected to the switch through the access port of the IP phone (native VLAN)

Isolating the phones on a separate, auxiliary VLAN increases the quality of the voice traffic and allows a large number of phones to be added to an existing network where there are not enough IP addresses. A new VLAN means a new subnet and a new set of IP addresses.

Understanding How CDP and VoIP Work

Cisco Discovery Protocol (CDP) was enhanced in software release 8.1(1) to facilitate backward compatibility with the newer, higher-powered Cisco IP phones. With this enhanced CDP, a Cisco IP phone can negotiate its power requirements to the switch within the CDP packet. The switch uses this information to ensure that it does not oversubscribe the available power.

We recommend that you enable CDP on the switch so that the switch can correctly detect and supply power to the IP phones that are connected to it. CDP is enabled on the Catalyst 6500 series switches by default; however, you should confirm that CDP is enabled when setting up your VoIP network. For more information on CDP, see Chapter 31, "Configuring CDP."

Configuring VoIP on a Switch

This section describes the command-line interface (CLI) commands and the procedures that are used to configure the Catalyst 6500 series switch for VoIP operation:

- Voice-Related CLI Commands, page 55-10
- Configuring Per-Port Power Management, page 55-11
- Configuring the Auxiliary VLANs on Catalyst LAN Switches, page 55-20
- Configuring the Access Gateways, page 55-23
- Displaying the Active Call Information, page 55-29
- Configuring QoS in the Cisco IP Phone 7960, page 55-31
- Configuring a Trusted Boundary to Ensure Port Security, page 55-33

Note For information on using automatic voice configuration, see the "Using SmartPorts" section on page 55-38.

Note You must enable CDP on the Catalyst 6500 series switch port that is connected to the IP phone in order to communicate the auxiliary VLAN ID, per-port power management details, and quality of service (QoS) configuration information.

Voice-Related CLI Commands

Table 55-3 lists the CLI commands that are described in the configuration procedures.

Table 55-3 Voice-Related CLI Command Module and Platform Support

CLI Commands	Ethernet Module[1]	WS-X6608-T1/E1[2]	WS-X6624-FXS[3]
Inline-power related commands			
set port inlinepower	X[4]		
set inlinepower defaultallocation	This is a switch-level command and does not affect the individual modules.		
show port inlinepower	X		

Table 55-3 Voice-Related CLI Command Module and Platform Support (continued)

CLI Commands	Ethernet Module[1]	WS-X6608-T1/E1[2]	WS-X6624-FXS[3]
show environment power	X	X	X
Voice-related commands			
set port auxiliaryvlan	X/X		
show port auxiliaryvlan	X/X		
set port voice interface		X	X
show port voice interface		X	X
show port voice	X	X	X
show port voice fdl		X	
show port voice active	X	X	X
QoS commands related to voice			
set port qos *mod/port* **cos-ext** **set port qos** *mod/port* **trust-ext**	X/X		
show port qos	X/X		

1. Ethernet Module = Ethernet switching module with voice daughter card.
2. WS-X6608-T1 and WS-X6608-E1 = 8-port T1/E1 ISDN PRI modules.
3. WS-X6624-FXS = 24-port FXS analog station interface module.
4. X = Command supported on Catalyst 6500 series switch only; XX = Command supported on Catalyst 4500 series, 5000 family, and 6500 series switches. All modules that are listed in Table 55-3 are supported only on Catalyst 6500 series switches.

Configuring Per-Port Power Management

This section describes the per-port power management and the CLI commands that are used to configure power management for IP phones.

Note To determine the exact power requirements for your configuration to ensure that you are within the system power budget, see the "Generating a System Status Report" section on page 22-17.

Note This section applies to the Ethernet switching modules with the voice daughter card only. For information on powering the IP phones that are connected to the other Ethernet switching modules, refer to the *Catalyst Family Inline-Power Patch Panel Installation Note* publication.

For each IP phone that is connected to an Ethernet switching module with a voice daughter card installed, the module allocates part of the available system power to power up and run the phone. You can apply the power on an individual port basis.

Only one IP phone can be powered per port; the phone must be connected directly to the switch port. If a second phone is daisy chained off the phone that is connected to the switch port, the second phone cannot be powered by the switch.

This section describes the following topics:

Using show Commands to Display Module Type and Version Information

To determine if the module has a voice daughter card installed, enter the **show module** command and look at the "Sub" field. For example, in the following display, the 10/100BASE-TX module in slot 3 has a voice daughter card.

To display the module status and information, perform this task in normal mode:

Task	Command
Display the module status and information.	**show module** [*mod*]

This example shows a submodule field that provides information about the submodules. The inline power daughter card that is installed on module 3, as shown in the display, is WS-F6K-SVDB-FE, and the inline power daughter card that is installed on module 6, as shown in the display, is WS-F6K-VPWR-GE-TX.

```
Console> (enable) show module
Mod Slot Ports Module-Type                Model               Sub Status
--- ---- ----- ------------------------- ------------------- --- --------
1   1    2     1000BaseX Supervisor      WS-X6K-SUP2-2GE     yes ok
3   3    48    10/100BaseTX Ethernet     WS-X6548-RJ-45      yes ok
4   4    48    10/100BaseTX Ethernet     WS-X6148-RJ45V      no  ok
6   6    48    10/100/1000BaseT Ethernet WS-X6148-GE-TX      yes ok

Mod Module-Name          Serial-Num
--- -------------------- -----------
1                        SAD04460M9G
3                        SAD0447099V
4                        SAD061901FL
6                        SAD0706025A

Mod MAC-Address(es)                         Hw     Fw          Sw
--- --------------------------------------- ------ ----------- -----------------
1   00-d0-c0-d4-04-4e to 00-d0-c0-d4-04-4f 1.1    6.1(2)      7.7(0.82-Eng)
    00-d0-c0-d4-04-4c to 00-d0-c0-d4-04-4d
    00-02-4a-30-88-00 to 00-02-4a-30-8b-ff
3   00-02-b9-ff-eb-70 to 00-02-b9-ff-eb-9f 0.203  6.3(1)      8.2(1)
4   00-00-00-00-00-00 to 00-00-00-00-00-2f 1.3    5.4(2)      7.7(0.81)
6   00-40-0b-ff-00-00 to 00-40-0b-ff-00-2f 0.304  7.2(1)      8.2(1)
```

```
Mod Sub-Type                 Sub-Model           Sub-Serial  Sub-Hw Sub-Sw
--- ------------------------ ------------------- ----------- ------ ------
1   L3 Switching Engine II   WS-F6K-PFC2         SAD044302EA 1.0
3   IEEE InlinePower Module  WS-F6K-FE48-AF      sasdfasdf   0.1    8.1(0)
6   Inline Power Module      WS-F6K-VPWR-GE      SAD070700GV 0.201  8.1(0)
Console> (enable)
```

To display the module and submodule versions, perform this task in normal mode:

Task	Command
Display the module and submodule versions.	**show version** [*mod*]

This example shows how to display the module and submodule versions:

```
Console> (enable) show version 6
Mod Port Model               Serial #    Versions
--- ---- ------------------- ----------- -------------------------------------
6   48   WS-X6148-GE-TX      SAD0706025A Hw :0.304
                                         Fw :7.2(1)
                                         Sw :8.1(0)
         WS-F6K-VPWR-GE      SAD070700GV Hw :0.201
                                         Sw :8.1(0)
Console>
```

Power Management Modes

Each port is configured through the CLI, SNMP, or a configuration file to be in one of the following modes. The CLI command is **set port inlinepower** *mod/port* {{**auto** | **static** | **limit**} [*wattage*] | **off**}.

- **auto**—Discovery is enabled and the supervisor engine directs the switching module to power up the port *only* if the switching module discovers the phone. You can specify the maximum wattage that is allowed on the port. If you do not specify a wattage, then the switch will deliver no more than the hardware-supported maximum value.
- **static**—Discovery is enabled and the supervisor engine directs the switching module to power up the port to the wattage that you specify *only* if the switching module discovers the phone. You can specify the maximum wattage that is allowed on the port. If you do not specify a wattage, then the switch allows the hardware-supported maximum value. The maximum wattage, whether determined by the switch or specified by you, is preallocated to the port. If the switch does not have enough power for the allocation, the command will fail.
- **off**—Discovery is disabled which prevents the port from providing power to an external device. If the external device is wall-powered and the inline power is off, the port should still link up, join the bridge group, and go to the STP forwarding state.
- **limit**—Discovery is enabled. This mode provides you with the option to limit the power allocated for an external device. If the wattage value that you specify with the **limit** keyword is less than the power determined through IEEE classification, instead of denying power, the minimum of these two values is allocated. If the device consumes more than the configured value, the port is shut down and an appropriate syslog message is displayed. The **limit** keyword is not supported on all modules. To check if the **limit** keyword is supported on a module, enter the **show environment power** *mod* command. If the output of the command indicates support for per-port power monitoring, the mode is supported.
- *max-wattage*—(Optional) The maximum power allowed on the port in either **auto** or **static** mode; valid values are from 4000 to 15400 milliwatts.

Each port also has a status that is defined as one of the following:

- on—Power is supplied by the port.
- off—Power is not supplied by the port.
- Power-deny—The supervisor engine does not have enough power to allocate to the port, or the power that is configured for the port is less than the power that is required by the port; the power is not being supplied by the port.
- err-disable—The port is unable to provide the power to the connected device that is configured in Static mode.
- faulty—The port failed the diagnostics tests.

These sections provide the information on the IP phone power requirements and management:

- Power Requirements, page 55-14
- Available Power, page 55-15
- Wall-Powered Phones, page 55-15
- Powering Off the Phone, page 55-15
- Phone Removal, page 55-15
- High-Availability Support, page 55-16

Power Requirements

The IP phones may have different power requirements. Table 55-4 lists the power requirements for the different classes of IP phones. The supervisor engine initially calculates the power allocation for each port based on the per-port configuration, classification (IEEE only), and default power. When the correct amount of power is determined from the CDP messaging with the Cisco IP Phone, the supervisor engine reduces or increases the allocated power for any ports that are set to Auto mode. The allocated power is not adjusted for ports that are set to Static mode.

For example, the default allocated power is 7 W for a Cisco IP Phone requiring 6.3 W. The supervisor engine allocates 7 W for the Cisco IP Phone and powers it up. Once the Cisco IP Phone is operational, it sends a CDP message with the actual power requirement to the supervisor engine. The supervisor engine then decreases the allocated power to the required amount if the port is set to Auto mode. If the port is set to Static mode, the supervisor engine allocates the wattage that you specified. If the port is set to off, the supervisor engine does not allot any power to the port.

Table 55-4 Power Requirements for IP Phones

Phone Class	Required Power (W)
Cisco	6.3
Cisco + IEEE	7
Cisco High Power	15.4
Class 0 IEEE	15.4
Class 1 IEEE	4
Class 2 IEEE	7.0
Class 3	15.4
Class 4 Refer to Class 0	Reserved

Available Power

Table 55-5 lists the available power that can be supplied for each port for the voice daughter cards.

Table 55-5 Efficiency of Voice Daughter Cards

Daughter Card	Maximum Power Per Port (W)	Efficiency
WS-F6K-PWR	6.3	100%
WS-F6K-VPWR-GE	6.3	89%
WS-F6K-GE48-AF	15	89%
WS-F6K-FE48-AF	15	89%
WS-F6K-FE96-AF	15	89%

For example, if the powered device requires 6.3 W, then the allotted power for that port using a daughter card with 89 percent efficiency must be 6.3/(0.89) = 7 W. If you are using a voice daughter card with 100 percent efficiency, then the allotted power is 6.3 W.

Wall-Powered Phones

When a wall-powered phone is present on a switching module port, the switching module cannot detect its presence. The supervisor engine discovers the phone through CDP messaging with the port. If the phone supports the inline power (the supervisor engine determines this through CDP), and the mode is set to Auto, Static, or Off, the supervisor engine does not attempt to power on the port. If a power outage occurs, and the mode is set to Auto, the phone loses power, but the switching module discovers the phone and informs the supervisor engine, which then applies the inline power to the phone. If a power outage occurs, and the mode is set to Static, the phone loses power, but the switching module discovers the phone and applies the preallocated inline power to the phone.

Powering Off the Phone

The supervisor engine can turn off power to a specific port by sending a message to the switching module. The power for a port in Auto mode is then added back to the available system power. The power for the ports in Static mode is not added back to the available system power. This situation occurs only when you power off the phone through the CLI or SNMP.

Phone Removal

The switching module informs the supervisor engine if a *powered* phone is removed using a link-down message. The supervisor engine then adds the allocated power for that port back to the available system power.

In addition, the switching module informs the supervisor engine if an *unpowered* phone is removed.

Caution When a phone cable is plugged into a port and the power is turned on, the supervisor engine has a 4-second timeout waiting for the link to go up on the line. During those 4 seconds, if the phone cable is unplugged and a network device is plugged in, the device could be damaged. We recommend that you wait at least 10 seconds between unplugging a device and plugging in a new device.

