Cisco Industrial Ethernet 4000, 4010 and 5000 Switch Software Configuration Guide

First Published: September 2016

Last Updated: February 2024

Cisco Systems, Inc. www.cisco.com

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco website at www.cisco.com/go/offices.

Contents

Preface

Note: The documentation set for this product strives to use bias-free language. For purposes of this documentation set, bias-free is defined as language that does not imply discrimination based on age, disability, gender, racial identity, ethnic identity, sexual orientation, socioeconomic status, and intersectionality. Exceptions may be present in the documentation due to language that is hardcoded in the user interfaces of the product software, language used based on RFP documentation, or language that is used by a referenced third-party product.

Audience

This guide is for the networking professional managing your switch. Before using this guide, you should have experience working with the Cisco IOS software and be familiar with the concepts and terminology of Ethernet and local area networking.

Purpose

This guide provides the information that you need to configure Cisco IOS software features on your switch.

This guide provides procedures for using the commands that have been created or changed for use with the switch. It does not provide detailed information about these commands.

For information about the standard Cisco IOS commands, see the Cisco IOS 15.0 documentation set available from the Cisco.com home page.

This guide does not provide detailed information on the graphical user interfaces (GUIs) for the embedded Device Manager. However, the concepts in this guide are applicable to the GUI user. For information about Device Manager, see the switch online help.

For documentation updates, see the release notes for this release.

Conventions

This publication uses these conventions to convey instructions and information:

Command descriptions use these conventions:

- Commands and keywords are in **boldface** text.
- Arguments for which you supply values are in *italic*.
- Square brackets ([ ]) mean optional elements.
- Braces ({ }) group required choices, and vertical bars (|) separate the alternative elements.
- Braces and vertical bars within square brackets ([{ | }]) mean a required choice within an optional element.

Interactive examples use these conventions:

- Terminal sessions and system displays are in `screen` font.

- Information you enter is in **`boldface screen`** font.
- Nonprinting characters, such as passwords or tabs, are in angle brackets (< >).

Notes, cautions, and timesavers use these conventions and symbols:

Note: Means *reader take note*. Notes contain helpful suggestions or references to materials not contained in this manual.

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

Related Publications

These documents provide complete information about the switch series and are available from this Cisco.com site:

http://www.cisco.com/c/en/us/support/switches/industrial-ethernet-4000-series-switches/tsd-products-support-series-home.html

http://www.cisco.com/c/en/us/support/switches/industrial-ethernet-4010-series-switches/tsd-products-support-series-home.html

http://www.cisco.com/c/en/us/support/switches/industrial-ethernet-5000-series-switches/tsd-products-support-series-home.html

Before installing, configuring, or upgrading the switch, see these documents:

- For initial configuration information, see the “Configuring the Switch with the CLI-Based Setup Program” appendix in the hardware installation guide.
- For Device Manager requirements, see the “System Requirements” section in the release notes (not orderable but available on Cisco.com).
- For upgrading information, see the “Downloading Software” section in the release notes.

See these documents for other information about the switch:

- Release Notes
- *Software Configuration Guide*
- *Hardware Installation Guide*
- *Regulatory Compliance and Safety Information*
- Device Manager online help (available on the switch)
- Compatibility matrix documents are available from this Cisco.com site: http://www.cisco.com/en/US/products/hw/modules/ps5455/products_device_support_tables_list.html

Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business impact you’re looking for with the technologies that matter, visit Cisco Services.
- To submit a service request, visit Cisco Support.

- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit Cisco Marketplace.
- To obtain general networking, training, and certification titles, visit Cisco Press.
- To find warranty information for a specific product or product family, access Cisco Warranty Finder.

Cisco Bug Search Tool

Cisco Bug Search Tool (BST) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.

Configuration Overview

Note: The documentation set for this product strives to use bias-free language. For purposes of this documentation set, bias-free is defined as language that does not imply discrimination based on age, disability, gender, racial identity, ethnic identity, sexual orientation, socioeconomic status, and intersectionality. Exceptions may be present in the documentation due to language that is hardcoded in the user interfaces of the product software, language used based on RFP documentation, or language that is used by a referenced third-party product.

Feature Availability

Unless otherwise indicated, all features and configurations in this guide are supported beginning with release 15.2(2)EA for the IE-4000, 15.2(2)EB for the IE-5000 and in release 15.2(4)EC for the IE-4010. Where new features or support for existing features was added after these releases, detailed release information will be indicated in the Feature History Table for that feature.