High-Availability Support

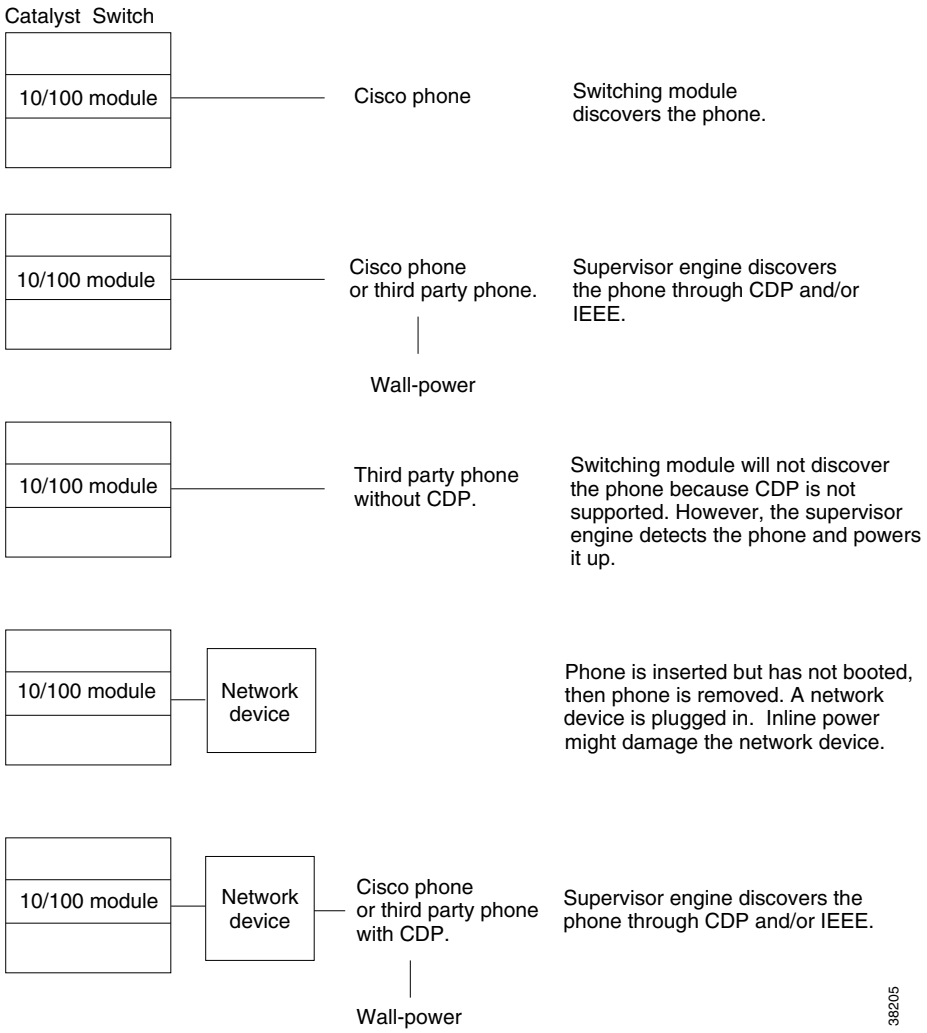
To support high availability during a failover from the active supervisor engine to the standby supervisor engine, the per-port power management and phone status information is synchronized between the active and standby supervisor engines.

The information to be synchronized (on a per-port basis) is the presence of a phone, the phone power status (on, off, denied, or faulty), allocated power, device class, device type, device maximum power, and device discovery. The active supervisor engine sends this information to the standby supervisor engine, and the standby supervisor engine updates its internal data structures. When a switchover occurs, the standby supervisor engine allocates the power to the modules and ports from the available power, one module at a time. Once the power for each module has been allocated, the supervisor engine allocates the power to the phones, beginning with the lowest slot number, until all inline powered ports have been either powered on, off, or denied.

Phone Detection Summary

Figure 55-4 shows how the system detects a phone that is connected to a Catalyst 6500 series switch port.

Figure 55-4 Power Detection Summary



Setting the Power Mode of a Port or a Group of Ports

To set the power mode of a port or a group of ports, perform this task in normal mode:

Task	Command
Set the power mode of a port or a group of ports.	**set port inlinepower** *mod/port* {[**auto** | **static**] [*max-wattage*] | **off**}

Note If you configure the *max-wattage* values that are multiples of 500 on a Catalyst 6500 series switch with the **set port inlinepower** *mod/port* **static | auto** *max-wattage* command, the power that is drawn from the global allocation is possibly slightly smaller than the power that is reported in the Total PWR Allocated to Module field of the **show environment power** command. This discrepancy is due to the internal conversion of units from Watts to cAmps and back to Watts. The difference between the total allocated power and the total power that is drawn from the system is no more than +/- 0.42 W.

This example shows how to set the power mode of a port or group of ports:

```
Console> (enable) set port inlinepower 2/5 off
Inline power for port 2/5 set to off.
Console> (enable) set port inlinepower 2/3-9 auto 800
Inline power for ports 2/3-9 set to auto and max-wattage to 800.
Console> (enable)
```

Setting the Default Power Allocation

The **set inlinepower defaultallocation** command is global and only affects Cisco IP phones. The inline power threshold notification generates a syslog message when the inline power usage exceeds the specified threshold. To set the default power allocation, perform this task in privileged mode (the default allocation value is 15400 milliwatts):

Caution The **set inlinepower defaultallocation** command can be harmful when there is not enough power in the system to bring up all connected inline power devices. If you set a small value for the power allocation, all connected inline power devices initially will be powered up. However, after receiving CDP messages, the system will learn that devices are consuming more power and deny power to some of the ports. Setting a small value might also result in the overdrawing of power for some time with unanticipated results, such as hardware failures and unexpected resets.

Note 7000 milliwatts is the maximum power supported for these modules: WS-X6348-RJ21V, WS-X6348-RJ-45V, WS-X6148-RJ-45V, and WS-X6148-RJ21V.

Task	Command
Set the default power allocation.	**set inlinepower defaultallocation** *value*

This example shows how to set the default power allocation:

```
Console> (enable) set inlinepower defaultallocation 9500
Default inline power allocation set to 9500 mWatt per applicable port.
Console> (enable)
```

Setting the Inline Power Notification Threshold for a Module

Use the **set inlinepower notify-threshold** command to set a threshold for inline power usage. The threshold is a percentage from 1 through 99, with 99 percent being the default. When the threshold is passed, a syslog and trap (if configured) are generated.

To set the inline power notification threshold for a module, perform this task in privileged mode:

Task	Command
Set the inline power notification threshold for a module.	**set inlinepower notify-threshold** {*percentage value*} **module** {*mod_num*}

This example shows how to set the inline power notification threshold to 50 for module 4:

```
Console> (enable) set inlinepower notify-threshold 50 mod 4
Module 4 inlinepower notify-threshold is set to 50%.
Console> (enable)
```

Displaying the Power Status for Modules and Individual Ports

To display the power status for the modules and individual ports, perform this task in normal mode:

Task	Command
Display the power status for the modules and individual ports.	**show port inlinepower** [*mod*[*/port*]] [**detail**]

This example shows how to display the power status for the modules and individual ports:

```
Console> show port inlinepower 6/1
Configured Default Inline Power allocation per port: 15.400 Watts (0.36
Amps @42V)
Total inline power drawn by module 4:  33.934 Watts ( 0.807 Amps @42V)

Port  InlinePowered  PowerAllocated   Device     IEEE class
                    From PS   To PD
      Admin  Oper   mWatts    mWatts
----- ------ ------ -------   ------- ---------- ----------
 6/1  auto   on     7079      6300    cisco      none

Port  MaximumPower  ActualConsumption
      mWatts        mWatts
----- ------------  -----------------
 6/1  15400         6300

Console>
```

This example shows how to display the detailed power status for the modules and individual ports:

```
Console> show port inlinepower 4/1 detail
Configured Default Inline Power allocation per port: 15.400 Watts (0.36
Amps @42V)
Total inline power drawn by module 4:  33.934 Watts ( 0.807 Amps @42V)

Port      InlinePowered      PowerAllocated  Device     IEEE class DiscoverMode
                             From PS To PD
      Admin  Oper   Detected mWatts  mWatts
----- ------ ------ -------- ------- ------- ---------- ---------- ------------
 4/1  auto   on     yes      7079    6300    cisco      none        cisco

Port  MaximumPower  ActualConsumption  absentCounter  OverCurrent
      mWatts        mWatts
----- ------------  -----------------  -------------  -----------
 4/1  15400         6300               0              0
Console>
```

Displaying the Switch Power Environment for Modules

To display the switch power environment for the modules, perform this task in privileged mode:

Task	Command
Display the switch power environment for the modules.	**show environment power** [*mod*]

This example shows how to display the switch power environment for the modules:

```
Console> (enable) show environment power 2
Feature not supported on module 2.
Console> (enable)

Console> (enable) show environment power
PS1 Capacity:1153.32 Watts (27.46 Amps @42V)
PS2 Capacity:none
PS Configuration :PS1 and PS2 in Redundant Configuration.
Total Power Available:1153.32 Watts (27.46 Amps @42V)
Total Power Available for Line Card Usage:1153.32 Watts (27.46 Amps @42V)
Total Power Drawn From the System:683.76 Watts (16.28 Amps @42V)
Total Inline Power Drawn From the System: 57.54 Watts ( 1.37 Amps @42V)
Remaining Power in the System:469.56 Watts (11.18 Amps @42V)
Configured Default Inline Power allocation per port:15.400 Watts (0.36 Amps
@42V)

Slot power Requirement/Usage :

Slot Card Type           PowerRequested PowerAllocated CardStatus
                         Watts   A @42V Watts   A @42V
---- ------------------- ------- ------ ------- ------ ----------
1    WS-X6K-SUP2-2GE     128.52   3.06  128.52   3.06  ok
2                          0.00   0.00  128.52   3.06  none
3    WS-X6548-RJ-45      123.06   2.93  123.06   2.93  ok
4    WS-X6148-RJ45V      100.38   2.39  100.38   2.39  ok
6    WS-X6148-GE-TX      145.74   3.47  145.74   3.47  ok
```

```
Slot Inline Power Requirement/Usage :

Slot CardType            Total Allocated   Max H/W Supported  Max H/W
Supported
                         To Module (Watts) Per Module (Watts) Per Port (Watts)
---- ------------------- ----------------- ------------------ ----------------
3    WS-X6548-RJ-45      31.08             315.84             15.400
6    WS-X6148-GE-TX      26.46             315.84             7.000
Console> (enable)
```

A partial-deny status indicates that some module ports are inline powered but not all the ports on the module are inline powered.

Configuring the Auxiliary VLANs on Catalyst LAN Switches

These sections describe how to configure auxiliary VLANs:

Understanding the Auxiliary VLANs

You can configure the switch ports to send CDP packets that instruct an attached Cisco IP Phone 7960 to transmit the voice traffic to the switch in these frame types:

- 802.1Q frames carrying the auxiliary VLAN ID and Layer 2 CoS set to 5 (the switch port drops all 802.1Q frames except those carrying the auxiliary VLAN ID).
  - Reset the Cisco IP Phone 7960 if the auxiliary VLAN ID changes.
  - Enter the **set port auxiliaryvlan** *mod*[*/port*] *aux_vlan_id* command.

Note We recommend that you use 802.1Q frames and a separate VLAN.

- 802.1p frames, which are 802.1Q frames carrying VLAN ID 0 and Layer 2 CoS set to 5 (enter the **set port auxiliaryvlan** *mod*[*/port*] **dot1p** command).
- 802.3 frames, which are untagged and carry no VLAN ID and no Layer 2 CoS value (enter the **set port auxiliaryvlan** *mod*[*/port*] **untagged** command).

Note The Cisco IP Phone 7960 always sets the Layer 3 IP precedence to 5 in the voice traffic.

Auxiliary VLAN Configuration Guidelines

This section describes the guidelines for configuring the auxiliary VLANs:

- An auxiliary VLAN port is operationally a trunk, even though it is not treated like a "normal" trunk port. When an auxiliary VLAN is added to a port and the **set dot1q-all-tagged** command is enabled, the **set dot1q-all-tagged** command tags the native VLAN on the port where the auxiliary VLAN is configured. A port with an auxiliary VLAN configured is not viewed as an 802.1Q trunk in the **show trunk** command output, but the port acts like an 802.1Q trunk if the **set dot1q-all-tagged** command is enabled.
- The IP phone and a device that is attached to the phone are in the same VLAN and must be in the same IP subnet if one of the following occurs:
  - They use the same frame type.
  - The phone uses 802.1p frames, and the device uses untagged frames.
  - The phone uses untagged frames, and the device uses 802.1p frames.
  - The phone uses 802.1Q frames, and the auxiliary VLAN equals the native VLAN.
- The IP phone and a device that is attached to the phone cannot communicate if they are in the same VLAN and subnet but use different frame types, because the traffic between the devices in the same subnet is not routed (routing would eliminate the frame type difference).
- You cannot use the switch commands to configure a frame type that is used by the traffic that is received from a device that is attached to the phone's access port.
- With software release 6.2(1) and later releases, the dynamic ports can belong to two VLANs—a native VLAN and an auxiliary VLAN. See Chapter 19, "Configuring Dynamic Port VLAN Membership with VMPS," for the configuration details for the auxiliary VLANs.

Configuring the Auxiliary VLANs

To configure the auxiliary VLANs, perform this task in privileged mode:

Task	Command
Configure the auxiliary VLANs.	**set port auxiliaryvlan** *mod*[*/ports*] {*vlan* | **untagged** | **dot1p** | **none**}

This example shows how to add the voice ports to the auxiliary VLANs, specify an encapsulation type, or specify that the VLAN will not send or receive CDP messages with voice-related information:

```
Console> (enable) set port auxiliaryvlan 2/1-3 222
Auxiliaryvlan 222 configuration successful.
AuxiliaryVlan AuxVlanStatus Mod/Ports
------------- ------------ ------------------------
222           active       1/2,2/1-3
Console> (enable) set port auxiliaryvlan 5/7 untagged
Port 5/7 allows the connected device send and receive untagged packets and without 802.1p
priority.
Console> (enable) set port auxiliaryvlan 5/9 dot1p
Port 5/9 allows the connected device send and receive packets with 802.1p priority.
Console> (enable) set port auxiliaryvlan 5/12 none
Port 5/12 will not allow sending CDP packets with Voice VLAN information.
Console> (enable)
```

The default setting is **none**. Table 55-6 lists the **set port auxiliaryvlan** command keywords and their descriptions.

Table 55-6 Keyword Descriptions

Keyword	Action
dot1p	Specify that the phone sends the packets with 802.1p priority 5.
untagged	Specify that the phone sends the untagged packets.
none	Specify that the switch does not send any auxiliary VLAN information in the CDP packets from that port.

Verifying the Auxiliary VLAN Configuration

To verify the auxiliary VLAN configuration status, perform this task in privileged mode:

Task	Command
Verify the auxiliary VLAN configuration status.	**show port auxiliaryvlan** {*vlan* | **untagged** | **dot1p** | **none**}

This example shows how to verify the auxiliary VLAN configuration status:

```
Console> show port auxiliaryvlan 123
AuxiliaryVlan AuxVlanStatus Mod/Ports
------------- ------------- ------------------------
222           active        1/2,2/1-3
Console>
```

Disabling the Auxiliary VLANs Until an IP Phone is Detected

With software release 8.3(1) and later releases, this feature provides security for the auxiliary VLANs by ensuring that the auxiliary VLAN is not enabled until an IP phone is detected. As soon the switch detects the presence of an IP phone, the auxiliary VLAN is enabled.

The presence of an IP phone is determined through the CDP packet exchange between the switch and the phone. This detection method is used for both the inline-powered and wall-powered IP phones.

Note If the auxiliary VLAN ID equals the port-VLAN ID or when the auxiliary VLAN ID is configured as **none**, **dot1p**, or **untagged**, this feature cannot be applied to the port. If any command entry results in the auxiliary VLAN ID equaling the port-VLAN ID, the feature is disabled and the following warning message is displayed: “cdpverify feature on port <mod>/<port> is disabled.”

To enable or disable the auxiliary VLAN IP phone detection, perform this task in privileged mode (the default is disabled):

Task	Command
Enable or disable the auxiliary VLAN IP phone detection.	**set port auxiliaryvlan** *mod*[/*port*] {*vlan* | **untagged** | **dot1p** | **none**} [**cdpverify** {**enable** | **disable**}]

This example shows how to enable or disable the auxiliary VLAN IP phone detection:

```
Console> (enable) set port auxiliaryvlan 3/1 50 cdpverify enable
AuxiliaryVlan Status   Mod/Ports
------------- -------- ----------------------------------------------------
50            active   3/1
Console> (enable)

Console> (enable) show config
This command shows non-default configurations only.
Use 'show config all' to show both default and non-default configurations.
.
.
.
!
#module 3 : 48-port 10/100BaseTX Ethernet
set port auxiliaryvlan 3/1 50 cdpverify enable
!
Console> (enable)
```

Configuring the Access Gateways

This section describes the commands that are used to configure the following Catalyst 6500 series access gateway modules:

- Analog station gateway—24-port FXS analog interface module
- Digital trunk gateway—8-port T1/E1 PSTN interface module

Configuring a Port Voice Interface

If DHCP is enabled for a port, the port obtains all other configuration information from the TFTP server. When disabling DHCP on a port, you must specify some mandatory parameters as follows:

- If you do not specify the DNS parameters, the software uses the system DNS configuration on the supervisor engine to configure the port.
- 8-port T1/E1 PSTN interface module only: You cannot specify more than one port at a time because a unique IP address must be set for each port.

To configure a port voice interface for the DHCP, TFTP, and DNS servers, perform this task in privileged mode:

Task	Command
Configure a port voice interface for the DHCP, TFTP, and DNS servers.	**set port voice interface** *mod/port* **dhcp enable** [**vlan** *vlan*] **set port voice interface** *mod/port* **dhcp disable** {*ipaddrspec*} {**tftp** *ipaddr*} [**vlan** *vlan*] [**gateway** *ipaddr*] [**dns** [*ipaddr*] [*domain_name*]]

These examples show how to configure the port voice interface for the DHCP, TFTP, and DNS servers:

```
Console> (enable) set port voice interface 7/1 dhcp enable
Port 7/1 DHCP enabled.

Console> (enable) set port voice interface 7/3 dhcp disable 171.68.111.41/24 tftp
173.32.43.11 dns 172.20.34.204 cisco.com
Port 7/3 dhcp disabled.
System DNS configurations applied.

Console> (enable) set port voice interface 7/4-6 dhcp enable vlan 3
Vlan 3 configuration successful
Ports 7/4-6 DHCP enabled.
Console> (enable)
```

Displaying a Port Voice Interface Configuration

To display a port voice interface configuration, perform this task in privileged mode:

Task	Command
Display a port voice interface configuration.	**show port voice interface** [*mod*[*/port*]]

This example shows how to display the port voice interface configuration (this display is from the 24-port FXS analog interface module):

```
Console> show port voice interface 5
Port     DHCP    MAC-Address       IP-Address      Subnet-Mask
-------- ------- ----------------- --------------- ---------------
 5/1-24  disable 00-10-7b-00-13-ea 10.6.15.158     255.255.255.0

Port     Call-Manager(s)   DHCP-Server     TFTP-Server     Gateway
-------- ----------------- --------------- --------------- ---------------
 5/1-24  10.6.15.155       -               10.6.15.155     -

Port     DNS-Server(s)     Domain
-------- ----------------- ------------------------------------------------
 5/1-24  12.2.2.1*         cisco.cisco.com
         7.7.7.7
(*): Primary
Console> (enable)
```

Displaying the FDL Statistics

Note Facilities Data Link (FDL) is a link management protocol that is used to diagnose the problems and gather the statistics.

To display the FDL statistics for the specified ports, perform this task in privileged mode:

Task	Command
Display the FDL statistics for the specified ports.	**show port voice fdl** [*mod*[*/port*]]

This example shows how to display the FDL statistics for the specified ports:

```
Console> (enable) show port voice fdl 7/1-3
Port  ErrorEvents       ErroredSecond     SeverlyErroredSecond
      Last 15' Last 24h Last 15' Last 24h Last 15' Last 24h
----- -------- -------- -------- -------- -------- -----------
 7/1  17       18       19       20       21       22
 7/2  17       18       19       20       21       22
 7/3  17       18       19       20       21       22

Port  FailedSignalState FailedSignalSecond
      Last 15' Last 24h Last 15' Last 24h
----- -------- -------- -------- ---------
 7/1  37       38       39       40
 7/2  37       38       39       40
 7/3  37       38       39       40

Port         LES               BES               LCV
      Last 15' Last 24h Last 15' Last 24h Last 15' Last 24h
----- -------- -------- -------- -------- -------- --------
 7/1  41       48       49       50       53       54
 7/2  41       48       49       50       53       54
 7/3  41       48       49       50       53       54
Console> (enable)
```

Table 55-7 describes the possible fields (depending on the port type queried) in the **show port voice fdl** command output.

Table 55-7 FDL Field Descriptions

Field	Description
ErrorEvents	Count of errored events.
ErroredSecond	Count of errored seconds.
SeverelyErroredSecond	Count of severely errored seconds.
FailedSignalState	Count of failed signal state errors.
FailedSignalSecond	Count of errored events.
LES	Line errored seconds detected.
BES	Bursty errored seconds detected.
LCV	Line code violation seconds detected.

Displaying the Port Configuration for the Individual Ports

To display the port configuration for the individual ports, perform this task in normal mode:

Task	Command
Display the port configuration for the individual ports.	**show port** [*mod*[*/port*]]

This section provides the **show port** command displays for these gateway modules:

- 8-Port T1/E1 PSTN Interface Module, page 55-26
- 8-Port T1/E1 PSTN Interface Module Configured for Trancoding/Conferencing, page 55-27
- 24-Port FXS Analog Interface Module, page 55-28

8-Port T1/E1 PSTN Interface Module

The Status field shows the Layer 2 status of the ports. The possible values are notconnect, connected, disabled, and faulty. The following display is for the T1 module. The E1 module display would be the same except that the port speed for the E1 module would be 2.048.

```
Console> show port 7
Port  Name               Status     Vlan       Duplex Speed Type
----- ------------------ ---------- ---------- ------ ----- ------------
 7/1                     connected  123        full   1.544 T1
 7/2                     connected  2          full   1.544 T1
 7/3                     disable    1          full   1.544 T1
 7/4                     connected  11         full   1.544 T1
 7/5                     connected  123        full   1.544 T1
 7/6                     connected  1          full   1.544 T1
 7/7                     faulty     2          full   1.544 T1
 7/8                     faulty     2          full   1.544 T1

Port     DHCP    MAC-Address       IP-Address      Subnet-Mask
-------- ------- ----------------- --------------- ---------------
 7/1     enable  00-10-7b-00-0a-58 172.20.34.68    255.255.255.0
 7/2     enable  00-10-7b-00-0a-59 172.20.34.70    255.255.255.0
 7/3     enable  00-10-7b-00-0a-5a 172.20.34.64    255.255.255.0
 7/4     enable  00-10-7b-00-0a-5b 172.20.34.66    255.255.255.0
 7/5     enable  00-10-7b-00-0a-5c 172.20.34.59    255.255.255.0
 7/6     enable  00-10-7b-00-0a-5d 172.20.34.67    255.255.255.0
 7/7     enable  00-10-7b-00-0a-5e (Port host processor not online)
 7/8     enable  00-10-7b-00-0a-5f (Port host processor not online)

Port     Call-Manager(s)   DHCP-Server     TFTP-Sever      Gateway
-------- ----------------- --------------- --------------- ---------------
 7/1     172.20.34.207*    172.20.34.207   172.20.34.207   -
         callm.cisco.com
 7/2     172.20.34.207     172.20.34.207   172.20.34.207   172.20.34.20
 7/3     172.20.34.207     172.20.34.207   172.20.34.207   -
 7/4     172.20.34.207     172.20.34.207   172.20.34.207   -
 7/5     172.20.34.207     172.20.34.207   172.20.34.207   -
 7/6     172.20.34.207     172.20.34.207   172.20.34.207   -
 7/7     (Port host processor not online)
 7/8     (Port host processor not online)

Port     DNS-Server(s)   Domain
-------- --------------- ------------------------------------------------
 7/1     172.20.34.207   cisco.com
 7/2     172.20.34.207*  int.cisco.com
         171.69.45.34
         172.78.111.132
 7/3     172.20.34.207   -
 7/4     172.20.34.207   -
 7/5     172.20.34.207   -
 7/6     172.20.34.207   -
 7/7     (Port host processor not online)
 7/8     (Port host processor not online)
```

```
Port     CallManagerState DSP-Type
-------- ---------------- --------
 7/1     registered       C549
 7/2     registered       C549
 7/3     registered       C549
 7/4     registered       C549
 7/5     registered       C549
 7/6     notregistered    C549
 7/7     (Port host processor not online)
 7/8     (Port host processor not online)

Port  NoiseRegen NonLinearProcessing
----- ---------- -------------------
 7/1  disabled   disabled
 7/2  disabled   disabled
 7/3  disabled   disabled
 7/4  disabled   disabled
 7/5  enabled    disabled
 7/6  disabled   enabled
 7/7  (Port host processor not online)
 7/8  (Port host processor not online)

(*): Primary
Console>
```

8-Port T1/E1 PSTN Interface Module Configured for Trancoding/Conferencing

MTP (media termination point) and Conf Bridge (conference bridge) are types of ports. Transcoding applies to a call on an MTP port.