Feature availability varies depending on your license. For more information about licenses and available features, refer to the datasheet:
http://www.cisco.com/c/en/us/products/switches/industrial-ethernet-5000-series-switches/datasheet-listing.html

http://www.cisco.com/c/en/us/products/switches/industrial-ethernet-4010-series-switches/datasheet-listing.html

http://www.cisco.com/c/en/us/products/switches/industrial-ethernet-4000-series-switches/datasheet-listing.html

Feature Software Licensing

Software Licensing is now simplified with the introduction of right-to-use (RTU) licensing. This allows you to order and activate a specific license type and level via command line. Uploading an extra license file is no longer necessary.

Note: Upgrading to the IP Services feature set requires the purchase of one of the following licenses (product IDs listed): The IE-5000 uses "**L-IE5000-RTU=**" and IE-4000 and IE-4010 use "**L-IE4000-RTU=**" to upgrade to IP Services.

Right to Use Licenses

The introduction of right-to-use (RTU) licensing allows you to order and activate a specific license type and level via command line. Uploading an extra license file is no longer necessary.

LanBase images provide basic Layer2 functionality, including:

- QOS
- Port-Security
- 1588 PTP
- EtherNet/IP
- Profinet

IPService: L3 routing features:

- RIP
- OSPF
- ISIS BGP
- Policy-based routing
- IPV6

Defaults

The default license is a **lanbase RTU permanent license**.

Configuring RTU Licenses

To configure RTU Licenses, follow these guidelines.

Displaying License Information

To determine which license is running on your device, do the following:

- Enter the **show version** privileged EXEC command. The first line of output indicates the image, such as LANBASE.
- Enter the **show license** privileged EXEC command, to see which is the active image:

```
Switch# show license
Index 1 Feature: ipservices
      Period left: 8  weeks 4  days
      License Type: Evaluation
      License State: Active, Not in Use, EULA not accepted
      License Priority: None
      License Count: Non-Counted

Index 2 Feature: lanbase
      Period left: Life time
      License Type: PermanentRightToUse
      License State: Active, In Use
      License Priority: High
      License Count: Non-Counted

Index 3 Feature: mrp-manager
      Period left: 8  weeks 4  days
      License Type: Evaluation
      License State: Active, Not in Use, EULA not accepted
      License Priority: None
      License Count: 1/0/0  (Active/In-use/Violation)

Index 4 Feature: mrp-client
      Period left: 8  weeks 4  days
        License Type: Evaluation
      License State: Active, Not in Use, EULA not accepted
      License Priority: None
      License Count: 1/0/0  (Active/In-use/Violation)
License Count: Non-Counted
```

ipservices license

To activate a Permanent Right-To-Use ipservices license, use the following command:

```
IE5000#license right-to-use activate ipservices
PLEASE  READ THE  FOLLOWING TERMS  CAREFULLY. INSTALLING THE LICENSE OR
```

```
LICENSE  KEY  PROVIDED FOR  ANY CISCO  PRODUCT  FEATURE  OR  USING SUCH
PRODUCT  FEATURE  CONSTITUTES  YOUR  FULL ACCEPTANCE  OF  THE FOLLOWING
TERMS. YOU MUST NOT PROCEED FURTHER IF YOU ARE NOT WILLING TO  BE BOUND
BY ALL THE TERMS SET FORTH HEREIN.

Use of this product feature requires an additional license from Cisco, together
with an additional payment.  You may use this product feature subject to the
Cisco end user license agreement
http://www.cisco.com/en/US/docs/general/warranty/English/EU1KEN_.html,
together with any supplements relating to such product feature.
It is your responsibility to make payment to Cisco for your use of the
product feature if not already licensed to do so. Your acceptance
of this agreement for the software features on one product shall be deemed
your acceptance with respect to all such software on all Cisco products you
purchase which includes the same software.  (The foregoing notwithstanding, you must
purchase a license for each software feature you use, so that if you enable
a software feature on 1000 devices, you must purchase 1000 licenses for use.)
    This license may be transferrable from another Cisco device of the same model
for the same functionality if such license already is owned.
Activation of the software command line interface will be evidence of your acceptance
of this agreement.
ACCEPT? (yes/[no]): yes
Activated Permanent Right-To-Use ipservices license

Next Reboot level is ipservices

IE5000#
```

Ease-of-Deployment and Ease-of-Use Features

- Express Setup for quickly configuring a switch for the first time with basic IP information, contact information, switch and Telnet passwords, and Simple Network Management Protocol (SNMP) information through a browser-based program.
- User-defined and Cisco-default Smartports macros for creating custom switch configurations for simplified deployment across the network.
- A removable SD flash card that stores the Cisco IOS software image and configuration files for the switch. You can replace and upgrade the switch without reconfiguring the software features.
- An embedded Device Manager GUI for configuring and monitoring a single switch through a web browser. For more information about Device Manager, see the switch online help.

Performance Features

- Autosensing of port speed and autonegotiation of duplex mode on all switch ports for optimizing bandwidth
- Automatic medium-dependent interface crossover (auto-MDIX) capability on 10/100 and 10/100/1000 Mb/s interfaces and on 10/100/1000 BASE-TX SFP module interfaces that enables the interface to automatically detect the required cable connection type (straight-through or crossover) and to configure the connection appropriately
- Support for up to 1546 bytes routed frames, up to 9000 bytes for frames that are bridged in hardware, and up to 2000 bytes for frames that are bridged by software
- IEEE 802.3x flow control on all ports (the switch does not send pause frames)
- Support for up to 10 EtherChannel groups

- Port Aggregation Protocol (PAgP) and Link Aggregation Control Protocol (LACP) for automatic creation of EtherChannel links
- Per-port storm control for preventing broadcast, multicast, and unicast storms
- Port blocking on forwarding unknown Layer 2 unknown unicast, multicast, and bridged broadcast traffic
- Cisco Group Management Protocol (CGMP) server support and Internet Group Management Protocol (IGMP) snooping for IGMP Versions 1, 2, and 3:
  - (For CGMP devices) CGMP for limiting multicast traffic to specified end stations and reducing overall network traffic
  - (For IGMP devices) IGMP snooping for forwarding multimedia and multicast traffic
- IGMP report suppression for sending only one IGMP report per multicast router query to the multicast devices (supported only for IGMPv1 or IGMPv2 queries)
- IGMP snooping querier support to configure switch to generate periodic IGMP general query messages
- IGMP helper to allow the switch to forward a host request to join a multicast stream to a specific IP destination address
- IGMP filtering for controlling the set of multicast groups to which hosts on a switch port can belong
- IGMP throttling for configuring the action when the maximum number of entries is in the IGMP forwarding table
- IGMP leave timer for configuring the leave latency for the network
- Switch Database Management (SDM) templates for allocating system resources to maximize support for user-selected features such as lanbase-routing, ipv6 routing.
- Cisco IOS IP Service Level Agreements (SLAs), a part of Cisco IOS software that uses active traffic monitoring for measuring network performance
- Configurable small-frame arrival threshold to prevent storm control when small frames (64 bytes or less) arrive on an interface at a specified rate (the threshold)
- FlexLink Multicast Fast Convergence to reduce the multicast traffic convergence time after a FlexLink failure
- RADIUS server load balancing to allow access and authentication requests to be distributed evenly across a server group
- Support for QoS marking of CPU-generated traffic and queue CPU-generated traffic on the egress network ports

Management Options

- An embedded Device Manager—Device Manager is a GUI application that is integrated in the software image. You use it to configure and to monitor a single switch. For more information about Device Manager, see the switch online help.
- Network Assistant—Network Assistant is a network management application that can be downloaded from Cisco.com. You use it to manage a single switch, a cluster of switches, or a community of devices. For more information about Network Assistant, see *Getting Started with Cisco Network Assistant*, available at software.cisco.com/download/.
- Prime Infrastructure—Cisco Prime Infrastructure simplifies the management of wireless and wired networks. It offers Day 0 and 1 provisioning, as well as Day N assurance from the branch to the data center. We call it One Management. With this single view and point of control, you can reap the benefits of One Management across both network and compute.

- CLI—The Cisco IOS software supports desktop- and multilayer-switching features. You can access the CLI either by connecting your management station directly to the switch console port or by using Telnet from a remote management station.
- SNMP—SNMP management applications such as CiscoWorks2000 LAN Management Suite (LMS) and HP OpenView. You can manage from an SNMP-compatible management station that is running platforms such as HP OpenView or SunNet Manager. The switch supports a comprehensive set of MIB extensions and four remote monitoring (RMON) groups. For more information about using SNMP, see Configuring SNMP, page 557
- Cisco IOS Configuration Engine (previously known as the Cisco IOS CNS agent)—Configuration service automates the deployment and management of network devices and services. You can automate initial configurations and configuration updates by generating switch-specific configuration changes, sending them to the switch, executing the configuration change, and logging the results.