This example shows a transcoding port as MTP and a conference port as Conf Bridge:

```
Console> (enable) show port 7
Port  Name               Status     Vlan       Duplex Speed Type
----- ------------------ ---------- ---------- ------ ----- ------------
 7/1                     notconnect 1            full 1.544 T1
 7/2                     notconnect 1            full 1.544 T1
 7/3                     connected  1            full 1.544 T1
 7/4                     connected  1            full 1.544 T1
 7/5                     connected  1            full 1.544 T1
 7/6                     connected  1            full 1.544 T1
 7/7                     enabled    1            full     - Conf Bridge
 7/8                     enabled    1            full     - MTP

Port     DHCP    MAC-Address       IP-Address      Subnet-Mask
-------- ------- ----------------- --------------- ---------------
 7/1     enable  00-10-7b-00-12-08 10.6.15.165     255.255.255.0
 7/2     enable  00-10-7b-00-12-09 10.6.15.166     255.255.255.0
 7/3     enable  00-10-7b-00-12-0a 10.6.15.167     255.255.255.0
 7/4     enable  00-10-7b-00-12-0b 10.6.15.168     255.255.255.0
 7/5     enable  00-10-7b-00-12-0c 10.6.15.169     255.255.255.0
 7/6     enable  00-10-7b-00-12-0d 10.6.15.170     255.255.255.0
 7/7     enable  00-10-7b-00-12-0e 10.6.15.171     255.255.255.0
 7/8     enable  00-10-7b-00-12-0f 10.6.15.172     255.255.255.0

Port     Call-Manager(s)   DHCP-Server     TFTP-Server     Gateway
-------- ----------------- --------------- --------------- ---------------
 7/1     10.6.15.155       10.6.15.155     10.6.15.155     -
 7/2     10.6.15.155       10.6.15.155     10.6.15.155     -
 7/3     10.6.15.155       10.6.15.155     10.6.15.155     -
 7/4     10.6.15.155       10.6.15.155     10.6.15.155     -
 7/5     10.6.15.155       10.6.15.155     10.6.15.155     -
 7/6     10.6.15.155       10.6.15.155     10.6.15.155     -
```

```
 7/7     10.6.15.155       10.6.15.155     10.6.15.155     -
 7/8     10.6.15.155       10.6.15.155     10.6.15.155     -

Port     DNS-Server(s)     Domain
-------- ----------------- ------------------------------------------------
 7/1     -                 -
 7/2     -                 -
 7/3     -                 -
 7/4     -                 -
 7/5     -                 -
 7/6     -                 -
 7/7     -                 -
 7/8     -                 -

Port     CallManagerState DSP-Type
-------- ---------------- --------
 7/1     registered       C549
 7/2     registered       C549
 7/3     registered       C549
 7/4     registered       C549
 7/5     registered       C549
 7/6     registered       C549
 7/7     registered       C549
 7/8     registered       C549

Port  NoiseRegen NonLinearProcessing
----- ---------- -------------------
 7/1  enabled    enabled
 7/2  enabled    enabled
 7/3  enabled    enabled
 7/4  enabled    enabled
 7/5  enabled    enabled
 7/6  enabled    enabled
 7/7  disabled   disabled
 7/8  disabled   disabled
Console> (enable)
```

24-Port FXS Analog Interface Module

This example shows that all ports should have a Type field of FXS, and all ports in the same module should belong to one VLAN:

```
Console> (enable) show port 3
Port  Name               Status     Vlan       Duplex Speed Type
----- ------------------ ---------- ---------- ------ ----- ------------
 3/1                     onhook     1            full   64k FXS
 3/2                     onhook     1            full   64k FXS
 3/3                     onhook     1            full   64k FXS
 3/4                     onhook     1            full   64k FXS
 3/5                     onhook     1            full   64k FXS
 3/6                     onhook     1            full   64k FXS
 3/7                     onhook     1            full   64k FXS
 3/8                     offhook    1            full   64k FXS
 3/9                     offhook    1            full   64k FXS
 3/10                    onhook     1            full   64k FXS
 3/11                    onhook     1            full   64k FXS
 3/12                    onhook     1            full   64k FXS
 3/13                    onhook     1            full   64k FXS
 3/14                    onhook     1            full   64k FXS
 3/15                    onhook     1            full   64k FXS
 3/16                    onhook     1            full   64k FXS
 3/17                    onhook     1            full   64k FXS
 3/18                    onhook     1            full   64k FXS
```

```
 3/19                    onhook     1            full   64k FXS
 3/20                    onhook     1            full   64k FXS
 3/21                    onhook     1            full   64k FXS
 3/22                    onhook     1            full   64k FXS
 3/23                    onhook     1            full   64k FXS
 3/24                    onhook     1            full   64k FXS

Port     DHCP    MAC-Address       IP-Address      Subnet-Mask
-------- ------- ----------------- --------------- ---------------
 3/1-24  enable  00-10-7b-00-13-e4 172.20.34.50    255.255.255.0

Port     Call-Manager(s)   DHCP-Server     TFTP-Sever      Gateway
-------- ----------------- --------------- --------------- ---------------
 3/1-24  172.20.34.207     172.20.34.207   172.20.34.207   -

Port     DNS-Server(s)     Domain
-------- ----------------- -------------------------------------------------
 3/1-24  172.20.34.207*    cisco.com
         172.34.23.111

Port     CallManagerState DSP-Type
-------- ---------------- --------
 3/1-24  registered       C549

Port     ToneLocal     Impedance InputGain(dB) OutputAtten(dB)
-------- ------------- --------- ------------- ---------------
 3/1-24  northamerica  0         0             0

Port     RingFreq Timing    Timing         Timing    Timing
         (Hz)     Digit(ms) InterDigit(ms) Pulse(ms) PulseDigit(ms)
-------- -------- --------- -------------- --------- --------------
 3/1-24  20       100       100            0         0
(*): Primary
Console> (enable)
```

Displaying the Active Call Information

Enter the **show port voice active** command to display the active call information on a port. There are up to 8 calls per port for the 8-port T1/E1 PSTN interface module but only one call per port for the 24-port FXS analog station interface module.

To display the active call information, perform this task in normal mode:

Task	Command
Display the active call information.	**show port voice active** [*mod/port*] [**all** | **call** | **conference** | **transcode**] [*ipaddr*]

Entering the **show port voice active** command without any parameters shows all the calls in the system (regular calls, conference calls, and transcoding calls). The display field descriptions are as follows:

- Type—The “call” notation is for the 24-port FXS analog interface module and 8-port PSTN interface module calls.

  When you configure the 8-port T1/E1 PSTN interfaces for transcoding and/or conferencing, the Type field displays “conferencing” for conferencing calls and “transcoding” for transcoding calls.

- Conference-ID, Transcoding-ID, and Party-ID are applicable only to the 8-port T1/E1 PSTN interfaces that are configured for transcoding and/or conferencing.

This example shows all the active calls in the system:

```
Console> show port voice active
Port  Type         Total Conference-ID/ Party-ID IP-Address
                         Transcoding-ID
----- ------------ ----- -------------- -------- ---------------
 3/1  call         1     -              -        199.22.25.254
 3/2  call         1     -              -        172.225.25.54
 4/5  call         3     -              -        165.34.234.111
                                                 172.32.34.12
                                                 198.96.23.111
 3/8  conferencing 2     1              1        255.255.255.241
                                        2        173.23.13.42
                                        3        198.97.123.98
                                        5        182.34.54.26
                           2            1        199.22.25.25
                                        3        182.34.54.2
                                        6        121.43.23.43
 3/2  call         1     -              -        172.225.25.54
 3/8  transcoding  1     1              1        255.255.255.241
                                        2        183.32.43.3
```

This example shows how to display the detailed call information for a port (specifying the module only, this example shows the detailed call information for all the ports on the module):

```
Console> show port voice active 3/2
Port 3/2:
Channel #1:
  Remote IP address                       : 165.34.234.111
  Remote UDP port                         : 124
  Call state                              : Ringing
  Codec Type                              : G.711
  Coder Type Rate                         : 35243
  Tx duration                             : 438543 sec
  Voice Tx duration                       : 34534 sec
  ACOM Level Current                      : 123213
  ERL Level                               : 123 dB
  Fax Transmit Duration                   : 332433
  Hi Water Playout Delay                  : 23004 ms
  Logical If index                        : 4
  Low water playout delay                 : 234 ms
  Receive delay                           : 23423 ms
  Receive bytes                           : 2342342332423
  Receive packets                         : 23423423402384
  Transmit bytes                          : 23472377
  Transmit packets                        : 94540
Channel #2:
  Remote IP address                       : 165.34.234.112
  Remote UDP port                         : 125
  Call state                              : Ringing
  Codec Type                              : G.711
  Coder Type Rate                         : 35243
  Tx duration                             : 438543 sec
  Voice Tx duration                       : 34534 sec
  ACOM Level Current                      : 123213
  ERL Level                               : 123 dB
  Fax Transmit Duration                   : 332433
  Hi Water Playout Delay                  : 23004 ms
  Logical If index                        : 4
  Low water playout delay                 : 234 ms
  Receive delay                           : 23423 ms
```

```
  Receive bytes                              : 2342342332423
  Receive packets                            : 23423423402384
  Transmit bytes                             : 23472377
  Transmit packets                           : 94540
Channel #3:
.
(display text omitted)
.
Console>
```

This example shows how to display a specific call at a specified IP address:

```
Console> show port voice active 3/2 171.69.67.91
  Remote IP address                          : 171.69.67.91
  Remote UDP port                            : 125
  Call state                                 : Ringing
  Codec Type                                 : G.711
  Coder Type Rate                            : 35243
  Tx duration                                : 438543 sec
  Voice Tx duration                          : 34534 sec
  ACOM Level Current                         : 123213
  ERL Level                                  : 123 dB
  Fax Transmit Duration                      : 332433
  Hi Water Playout Delay                     : 23004 ms
  Logical If index                           : 4
  Low water playout delay                    : 234 ms
  Receive delay                              : 23423 ms
  Receive bytes                              : 2342342332423
  Receive packets                            : 23423423402384
  Transmit bytes                             : 23472377
  Transmit packets                           : 94540
Console>
```

Configuring QoS in the Cisco IP Phone 7960

These sections describe QoS in the Cisco IP Phone 7960:

- Understanding How QoS Works in the Cisco IP Phone 7960, page 55-31
- Configuring QoS in the Cisco IP Phone 7960, page 55-32

Note For information on using automatic QoS, see Chapter 53, "Using Automatic QoS."

Note For information on using automatic voice configuration, see the "Using SmartPorts" section on page 55-38.

Understanding How QoS Works in the Cisco IP Phone 7960

Note The Cisco IP Phone 7960 always sets the Layer 3 IP precedence and Layer 2 CoS to 5 in the voice traffic that is generated by the phone. The Layer 3 IP precedence and Layer 2 CoS values in the voice traffic that is generated by the phone are not configurable.

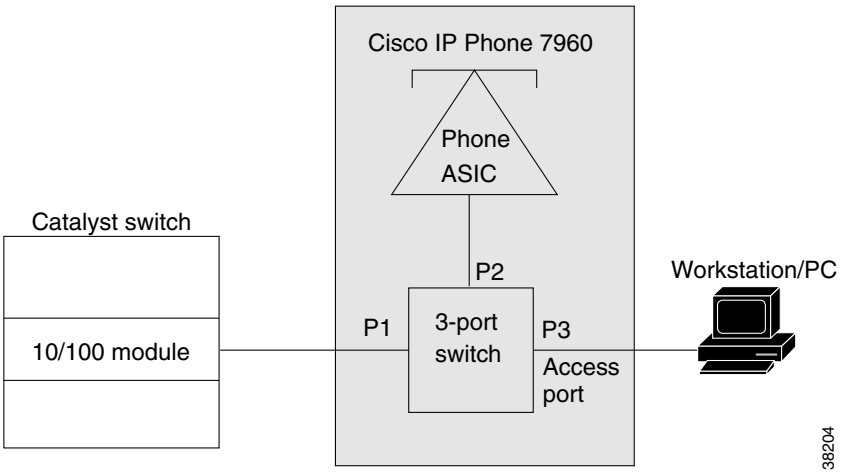
You can configure the Cisco IP Phone 7960 access port (see Figure 55-5) to either *trusted* or *untrusted* mode.

In untrusted mode, all the traffic in the 802.1Q or 802.1p frames that are received through the access port is marked with a configured Layer 2 CoS value. The default Layer 2 CoS value is 0. The untrusted mode is the default when the phone is connected to a Cisco LAN switch.

In trusted mode, all the traffic that is received through the access port passes through the phone switch unchanged. The trusted mode is the default when the phone is not connected to a Cisco LAN switch.

The traffic in the frame types other than 802.1Q or 802.1p passes through the phone switch unchanged, regardless of the access port trust state.

Figure 55-5 Configuring QoS on the IP Phone Ports



Configuring QoS in the Cisco IP Phone 7960

These sections describe how to configure QoS in the Cisco IP Phone 7960:

- Setting the Phone Access Port Trust Mode, page 55-32
- Setting the Phone Access Port CoS Value, page 55-33
- Verifying the Phone Access Port QoS Configuration, page 55-33

Setting the Phone Access Port Trust Mode

To set the phone access port trust mode, perform this task in privileged mode:

Task	Command
Set the phone access port trust mode.	**set port qos** *mod/ports...***trust-ext** {**trusted** | **untrusted**}

This example shows how to set the phone access port to the trusted mode:

```
Console> (enable) set port qos 3/7 trust-ext trusted
Port in the phone device connected to port 3/7 is configured to be trusted.
Console> (enable)
```

This example shows how to set the phone access port to the untrusted mode:

```
Console> (enable) set port qos 3/7 trust-ext untrusted
Port in the phone device connected to port 3/7 is configured to be untrusted.
Console> (enable)
```

Setting the Phone Access Port CoS Value

To set the phone access port CoS value, perform this task in privileged mode:

Task	Command
Set the phone access port CoS value.	**set port qos** *mod/ports* **cos-ext** *cos_value*

This example shows how to set the Layer 2 CoS value that is used by a phone access port in untrusted mode:

```
Console> (enable) set port qos 2/1 cos-ext 3
Port 2/1 qos cos-ext set to 3.
Console> (enable)
```

Verifying the Phone Access Port QoS Configuration

To verify the phone access port QoS configuration, perform this task in normal mode:

Task	Command
Verify the phone access port QoS configuration.	**show port qos** [*mod*[*/port*]]

This example shows how to verify the phone access port QoS configuration:

```
Console> (enable) show port qos 3/4
<...Output Truncated...>
Port  Ext-Trust Ext-Cos
----- --------- -------
 3/4  untrusted       0
<...Output Truncated...>
```

Configuring a Trusted Boundary to Ensure Port Security

This section describes the trusted boundary that is used to prevent security problems if users disconnect their PCs from the networked Cisco IP Phones and plug them directly into the switch port to take advantage of the QoS **trust-cos** switch port settings.

These sections describe the trusted boundary:

Supported Cisco IP Phones

These Cisco IP phones are supported with the trusted boundary feature:

- Cisco IP Phone 7910
- Cisco IP Phone 7935
- Cisco IP Phone 7940
- Cisco IP Phone 7960

QoS and Cisco IP Phone Configuration

The Cisco IP Phones are directly attached to the Catalyst 6500 series switch ports. Typically, the traffic that is coming from the phone and entering the switch is marked with a tag using the 802.1Q header. The header contains the VLAN information and the class of service (CoS) 3-bit field. The CoS determines the priority of the packet. For most Cisco IP Phone configurations, the traffic that comes from the phone and enters the switch is trusted to ensure that the voice traffic is properly prioritized over other types of traffic in the network. The port on the switch where the phone is attached is configured to **trust-cos**, which means that the port trusts the CoS labeling of all packets arriving on that port.

QoS, Cisco IP Phone, and PC Configuration

A PC or workstation can be attached to the Cisco IP Phone. The phone has a built-in hub that mixes the traffic coming from the PC, the phone, and the switch port. To distinguish the traffic that comes from the PC from the traffic that comes from the phone, use the 3-bit CoS labels.

You need to configure the QoS features on the phone for proper labeling to occur. The QoS configuration information is sent to the phone using CDP from the switch. The QoS configuration determines the trust state of the phone and the classification information (Ext-Cos). The phone supports two trust states:

- Trusted
- Untrusted and marked with a new COS value (Ext-Cos)

If the phone is in trusted mode, all the labels that are produced by the PC are sent directly through the phone toward the switch, untouched. If the phone is in untrusted mode, all traffic coming from the PC is marked with the Ext-Cos value before it is sent to the switch.

For most setups, the PC or workstation that is attached to the phone is unable to tag its packets. In these cases, all the traffic that comes from the PC and enters the switch through the phone, is marked with the "default ext-cos" that is configured on the phone.

In some cases, the PC can tag its own packets. A PC running Windows 2000 can be configured to send the 802.1Q frames of any priority. To solve this problem, the phones should be configured to be untrusted, which marks all the traffic coming from the PC to the appropriate priority.

The trusted boundary prevents the users from taking advantage of the trust-cos setting on the switch by disconnecting their phone from the network and plugging their PC directly into the switch port. It uses CDP to detect the phone's presence on a port. If the phone leaves the port, the feature automatically configures the port to be untrusted, which solves the security issue.

The trusted boundary is implemented using a configuration command to create a new type of trust. The command allows you to configure the port trust based on the presence of a given device on a port. For the Cisco IP Phones, you configure the trust as "**trust-device ciscoipphone**."

Trusted Boundary Configuration Guidelines

This section describes the guidelines for configuring the trusted boundary:

- Common Open Policy Service (COPS) considerations

  COPS directly affects how the QoS parameters are applied. A port may have either a local policy or a COPS policy. This setting specifies whether the port should get its QoS configuration information from the local configuration or through a COPS server. If COPS is enabled on a port and is also globally enabled, the policy that is specified by the COPS server applies. If COPS is disabled and/or the run-time policy is local, the local configuration QoS policy applies. The extended trust boundary feature overrides the "local" policy on a port.

- QoS configuration support

  All the QoS port trust configuration settings are supported (**trust-cos**, **trust-ipprec**, **trust-dscp**), but you should use **trust-cos** for the Cisco IP Phone networks.

- System log messaging

  New QoS syslogs were added for the trusted boundary to notify you of the changes to a port's trust state and to warn of improper configuration. To see these syslogs, set the QoS logging level to 5 (**set logging level qos 5**). The default is 3. Refer to the *Catalyst 6500 Series System Message Guide* for the descriptions of the syslogs.

- Final run-time port trust value

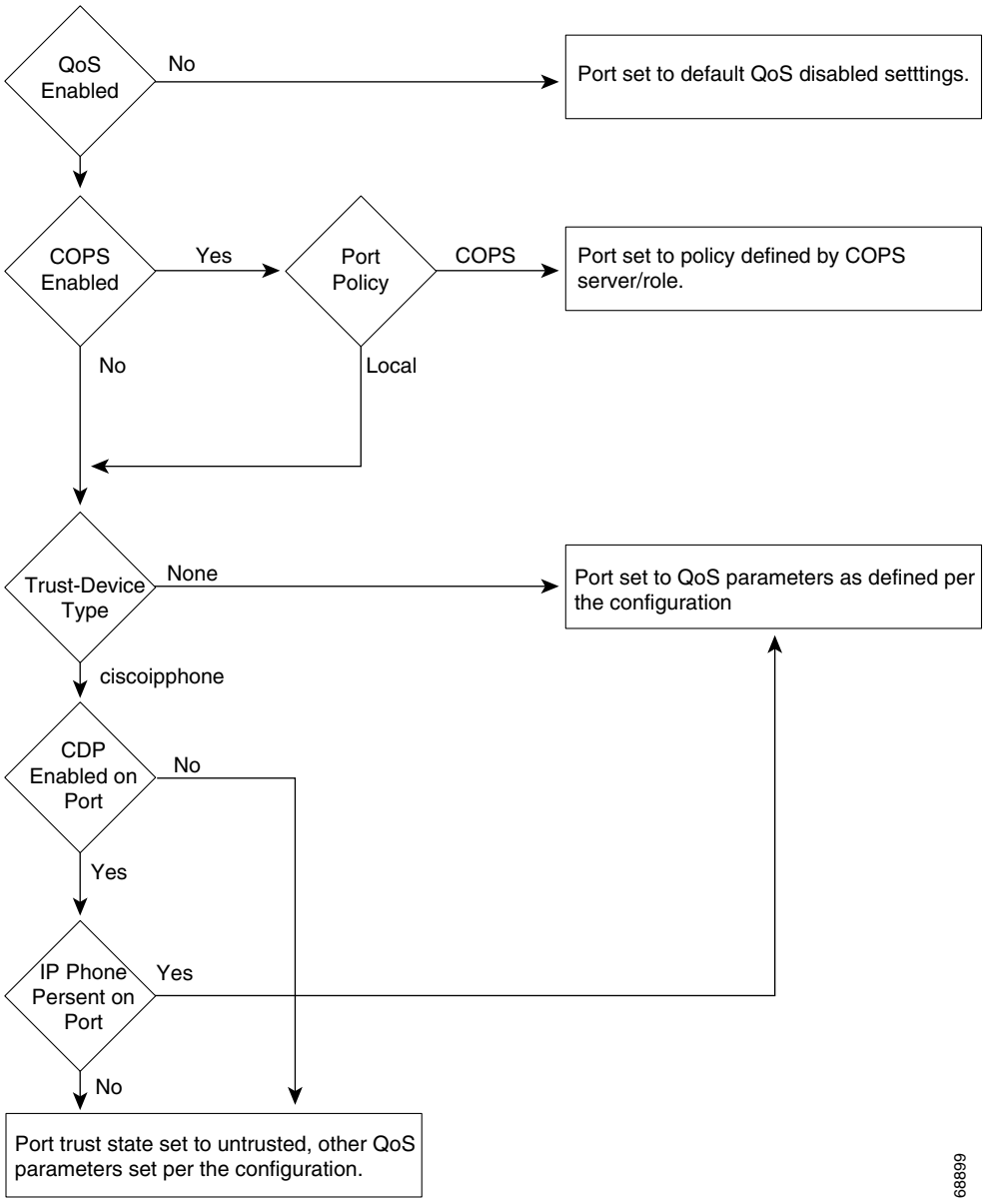
  The final run-time port trust on any port is dependent on the following:

  - Trusted boundary configuration
  - Phone's presence on the port
  - QoS configuration
  - COPS configuration

  To enable the trusted boundary, you must enable QoS and you must enable CDP globally and on the port, running in version 2 mode. You must set COPS to local policy (the COPS default) or to disabled (the COPS default). When **ciscoipphone** is configured as the trust-device on the port, the feature is enabled and detects the presence of a Cisco IP Phone and sets the trust values.

  See Figure 55-6 to determine the final trust value on a port.

Figure 55-6 Determining the Final Trust Value of a Port



Configuring a Trusted Boundary

These sections describe how to configure the trusted boundary feature:

- Default Configuration, page 55-37
- Specifying a Cisco IP Phone as the Trust Device, page 55-37
- Verifying a Port's Trust-Device State, page 55-37

Default Configuration

The default setting for all ports is **trust-device none**.

Specifying a Cisco IP Phone as the Trust Device

To specify a Cisco IP Phone as the trust device, perform this task in privileged mode:

Task	Command
Specify a Cisco IP Phone as the trust device.	**set port qos** *mod/ports...***trust-device** [**ciscoipphone** | **none**]

This example shows how to trust only Cisco IP phones on port 4/1:

```
Console> (enable) set port qos 4/1 trust-device ciscoipphone
Port 4/1 set to only trust device of type ciscoIPPhone.
Console> (enable)
```

This example shows how to disable the device trust on port 4/1:

```
Console> (enable) set port qos 4/1 trust-device none
Port 4/1 trust device feature disabled.
Console> (enable)
```

Verifying a Port's Trust-Device State

To verify a port's trust-device state, perform this task in normal mode:

Task	Command
Verify a port's trust-device state.	**show port qos** [*mod*[/*port*]]

When the trusted boundary is active, the run-time trust state of the port changes depending on the presence of the phone.

Note The moment that the phone leaves the switch port, there is a slight convergence time for the port to change to the untrusted state (a maximum time of 15 seconds).

This example shows how to verify the trust-device state and trust state on port 4/1:

```
Console> (enable) show port qos 4/1

<truncated ...>

Port  TxPort Type  RxPort Type  Trust Type   Trust Type    Def CoS Def CoS
                                config       runtime       config  runtime
----- ------------ ------------ ------------ ------------- ------- -------
 4/1        1p3q1t       1p1q0t    trust-cos    trust-cos*       0       0

Port  Ext-Trust Ext-Cos Trust-Device

----- --------- ------- ------------

 4/1  untrusted       0 ciscoIPPhone
```

```
(*)Runtime trust type set to untrusted.

Config:
Port  ACL name                          Type
----- -------------------------------- ----
No ACL is mapped to port 4/1.

Runtime:
Port  ACL name                          Type
----- -------------------------------- ----
No ACL is mapped to port 4/1.
Console> (enable)
```

Using SmartPorts

The SmartPorts feature consists of two macros that simplify voice configuration on the Catalyst 6500 series switches. The SmartPorts macros cover all the voice configuration tasks that are required for implementing the recommended Architecture for Voice, Video, and Integrated Data (AVVID) settings for a voice port.

SmartPorts focuses on the voice networks that are built using the Cisco IP Phone 79xx series and the Cisco SoftPhone. With SmartPorts, you use the **ciscoipphone** or **ciscosoftphone** keywords to initiate the macros that specify the type of voice parameters that you desire on a particular port.

SmartPorts is described in these sections:

Understanding SmartPorts Macros

When you execute the SmartPorts macros on a port using the **ciscoipphone** or **ciscosoftphone** keywords, these features are implemented:

- The port is enabled.
- The Layer 2 protocol is disabled for CDP, STP, and VTP.
- The port membership is set to “static.”
- The **set port host** command is executed on the port.
- The specified data VLAN is associated with the port.
- The global automatic QoS command is executed.

When you execute the **ciscoipphone** keyword on a port, in addition to the previous features, these features are also implemented:

- The specified auxiliary VLAN is associated with the port.
- The inline power is enabled.
- CDP is enabled globally and on the port.
- CDP is configured to version v2.
- The port-based automatic QoS command for the Cisco IP phone is executed.

When you execute the **ciscosoftphone** keyword on a port, in addition to the previous features, these features are also implemented:

- The auxiliary VLAN for the port is set to "none."
- The port-based automatic QoS command for the Cisco SoftPhone is executed.

SmartPorts—Cisco IP Phone

In most configurations, the Cisco IP Phone 79xx is connected directly to the Catalyst switch port. Optionally, you can attach a PC to the phone and use the phone as a hop to the switch.

Typically, the traffic that comes from the phone and enters the switch is marked with a tag using the 802.1Q/p header. The header contains the VLAN information and the CoS 3-bit field. The CoS determines the priority of the packet. The switch uses the CoS field to distinguish the PC traffic from the phone traffic. The switch can also use the DSCP field for the same purpose.

In most Cisco IP Phone 79xx configurations, the traffic that comes from the phone and enters the switch is trusted. You set the port trust to trust-cos to properly prioritize the voice traffic over other types of traffic in the network.

The Cisco IP Phone 79xx has a built-in switch that mixes the traffic that comes from the PC, the phone, and the switch port. The Cisco IP Phone 79xx has the trust and classification capabilities that you need to configure.

The ports that connect the IP phones need to have several features enabled or disabled. SmartPorts ensures that the necessary features are enabled. Most of these features are implemented when you execute the **set port host** command (such as disabling channels, enabling PortFast, and so on). A VLAN and an auxiliary VLAN must be configured on the port for QoS to work. The inline power needs to be enabled (if available), and CDP must be enabled for the trusted boundary feature to work. QoS configuration is handled by the automatic QoS feature (see Chapter 53, "Using Automatic QoS").