  For more information about CNS, see Configuring Cisco IOS Configuration Engine, page 79

Industrial Application

- CIP—Common Industrial Protocol (CIP) is a peer-to-peer application protocol that provides application level connections between the switch and industrial devices such as I/O controllers, sensors, relays, and so forth.You can manage the switch using RSlogix/RSlinx then monitor the CIP functionality via IOS command lines or Web based Device Manager.
- Profinet Version 2—Support for PROFINET IO, a modular communication framework for distributed automation applications. The embedded Profinet GSD file allows user to bring up Cisco IE switch using Siemens STEP7 or TIA Portal software then monitor the functionality via command line or Web based Device Manger.

Default Settings After Initial Switch Configuration

The switch is designed for plug-and-play operation, requiring only that you assign basic IP information to the switch and connect it to the other devices in your network. If you have specific network needs, you can change the interface-specific and system-wide settings.

Note: For information about assigning an IP address by using the CLI-based setup program, see the hardware installation guide.

If you do not configure the switch at all, the switch operates with these default settings:

Note: For more information about the following default settings, see the corresponding sections of this guide.

- Default switch IP address, subnet mask, and default gateway is 0.0.0.0.
- Default domain name is not configured.
- DHCP client is enabled, the DHCP server is enabled, and the DHCP relay agent is enabled.
- Switch cluster is disabled.
- No passwords are defined.
- System name and prompt is Switch.
- NTP is enabled.
- DNS is enabled.
- TACACS+ is disabled.

- RADIUS is disabled.
- The standard HTTP server and Secure Socket Layer (SSL) HTTPS server are both enabled.
- IEEE 802.1x is disabled.
- Port parameters
  - Interface speed and duplex mode is autonegotiate.
  - Auto-MDIX is enabled.
  - Flow control is off.
- VLANs
  - Default VLAN is VLAN 1.
  - VLAN trunking setting is dynamic auto (DTP).
  - Trunk encapsulation is negotiate.
  - VTP mode is server.
  - VTP version is Version 1.
  - Voice VLAN is disabled.
- STP, PVST+ is enabled on VLAN 1.
- MSTP is disabled.
- Optional spanning-tree features are disabled.
- FlexLinks are not configured.
- DHCP snooping is disabled.
- IP source guard is disabled.
- DHCP server port-based address allocation is disabled.
- Dynamic ARP inspection is disabled on all VLANs.
- IGMP snooping is enabled. No IGMP filters are applied.
- IGMP throttling setting is deny.
- The IGMP snooping querier feature is disabled.
- MVR is disabled.
- Port-based traffic
  - Broadcast, multicast, and unicast storm control is disabled.
  - No protected ports are defined.
  - Unicast and multicast traffic flooding is not blocked.
  - No secure ports are configured.
- CDP is enabled.

- UDLD is disabled.
- LLDP is disabled.
- SPAN and RSPAN are disabled.
- RMON is disabled.
- Syslog messages are enabled and appear on the console.
- SNMP is enabled (Version 1).
- No ACLs are configured.
- QoS is enabled.
- No EtherChannels are configured.
- IP unicast routing is disabled.

Using the Command-Line Interface

Information About Using the Command-Line Interface

This chapter describes the Cisco IOS command-line interface (CLI) and how to use it to configure your switch.

Command Modes

The Cisco IOS user interface is divided into many different modes. The commands available to you depend on which mode you are currently in. Enter a question mark (?) at the system prompt to obtain a list of commands available for each command mode.

When you start a session on the switch, you begin in user mode, often called user EXEC mode. Only a limited subset of the commands are available in user EXEC mode. For example, most of the user 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 user EXEC commands are not saved when the switch reboots.

To have access to all commands, you must enter privileged EXEC mode. You must enter a password to enter privileged EXEC mode. From this mode, you can enter any privileged EXEC command or enter global configuration mode.

Using the configuration modes (global, interface, and line), you can make changes to the running configuration. If you save the configuration, these commands are stored and used when the switch reboots. To access the various configuration modes, you must start at global configuration mode. From global configuration mode, you can enter interface configuration mode and line configuration mode.

Table 1 on page 9 describes the main command modes, how to access each one, the prompt you see in that mode, and how to exit the mode. The examples in the table use the hostname Switch.

Table 1 Command Mode Summary

Mode	Access Method	Prompt	Exit Method	About This Mode
User EXEC	Begin a session with your switch.	`Switch>`	Enter **logout** or **quit**.	Use this mode to ■ Change terminal settings. ■ Perform basic tests. ■ Display system information.
Privileged EXEC	While in user EXEC mode, enter the **enable** command.	`Switch#`	Enter **disable** to exit.	Use this mode to verify commands that you have entered. Use a password to protect access to this mode.
Global configuration	While in privileged EXEC mode, enter the **configure** command.	`Switch(config)#`	To exit to privileged EXEC mode, enter **exit** or **end**, or press **Ctrl-Z**.	Use this mode to configure parameters that apply to the entire switch.