SmartPorts—Cisco Softphone

The Cisco SoftPhone is a software product that runs on a standard PC and emulates an IP phone. The main difference between the Cisco SoftPhone and the Cisco IP Phone 79xx is that the Cisco SoftPhone marks its voice traffic through a DSCP, while the Cisco IP Phone 79xx marks its traffic through a CoS. The QoS settings on the switch accommodate this behavior by trusting the Layer 3 marking of the traffic entering the port. All other behavior is similar to the Cisco IP Phone 79xx. Some features, such as CDP, do not need to be enabled because the trusted boundary does not support Cisco SoftPhone.

SmartPorts Guidelines and Restrictions

These sections provide the configuration guidelines and restrictions for SmartPorts:

Supported Phones

When you use SmartPorts with the **ciscoipphone** keyword, some of the QoS configuration requires phone-specific configuration (trust-ext, ext-cos) which is supported only on the following phones: Cisco IP Phone 7910, Cisco IP Phone 7940, Cisco IP Phone 7960, and Cisco IP Phone 7935. However, the **ciscoipphone** keyword is not exclusive to these models only; any phone can benefit from all the other QoS settings that are configured on the switch.

The Cisco SoftPhone is supported through the **ciscoipsoftphone** keyword.

CDP Dependencies

To configure the QoS settings and the trusted boundary on the Cisco IP Phone, you must enable CDP version 2 or later on the port.

You need to enable CDP only for the **ciscoipphone** QoS configuration; CDP does not affect the other components of the SmartPorts feature.

EtherChannel Considerations

The SmartPorts commands do not support channeling.

PFC/PFC2 Support

No PFC or PFC2 is required for the **ciscoipphone** keyword. A PFC or PFC2 is required for the **ciscosoftphone** keyword.

Module Support

The **ciscoipphone** keyword is supported only on the 10/100 and 10/100/1000 Ethernet ports.

The **ciscosoftphone** keyword is supported on all Ethernet ports.

CLI Interface for SmartPorts

These sections describe the CLI interface for SmartPorts:

Command Description

You must specify either the **ciscoipphone** or **ciscosoftphone** keywords and a data VLAN. Specifying an auxiliary VLAN is optional for the **ciscoipphone** keyword. The RSPAN and private VLANs are not supported. The command syntax for SmartPorts is as follows:

```
Console> (enable) set port macro
Usage: set port macro <mod/ports..> ciscoipphone vlan <vlan> [auxvlan <auxvlan>]
       set port macro <mod/ports..> ciscosoftphone vlan <vlan>
Console> (enable)
```

Note The **set port macro** *mod/ports...* **ciscoipphone vlan** *vlan* [**auxvlan** *auxvlan*] command enables the "cdpverify" feature on the port.

ciscoipphone Command Output

When you enter the **ciscoipphone** keyword, the following displays (specifying the auxiliary VLAN is optional):

```
Console> (enable) set port macro 3/1 ciscoipphone vlan 2 auxvlan 3
Port 3/1 enabled.
Layer 2 protocol tunneling disabled for CDP STP VTP on port(s) 3/1.
Port 3/1 vlan assignment set to static.
Spantree port fast start option set to default for ports 3/1.
Port(s) 3/1 channel mode set to off.

Warning: Connecting Layer 2 devices to a fast start port can cause
temporary spanning tree loops. Use with caution.

Spantree port  3/1 fast start enabled.
Dot1q tunnel feature disabled on port(s)  3/1.
Port(s)  3/1 trunk mode set to off.
VLAN  Mod/Ports
---- -----------------------
2     2/1
      3/1
      16/1
AuxiliaryVlan Status   Mod/Ports
------------- --------
-------------------------------------------------------
3             inactive 3/1

Vlan 3 is not active.
Inline power for port 3/1 set to auto.

CDP enabled globally
CDP enabled on port 3/1.
CDP version set to v2
........
```

```
All ingress and egress QoS scheduling parameters configured on all ports.
CoS to DSCP, DSCP to COS, IP Precedence to DSCP and policed dscp maps
configured.  Global QoS configured.
Port 3/1 ingress QoS configured for Cisco IP Phone.
Macro completed on port 3/1.
Console> (enable)
```

If you do not specify an auxiliary VLAN, the following warning message displays:

```
Console> (enable) set port macro 3/1 ciscoipphone vlan 2
Warning:  All inbound QoS tagging information will be lost as no auxiliary
vlan was specified.
Do you want to continue (y/n) [n]?
```

ciscosoftphone Command Output

When you enter the **ciscosoftphone** keyword, the following displays:

```
Console> (enable) set port macro 3/1 ciscosoftphone vlan 32
Port 3/1 enabled.
Layer 2 protocol tunneling disabled for CDP STP VTP on port(s) 3/1.
Port 3/1 vlan assignment set to static.
Spantree port fast start option set to default for ports 3/1.
Port(s) 3/1 channel mode set to off.

Warning: Connecting Layer 2 devices to a fast start port can cause
temporary spanning tree loops. Use with caution.

Spantree port  3/1 fast start enabled.
Dot1q tunnel feature disabled on port(s)  3/1.
Port(s)  3/1 trunk mode set to off.
Vlan 32 configuration successful
VLAN 32 modified.
VLAN 2 modified.
VLAN  Mod/Ports
---- ------------------------
32    3/1
      16/1
Port 3/1 will not send out CDP packets with AuxiliaryVlan information.
Executing autoqos........
All ingress and egress QoS scheduling parameters configured on all ports.
CoS to DSCP, DSCP to COS, IP Precedence to DSCP and policed dscp maps
configured.  Global QoS configured.
Port 3/1 ingress QoS configured for Cisco Softphone.
Macro completed on port 3/1.
Console>> (enable)
```

Detailed SmartPorts Statements

These sections provide the detailed SmartPorts macro statements:

- ciscoipphone Macro Statement, page 55-43
- ciscosoftphone Macro Statement, page 55-43

ciscoipphone Macro Statement

The **ciscoipphone** macro command results in the following configuration:

```
set port macro mod/port ciscoipphone vlan vlan [auxvlan auxvlan]
------------------------------------------------------------
set port enable mod/port
set port l2protocol-tunnel mod/port cdp stp vtp disable
set port membership mod/port static
set port host mod/port
set vlan mod/port vlan
set port auxiliaryvlan mod/port auxvlan  (set to none if not specified)
set port inlinepower mod/port auto (if supported by module)
set cdp enable
set cdp enable mod/port
set cdp version v2
set qos autoqos
set port qos mod/port autoqos voip ciscoipphone
```

ciscosoftphone Macro Statement

The **ciscosoftphone** macro command results in the following configuration:

```
set port macro mod/port ciscosoftphone vlan vlan
------------------------------------------------------------
set port enable mod/port
set port l2protocol-tunnel mod/port cdp stp vtp disable
set port membership mod/port static
set port host mod/port
set vlan mod/port vlan
set port auxiliaryvlan mod/port none
set qos autoqos
set port qos mod/port autoqos voip ciscosoftphone
```

How to Use SmartPorts in Your Network

Depending on the interface and what is connected to it, you need to execute different automatic voice macros. For each port, enter the port-based macro command with the appropriate keyword as shown in Table 55-8.

Table 55-8 Using Automatic Voice Configuration Keywords

Keyword	Port Type
ciscoipphone	Ports that connect only a Cisco IP Phone 79xx.
ciscoipphone	Ports that connect a Cisco IP Phone 79xx with a PC connected to the 79xx.
ciscoipphone	Ports that connect a Cisco IP Phone 79xx with a PC connected to the 79xx running Cisco SoftPhone[1].
ciscosoftphone	Ports that connect a PC running Cisco SoftPhone without a Cisco IP Phone 79xx.

1. For cases where the ports connect a Cisco IP Phone 79xx with a PC running Cisco SoftPhone, the control traffic through CTI communication with the Cisco CallManager is tagged but is remarked to DSCP 0.

SmartPorts Enhancements in Software Release 8.4(1)

These sections describe the SmartPorts enhancements in software release 8.4(1):

Ciscorouter SmartPorts Template

The **ciscorouter** interface macro command results in the following configuration:

Note Specifying the **nativevlan** is required. Specifying the **allowedvlans** is optional.

```
set port macro mod/port ciscorouter nativevlan nativevlan allowedvlans vlans
--------------------------------------------------------------------------------
set port enable mod/port
set vlan nativevlan mod/port
set port auxiliaryvlan mod/port auxvlan none
set port inlinepower mod/port auto
set cdp enable mod/port
set port membership mod/port static
set port l2protocol-tunnel mod/port cdp stp vtp dis
set udld enable mod/port
set spantree portfast mod/port enable trunk
set spantree bpdu-guard mod/port enable
set trunk mod/port nonegotiate dot1q
```

If the **allowedvlans** parameter is not specified, the following configuration is used:

```
set trunk mod/port 1-4094 (if all specified)
```

If the **allowedvlans** parameter is specified, the following configuration is used:

```
set trunk mod/port none
set trunk mod/port vlans (if specified)

set port qos mod/port autoqos trust dscp
```

Ciscoswitch SmartPorts Template

The **ciscoswitch** interface macro command results in the following configuration:

Note Specifying the **nativevlan** is required. Specifying the **allowedvlans** is optional.

```
set port macro mod/port ciscoswitch nativevlan nativevlan allowedvlans vlans
--------------------------------------------------------------------------------
set port enable mod/port
set vlan nativevlan mod/port
set port auxiliaryvlan mod/port auxvlan none
set port inlinepower mod/port auto
set cdp enable mod/port
set port membership mod/port static
set port l2protocol-tunnel mod/port cdp stp vtp dis
set udld enable mod/port
set spantree portfast mod/port disable
set spantree bpdu-guard mod/port disable
set spantree link-type mod/port point-to-point
set trunk mod/port nonegotiate dot1q
```

If the **allowedvlans** parameter is not specified, the following configuration is used:

```
set trunk mod/port 1-4094 (if all specified)
```

If the **allowedvlans** parameter is specified, the following configuration is used:

```
set trunk mod/port none
set trunk mod/port vlans (if specified)

set port qos mod/port autoqos trust dscp
```

Ciscodesktop SmartPorts Template

The **ciscodesktop** interface macro command results in the following configuration:

Note Specifying the **vlan** is required.

```
set port macro mod/port ciscodesktop vlan vlan
------------------------------------------------------
set port enable mod/port
set port host mod/port
      set vlan vlan mod/port
set port auxiliaryvlan mod/port auxvlan none
set port inlinepower mod/port auto
set cdp enable mod/port
set port membership mod/port static
set port l2protocol-tunnel mod/port cdp stp vtp dis
set spantree bpdu-guard mod/port enable
set port security mod/port enable age 2 maximum 1
                   violation restrict
set port qos mod/port autoqos trust dscp
set port qos mod/port trust untrusted
```

Ciscoipphone SmartPorts Template

The **ciscoipphone** interface macro command results in the following configuration:

Note Specifying the **vlan** (*nativevlan)* is required. Specifying the **auxvlan** is optional. The port security is set to the maximum of 3 for the IP phone because the phone's MAC address can appear in both the native and the auxiliary VLAN.

```
set port macro mod/port ciscoipphone vlan nativevlan auxvlan auxvlan
------------------------------------------------------------------
set port enable mod/port
set port l2protocol-tunnel mod/port cdp stp vtp dis
set port membership mod/port static
set port host mod/port
set spantree bpdu-guard mod/port enable
set vlan nativevlan mod/port
set port auxiliaryvlan mod/port auxvlan  (set to none if not specified)
set port inlinepower mod/port auto (if supported by module)
set cdp enable mod/port
set port security mod/port enable age 2 maximum 3 violation restrict
set port qos mod/port autoqos voip ciscoipphone
```

Ciscosoftphone SmartPorts Template

The **ciscosoftphone** interface macro command results in the following configuration:

Note Specifying the **vlan** (*nativevlan)* is required.

```
set port macro mod/port ciscosoftphone vlan nativevlan
------------------------------------------------------------------
set port enable mod/port
set port l2protocol-tunnel mod/port cdp stp vtp dis
set port membership mod/port static
set port host mod/port >
set spantree bpdu-guard mod/port enable
set vlan nativevlan mod/port
set port auxiliaryvlan mod/port auxvlan  none
set port inlinepower mod/port auto
set cdp enable mod/port
set port security mod/port enable age 2 maximum 1 violation restrict
set port qos mod/port autoqos voip ciscosoftphone
```

Global SmartPorts Template

The **ciscosmartports** global macro command results in the following configuration:

```
set macro ciscosmartports
----------------------------------------------------
set udld enable
set errdisable-timeout enable udld
set errdisable-timeout enable duplex-mismatch
set errdisable-timeout enable channel-misconfig
set errdisable-timeout enable bpdu-guard
set errdisable-timeout interval 60
set cdp enable
set cdp version v2
set spantree mode rapid-pvst+
set spantree macreduction enable
set spantree portfast bpdu-guard enable
set spantree global-default loop-guard enable
set qos autoqos
```

Configuring User-Definable SmartPorts Macros

These sections describe how to define and implement SmartPorts macros:

Overview

This section describes the user-definable SmartPorts macros:

- Creating a macro—The user-definable macro approach is similar in concept to the **alias** command. The **alias** command is an alias for only one command; the user-definable macro approach creates a command set macro for one or more commands. The macros are created using the **set macro name** *name* command after which you enter a list of commands that become part of the macro.
- Creating variables for macros—When defining macros, some commands require parameters that need to be specified by variables (such as the VLAN ID for Ethernet ports or the IP address for ACLs). The variables are defined as "'keyword-value" pairs, where the first parameter must be the name of the variable and the second parameter is its value. Each variable can be defined on a per-port or global basis. The variables are created using the **set macro variable** *name_of_variable variable_value mod/port* command. The variables and their values are stored in the switch in a table/database. When a macro with a variable in its definition is applied to a port, the macro takes the values from the table/database and executes the commands in the macro.
- Displaying macros and variable definitions—To display macros and their variable definitions, enter the **show macro** *macro-name* command and the **show macro variable** [**all**] [**name** *name_of_macro*] [*mod/port*] command.
- Applying a macro—After you create a macro, it needs to be applied to a port. When the macro is applied to a port, if the macro contains any variables, the variables are replaced with the respective values that are predefined in the table/database, and then the commands in the macro definition are executed. To apply a macro to a port, enter the **set port macro** *mod/port name_of_macro* command.

- Clearing (deleting) a macro—You can clear a macro when it is no longer needed. When you clear a macro, only the macro and its definition are cleared from the system; the configuration on the ports that the macro was applied to is not cleared. To clear a macro, enter the **clear macro** *name* command.
- Types of macros—The two types of macros are the global macros and the port-based macros.

Using the CLI to Configure User-Definable SmartPorts Macros

These sections describe how to use the CLI to configure user-definable SmartPorts macros:

Creating User-Defined Macros

To create (define) a macro, use the **set macro name** *name* command to enter a list of commands (one command per line). To end the macro and exit from the macro mode, type the @ break character and then press Enter. An example is as follows:

```
Console> (enable) set macro name videophone
Enter macro commands one per line. End with character '@'.
set port enable #MODPORT
set vlan $DATAVLAN #MODPORT
set port auxiliaryvlan #MODPORT $AUXVLAN
set qos autoqos
@
Console> (enable)
```

Follow these guidelines and restrictions when creating user-defined macros:

- The maximum length of a macro name is 16 characters. The maximum number of command lines in a macro is 64. A macro cannot have the same name as a static macro (such as ciscoswitch or ciscorouter).
- You can have a macro inside a macro in user-defined and static macros.
- Syntax checking is not done when you create or modify a macro. If you enter incorrect commands when creating the macro, the incorrect commands fail when the macro is applied to a port.
- In the above example, #MODPORT is a variable that specifies the port to which the macro is applied. If the macro is applied on port 3/2, then #MODPORT is replaced by 3/2 when the macro is applied to a port.

- In the above example, $DATAVLAN and $AUXVLAN are variables and are substituted with appropriate values when the macro is applied to a port.
- After the macro is defined, it is stored in NVRAM.

Modifying Existing User-Defined Macros

To modify an existing user-defined macro, use the **set macro name** *name* command. When modifying a macro, the new definition replaces the old definition but the new definition is not automatically applied to all the ports on which it was previously applied. You need to explicitly apply the modified macro. An example is as follows:

```
Console> (enable) set macro name fileserver
Enter macro commands one per line. End with the character '@'.
cmd1
cmd2
@
Console> (enable)
```

The macro named “fileserver” can be overwritten by creating a macro with the same name and new definitions. An example is as follows:

```
Console> (enable) set macro name fileserver
Enter macro commands one per line. End with the character '@'.
cmd2
cmd3
@
Warning: The macro fileserver has been modified; Do you want to modify (y/n) y
Console> (enable)
```

Defining Variables

To define a variable, use the **set macro variable** *name_of_variable variable_of_value* [*mod/port*] command. You can define the variable on a per-port basis or a global basis. When a macro is applied to a port, the variables are replaced with the values that you have defined. The maximum length of a variable name is 16 characters. A macro definition can use multiple variables in a single line. Per-port variables are defined on a per-port basis. Individual ports can be configured with different values by defining variables with different values for different ports. If a variable definition does not have port information, then it is treated as a global variable. The global variable definition is used if the per-port variable is not defined. An example is as follows:

```
Console> (enable) set macro variable $DATAVLAN 3 3/2

Variable DATAVLAN successfully created
Console> (enable) set macro variable $DATAVLAN 5 3/3
Console> (enable) set macro variable $AUXVLAN 4 3/2

Variable AUXVLAN successfully created
Console> (enable)
```

If a port is not specified in the variable definition, the variable is considered a global variable. An example is as follows:

```
Console> (enable) set macro variable $CDPVER v2

Variable CDPVER successfully created
Console> (enable)

Console> (enable) set macro variable $DATAVLAN 77
Console> (enable)
```

In the above examples, $CDPVER is a global variable and $DATAVLAN and $AUXVLAN are per-port variables. $DATAVLAN is also defined as a global variable. If a macro is using the variable $DATAVLAN and the macro is applied to a port other than ports 3/2 or 3/3, the macro would use the value of 77 for that port. After a variable and its values are defined, they are stored in NVRAM.

Using Special Variables

A macro could have a variable that is not predefined; the variable would get its value when the macro is applied. #MODPORT is one such variable. For example, assume that a macro has the variable #MODPORT in its definition. When the macro is applied on a module/port, the variable #MODPORT is replaced by the module/port (*mod/port*) on which the macro is applied. An example is as follows:

```
Console> (enable) set macro name videophone
Enter macro commands one per line. End with character @.
set port enable #MODPORT
set vlan $DATAVLAN #MODPORT
set port auxiliaryvlan #MODPORT $AUXVLAN
@
Console> (enable)
```

In the above example, #MODPORT is a special variable that gets its value when the macro videophone is applied on a port.

Note #MODPORT is currently the only special variable supported.

Applying a User-Defined Macro

After the macro is created, it can be applied to a port. When a macro is applied to a port, the commands in the macro definition are executed on the switch. If the commands in the macro definition use any variables, the variables are replaced by their respective user-defined values and then the commands are executed. Use the **set port macro** *mod/port name_of_macro* command to apply a macro to a port.

To create and execute a user-defined macro, perform these steps:

Step 1 Create the macro.

```
Console> (enable) set macro name videophone
Enter macro commands one per line. End with character @.
set port enable #MODPORT
set vlan $DATAVLAN #MODPORT
set port auxiliaryvlan #MODPORT $AUXVLAN
@
Macro videophone successfully created
Console> (enable)
```

Step 2 Define the macro variables.

```
Console> (enable) set macro variable $DATAVLAN 3 3/2

Variable DATAVLAN successfully created
Console> (enable) set macro variable $DATAVLAN 5 3/3
Console> (enable) set macro variable $AUXVLAN 4 3/2
```

```
Variable AUXVLAN successfully created
Console> (enable) set macro variable $AUXVLAN 77 3/7
Console> (enable) set macro variable $DATAVLAN 99
Console> (enable) set macro variable $CDPVER v2

Variable CDPVER successfully created
Console> (enable)
```

Step 3 Apply the macro on port 3/2.

```
Console> (enable) set port macro 3/2 videophone
```

Before the macro is applied, the $DATAVLAN and $AUXVLAN variables are replaced by "3" and "4," respectively, and then the following commands are executed:

```
set port enable 3/2
set vlan 3 3/2
set port auxiliaryvlan 3/2 4
set cdp enable
set cdp version v2
set qos autoqos
```

Step 4 Apply the macro on port 3/7.

```
Console> (enable) set port macro 3/7 videophone
```

Before the macro is applied, the $AUXVLAN variable is replaced by "77." $DATAVLAN is not defined for port 3/7, so the macro searches the list of global variables and finds $DATAVLAN. In this case, the $DATAVLAN variable is replaced by the global definition "99," and then the following commands are executed:

```
set port enable 3/7
set vlan 99 3/7
set port auxiliaryvlan 3/7 77
set cdp enable
set cdp version v2
set qos autoqos
```

Follow these guidelines and restrictions when applying user-defined macros:

- If you attempt to apply a macro on a port and the macro has a variable that is not defined in its definition, the macro is not applied on the port and an appropriate error message is displayed. This error response does not affect the definition of the macro.
- If you attempt to apply a macro on a port and the macro has some valid and some invalid commands in its definition, the macro is still applied on the port and an appropriate error message is displayed when the invalid command is executed. This error response does not affect the definition of the macro.
- When you apply a macro, a record of the macro being applied is not stored in the configuration file or NVRAM. However, each port has a record of the latest macro that was applied to it.
- Once a macro is applied to a port, you cannot clear the macro. However, one way to back out a macro on a port is to define another macro that clears the configurations on the port and then apply the newly created macro on the port.

Displaying Macros

This section describes the various methods of displaying macros:

- The syntax is as follows:

  show macro name *name_of_macro*

  show macro all

- Display the definition of a macro by entering the **show macro name** *name_of_macro* command as follows:

```
Console> (enable) show macro name videophone

The macro definition for videophone is:

set port enable #MODPORT
set vlan $DATAVLAN #MODPORT
set port auxiliaryvlan #MODPORT $AUXVLAN
Console> (enable)
```

- Display the names of all the macros in the switch by entering the **show macro all** command as follows:

```
Console> (enable) show macro all
Macro Names
-----------
fileserver
videophone
Console> (enable)
```

Displaying Macro Variables

This section describes the various methods of displaying macro variables:

- The syntax is as follows:

  show macro variable [**all**] [**name** *name_of_macro*] [*mod/port*]

  show macro variables name *name_of_macro mod/port*

- Display all the macro variables in the switch by entering the **show macro variable all** command as follows:

```
Console> (enable) show macro variable all

Variable                Port            Value           Type
--------                ----            -----           ------
DATAVLAN                3/2              3               Per-port
DATAVLAN                3/3              5               Per-port
DATAVLAN                NA               99              Global
AUXVLAN                 3/2              4               Per-port
AUXVLAN                 3/7              77              Per-port
CDPVER                  NA               v2              Global
Console> (enable)
```

- Display an individual macro variable and all of the ports that it is applied by entering the **show macro variable name** *name_of_macro* command as follows:

```
Console> (enable) show macro variable name $DATAVLAN

Variable                Port            Value           Type
--------                ----            -----           ------
DATAVLAN                3/2              3                Per-port
DATAVLAN                3/3              5                Per-portGlobal
DATAVLAN                NA               99               Global
Console> (enable)
```

- Display an individual macro variable and a specific port that it is applied by entering the **show macro variable name** *name_of_macro mod/port* command as follows:

```
Console> (enable) show macro variable name $DATAVLAN 3/2

Variable                Port            Value           Type
--------                ----            -----           ------
DATAVLAN                3/2              3                Per-port
Console> (enable)
```

- Display macro variables by macro name by entering the **show macro variables name** *name_of_macro mod/port* command as follows:

```
Console> (enable) show macro variables name videophone 3/2

Variable-Name                   Variable Value          Port
-------------                   --------------          -----
DATAVLAN                        3                       3/2
AUXVLAN                         4                       3/2
Console> (enable)
```

Clearing Macros and Macro Variables

When you clear a macro by entering the the **clear macro name** *name_of_macro* command, you clear the commands from the macro and remove the macro from the switch. The configurations that were applied using the macro that is being cleared are retained. If the macro that is being cleared is using any variables, and if the variables are not being used by any other macros, the variables are automatically cleared.

This section describes the various methods of clearing macros and macro variables:

- The syntax is as follows:

  clear macro name *name_of_macro*

  clear macro all

  clear macro variable [**all**] [*name_of_variable*] [*mod/ports*]

- Clear an individual macro and its variables by entering the the **clear macro name** *name_of_macro* command as follows:

```
Console> (enable) clear macro name videophone

Clearing macro videophone....
Cleared Macro videophone ....
Console> (enable)
```

- Clear all macros and their variables by entering the **clear macro all** command as follows:

```
Console> (enable) clear macro all

Clearing all macros....
All macros are cleared
Console> (enable)
```

- Clear an individual macro variable from all ports by entering the **clear macro variable** *name_of_variable* command as follows:

```
Console> (enable) clear macro variable $DATAVLAN

Clearing variable $DATAVLAN for all mod/ports...

Deleting Variable: DATAVLAN ...
Cleared variable DATAVLAN
Console> (enable)
```

- Clear an individual macro variable from a single port by entering the **clear macro variable** *name_of_variable mod/ports* command as follows:

```
Console> (enable) clear macro variable $AUXVLAN 3/7

Clearing variable $AUXVLAN for mod/port.3/7..
Console> (enable)
```

- Clear all macro variables from all ports as follows:

```
Console> (enable) clear macro variable all

Clearing all variables for all mod/ports...

All variables in the switch are cleared
Console> (enable)
```

Displaying Macro Port Mappings

This section describes the various methods of displaying macro port mappings:

- The syntax is as follows:

  show macro map [**all**] [**name** *name_of_macro*] [**port** *mod/port*]

- Display all macro port mappings by entering the **show macro map all** command as follows:

```
Console> (enable) show macro map all

Port                    Macro
-----                   -----
3/2                       videophone
3/7                       videophone
Console> (enable)
```

- Display the macro port mappings for a specific macro by entering the **show macro map name** *name_of_macro* command as follows:

```
Console> (enable) show macro map name videophone

Port                    Macro
-----                   -----
3/2                       videophone
3/7                       videophone
Console> (enable)
```

- Display the macro port mappings for a specific port by entering the **show macro map port** *mod/port* command as follows:

```
Console> (enable) show macro map port 3/2

Port                    Macro
-----                   -----
3/2                     videophone
Console> (enable)
```

Displaying the User-Definable SmartPorts Macro Configuration

The macro and variable definitions are stored in NVRAM and can be displayed by entering the **show config** command as follows:

```
Console> (enable) show config
This command shows non-default configurations only.
Use 'show config all' to show both default and non-default configurations.
.
.
.
.....................

begin
!
# ***** NON-DEFAULT CONFIGURATION *****
!
!
#time: Tue Mar 22 2005, 09:39:57
!
#version 8.5(0.52)JAC
!

!
#Macros
set macro name videophone

set port enable #MODPORT
set vlan $DATAVLAN #MODPORT
set port auxiliaryvlan #MODPORT $AUXVLAN
@
!
#Macro-Port mapping
set port macro 3/2 videophone
set port macro 3/7 videophone
!
.
.
.
```

Configuring a Macro within a Macro

You can have a macro within a macro definition. When the root macro is applied to a port, the macro inside the root macro gets replaced by its definition and the root macro is applied to the port. You can also have a static macro (such as ciscoswitch or ciscorouter) inside a user-defined macro definition.

Note If there is a macro inside a macro definition and if the root macro is applied on a port, the root macro is displayed by entering the **show macro map** commands.

CHAPTER 56

Configuring the MSFC Cisco IOS Features

This chapter describes the Cisco IOS features that are used with the Catalyst operating system to provide feature functionality and parity between these operating systems.

These sections describe the Cisco IOS features that are used with the Catalyst operating system:

IP-in-IP Tunneling

IP-in-IP tunneling allows a mobile host to move between networks without changing its IP address. IP-in-IP tunneling allows an IPv4 datagram to be encapsulated within another IPv4 datagram and carried as a payload to its destination. This IPv4 within IPv4 encapsulation is a type of Generic Routing Encapsulation (GRE) that is similar to GRE tunneling.

The PFC3 and DFC3s support the following tunnel commands:

- **tunnel destination**
- **tunnel mode gre**
- **tunnel mode ipip**
- **tunnel source**
- **tunnel ttl**
- **tunnel tos**

Other supported types of tunneling are run in the software on the MSFC3.

Enter the **tunnel ttl** command (default 255) to set the TTL of encapsulated packets.

Enter the **tunnel tos** command, if present, to set the ToS byte of a packet when it is encapsulated. If the **tunnel tos** command is not present and you do not enable QoS, the ToS byte of a packet sets the ToS byte of the packet when it is encapsulated. If the **tunnel tos** command is not present and you enable QoS, the ToS byte of a packet as modified by PFC QoS sets the ToS byte of the packet when it is encapsulated.

To configure GRE tunneling and IP-in-IP tunneling, refer to these URLs:

http://www.cisco.com/en/US/docs/ios/12_2/interface/configuration/guide/icflogin.html

To configure the **tunnel tos** and **tunnel ttl** commands, refer to this URL:

http://www.cisco.com/en/US/docs/ios/12_0s/feature/guide/12s_tos.html

IP-in-IP Configuration Guidelines

This section describes the guidelines for configuring IP-in-IP tunneling:

- Each hardware-assisted tunnel must have a unique source.
- Hardware-assisted tunnels cannot share a source even if the destinations are different.
- Use secondary addresses on loopback interfaces or create multiple loopback interfaces.
- Each tunnel interface uses one internal VLAN.
- Each tunnel interface uses one additional router MAC address entry per router MAC address.
- The PFC3A does not support any PFC QoS features on tunnel interfaces.
- The PFC3B and PFC3BXL support PFC QoS features on tunnel interfaces.
- The PFC3 does not support GRE tunnel encapsulation and deencapsulation of multicast traffic.
- The MSFC3 supports tunnels that are configured with egress features on the tunnel interface. Examples of egress features are output Cisco IOS ACLs, NAT and PAT (for inside-to-outside translation), TCP intercept, context-based access control (CBAC), and encryption.

WCCP

The Web Cache Communication Protocol (WCCP) allows you to redirect traffic to a cache engine (web caches) and manage cache engine clusters (cache farms).

Note

- Release 12.2(17d)SXB1 and later releases support WCCP on Supervisor Engine 2.
- Release 12.2(18)SXD1 and later releases support WCCP on Supervisor Engine 720.
- To use the WCCP Layer 2 PFC redirection feature, configure WCCP on the Catalyst 6500 series switch as described in this chapter and configure accelerated WCCP on the cache engine as described in the following publication:

  http://www.cisco.com/en/US/docs/app_ntwk_services/waas/acns/v42/configuration/guide/transprt.html
- A future release of Cisco Application and Content Networking System (ACNS) software, Release 4.2.2 and later releases support the **ip wccp** *service* **accelerated** command with a PFC2.

Because the WCCP service group list is scanned in the order in which service groups are created, not by priority, with multiple dynamic WCCP services defined the traffic that matches the selection criteria for more than one service group is not redirected to the service group with the highest priority. This problem is resolved in Release 12.2(18)SXE.

In Release 12.2(18)SXE where caveat CSCec55429 is resolved, after a number of Web Cache Communication Protocol (WCCP) “cache lost” and “cache found” events have occurred for all the caches in a service group, spurious memory accesses might occur, the addition and deletion of WCCP services might fail, and the **show ip wccp** command displays the WCCP service, but the output of the **show ip wccp** *service_number* command does not show the WCCP service. This problem is resolved in Release 12.2(18)SXE.

Configuring WCCPv2 on a Supervisor Engine 720 causes high CPU utilization. This problem is resolved in Release 12.2(18)SXD4.

Network Address Translation (NAT) does not work with WCCP configured. This problem is resolved in Release 12.2(18)SXD1.

WCCP-redirected packets that have no next-hop ARP cache entry are process switched to generate an ARP request, but because of the WCCP redirection, no ARP request is sent and the ARP cache is never populated for the next hop and subsequent WCCP-redirected packets continue to be process switched. This problem is resolved in Release 12.2(17d)SXB2.

For more information about Web Cache Control Protocol (WCCP) support with Supervisor Engine 720, refer to this URL:

http://www.cisco.com/en/US/docs/switches/lan/catalyst6500/ios/12.2SXF/native/configuration/guide/wccp.html

For more information about Web Cache Control Protocol (WCCP) that is supported only with Supervisor Engine 2, refer to this URL:

http://www.cisco.com/en/US/docs/switches/lan/catalyst6500/ios/12.2SXF/native/configuration/guide/wccp.html

APPENDIX A

Acronyms

Table A-1 defines the acronyms that are used in this publication.

Table A-1 List of Acronyms

Acronym	Expansion
AAA	authentication, authorization, accounting
AAL	ATM adaptation layer
ACE	access control entry
ACL	access control list
AFI	authority and format identifier
AMP	active monitor present
APaRT	automated packet recognition and translation
ARP	Address Resolution Protocol
ASLB	accelerated server load balancing
ATM	Asynchronous Transfer Mode
BES	bursty errored seconds
BIA	bottom interface adapter
BPDU	bridge protocol data unit
BRF	bridge relay function
BUS	broadcast and unknown server
CAM	content-addressable memory
CDP	Cisco Discovery Protocol
CEF	Cisco Express Forwarding
CIR	committed information rate
CLI	command-line interface
CMM	Communications Media Module
COPS	Common Open Policy Service
COPS-DS	COPS Differentiated Services
COPS-PR	COPS for Provisioning
CoS	class of service

Table A-1 List of Acronyms (continued)

Acronym	Expansion
CPLD	Complex Programmable Logic Device
CRAM	compression and reordering of the ACL masks
CRC	cyclic redundancy check
CRF	concentrator relay function
CSG	Content Services Gateway
DAI	Dynamic ARP Inspection
DCC	Data Country Code
DEC	Digital Equipment Corporation
DFI	Domain-Specific Part Format Identifier
DHCP	Dynamic Host Configuration Protocol
DISL	Dynamic Inter-Switch Link
DMP	data movement processor
DNS	Domain Name System
DRAM	dynamic RAM
DRiP	Dual Ring Protocol
DSAP	destination service access point
DSBM	Designated Subnet Bandwidth Manager
DSCP	differentiated services code point
DSP	digital signal processing or processor
DTP	Dynamic Trunking Protocol
EAP	Extensible Authentication Protocol
EARL	Enhanced Address Recognition Logic
EEPROM	electrically erasable programmable read-only memory
ESI	end-system identifier
FCS	frame check sequence
FEFI	far end fault indication
GARP	General Attribute Registration Protocol
GBIC	Gigabit Interface Converter
GMRP	GARP Multicast Registration Protocol
GSR	Gigabit Switch Router
GVRP	GARP VLAN Registration Protocol
HCRMON	High Capacity RMON
HDD	hard disk drive driver
HTTP	HyperText Transfer Protocol
ICD	International Code Designator
ICMP	Internet Control Message Protocol

Table A-1 List of Acronyms (continued)

Acronym	Expansion
IETF	Internet Engineering Task Force
IDP	initial domain part
IDSM	Intrusion Detection System Module
IGMP	Internet Group Management Protocol
ILMI	Integrated Local Management Interface
IP	Internet Protocol
IPC	interprocessor communication
IPX	Internetwork Packet Exchange
ISL	Inter-Switch Link
ISO	International Organization of Standardization
KDC	Key Distribution Center
LACP	Link Aggregation Control Protocol
LAN	local-area network
LANE	LAN Emulation
LCP	Link Control Protocol
LCV	line code violation seconds
LD	LocalDirector
LEC	LAN Emulation Client
LECS	LAN Emulation Configuration Server
LEM	link error monitor
LER	link error rate
LES	LAN Emulation Server or line errored seconds
LLC	logical link control
MAC	Media Access Control
MDG	multiple default gateway
MIB	Management Information Base
MII	media-independent interface
MISTP	Multi-Instance Spanning Tree Protocol
MLS	Multilayer Switching
MMLS	Multicast Multilayer Switching
MOP	Maintenance Operation Protocol
MOTD	message-of-the-day
MSFC	Multilayer Switch Feature Card
MSM	Multilayer Switch Module
MST	Multiple Spanning Tree
MTP	Media Termination Point

Table A-1 List of Acronyms (continued)

Acronym	Expansion
MTU	maximum transmission unit
MVAP	multiple VLAN access port
MVRP	multiple VLAN registration protocol
NAM	Network Analysis Module
NDE	NetFlow Data Export
NMP	Network Management Processor
NSAP	network service access point
NTP	Network Time Protocol
NVRAM	nonvolatile RAM
OAM	Operation, Administration, and Maintenance
OSI	Open System Interconnection
OUI	organizational unique identifier
PAE	port access entity
PAgP	Port Aggregation Protocol
PBF	policy-based forwarding
PCM	pulse code modulation
PCR	peak cell rate
PDP	policy decision point
PDU	protocol data unit
PEP	policy enforcement point
PFC	Policy Feature Card
PHY	physical sublayer
PIB	policy information base
PPP	Point-to-Point Protocol
PRID	policy rule identifiers
PROM	programmable read-only memory
PVID	port VLAN identifier
PVST+	per VLAN spanning tree
QoS	quality of service
RADIUS	Remote Access Dial-In User Service
RAM	random-access memory
rcp	Remote Copy Protocol
RGMP	Router-Ports Group Management Protocol
RIF	Routing Information Field
RMON	Remote Monitoring
ROM	read-only memory

Table A-1 List of Acronyms (continued)

Acronym	Expansion
RSA	Rivest, Shamir, and Adleman (a public-key cryptographic system)
RSPAN	remote SPAN
RST	reset
RSVP	ReSerVation Protocol
SAID	Security Association Identifier
SAP	service access point
SIMM	single in-line memory module
SLCP	Supervisor Line-Card Processor
SLIP	Serial Line Internet Protocol
SMP	standby monitor present
SMT	station management
SNAP	Subnetwork Access Protocol
SNMP	Simple Network Management Protocol
SPAN	Switched Port Analyzer
SRB	source-route bridging
SRT	source-route transparent bridging
SSH	Secure Shell
SSL	Secure Sockets Layer
SSLM	Secure Sockets Layer Module
STE	Spanning Tree Explorer
STP	Spanning Tree Protocol
SVC	switched virtual circuit
TAC	Technical Assistance Center (Cisco)
TACACS+	Terminal Access Controller Access Control System Plus
TCP/IP	Transmission Control Protocol/Internet Protocol
TFTP	Trivial File Transfer Protocol
TGT	ticket granting ticket
TLV	type-length-value
ToS	type of service
TrBRF	Token Ring Bridge Relay Function
TrCRF	Token Ring Concentrator Relay Function
TTL	time to live
UART	Universal Asynchronous Receiver/Transmitter
UDLD	UniDirectional Link Detection
UDLP	UniDirectional Link Protocol
UDP	User Datagram Protocol

Table A-1 List of Acronyms (continued)

Acronym	Expansion
UNI	User-Network Interface
UTC	Coordinated Universal Time
VACL	VLAN access control list
VCC	virtual channel connection (in ATM technology), virtual channel circuit
VCI	virtual circuit identifier
VCR	virtual configuration register
VID	VLAN ID
VIP	virtual IP address
VLAN	virtual LAN
VMPS	VLAN Membership Policy Server
VoIP	Voice over IP
VPN	Virtual Private Network
VPNSM	Virtual Private Network Services Module
VTP	VLAN Trunk Protocol
VVID	voice VLAN identifier
WRED	weighted random early detection
WRR	weighted round-robin

INDEX

Numerics

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Y