Table 1 Command Mode Summary (continued)

Mode	Access Method	Prompt	Exit Method	About This Mode
Config-vlan	While in global configuration mode, enter the **vlan** *vlan-id* command.	`Switch(config-vlan)#`	To exit to global configuration mode, enter the **exit** command. To return to privileged EXEC mode, press **Ctrl-Z** or enter **end**.	Use this mode to configure VLAN parameters. When VTP mode is transparent, you can create extended-range VLANs (VLAN IDs greater than 1005) and save configurations in the switch startup configuration file.
VLAN configuration	While in privileged EXEC mode, enter the **vlan database** command.	`Switch(vlan)#`	To exit to privileged EXEC mode, enter **exit**.	Use this mode to configure VLAN parameters for VLANs 1 to 1005 in the VLAN database.
Interface configuration	While in global configuration mode, enter the **interface** command (with a specific interface).	`Switch(config-if)#`	To exit to global configuration mode, enter **exit**. To return to privileged EXEC mode, press **Ctrl-Z** or enter **end**.	Use this mode to configure parameters for the Ethernet ports.
Line configuration	While in global configuration mode, specify a line with the **line vty** or **line console** command.	`Switch(config-line)#`	To exit to global configuration mode, enter **exit**. To return to privileged EXEC mode, press **Ctrl-Z** or enter **end**.	Use this mode to configure parameters for the terminal line.

Help System

You can enter a question mark (?) at the system prompt to display a list of commands available for each command mode. You can also obtain a list of associated keywords and arguments for any command, as shown in Table 2 on page 10.

Table 2 Help Summary

Command	Purpose
help	Obtain a brief description of the help system in any command mode.

*abbreviated-command-entry***?**	Obtain a list of commands that begin with a particular character string.
	For example:
	`Switch# di?`
	`dir disable disconnect`

abbreviated-command-entry<**Tab**>	Complete a partial command name.
	For example:
	`Switch# sh conf<tab>`
	`Switch# show configuration`

Table 2 Help Summary (continued)

Command	Purpose
?	List all commands available for a particular command mode. For example: `Switch> ?`
command **?**	List the associated keywords for a command. For example: `Switch> show ?`

command keyword **?**	List the associated arguments for a keyword.
	For example:
	`Switch(config)# cdp holdtime ?`
	`  <10-255> Length of time (in sec) that receiver must keep this packet`

Understanding Abbreviated Commands

You need to enter only enough characters for the switch to recognize the command as unique.

This example shows how to enter the **show configuration** privileged EXEC command in an abbreviated form:

```
Switch# show conf
```

No and default Forms of Commands

Almost every configuration command also has a **no** form. In general, use the **no** form to disable a feature or function or reverse the action of a command. For example, the **no shutdown** interface configuration command reverses the shutdown of an interface. Use the command without the keyword **no** to reenable a disabled feature or to enable a feature that is disabled by default.

Configuration commands can also have a **default** form. The **default** form of a command returns the command setting to its default. Most commands are disabled by default, so the **default** form is the same as the **no** form. However, some commands are enabled by default and have variables set to certain default values. In these cases, the **default** command enables the command and sets variables to their default values.

CLI Error Messages

Table 3 on page 12 lists some error messages that you might encounter while using the CLI to configure your switch.

Table 3 Common CLI Error Messages

Error Message	Meaning	How to Get Help
`% Ambiguous command: "show con"`	You did not enter enough characters for your switch to recognize the command.	Reenter the command followed by a question mark (?) with a space between the command and the question mark. The possible keywords that you can enter with the command appear.
`% Incomplete command.`	You did not enter all the keywords or values required by this command.	Reenter the command followed by a question mark (?) with a space between the command and the question mark. The possible keywords that you can enter with the command appear.
`% Invalid input detected at '^' marker.`	You entered the command incorrectly. The caret (^) marks the point of the error.	Enter a question mark (?) to display all the commands that are available in this command mode. The possible keywords that you can enter with the command appear.

Configuration Logging

You can log and view changes to the switch configuration. You can use the Configuration Change Logging and Notification feature to track changes on a per-session and per-user basis. The logger tracks each configuration command that is applied, the user who entered the command, the time that the command was entered, and the parser return code for the command. This feature includes a mechanism for asynchronous notification to registered applications whenever the configuration changes. You can choose to have the notifications sent to the syslog.

Note: Only CLI or HTTP changes are logged.

How to Use the CLI to Configure Features

Configuring the Command History

The software provides a history or record of commands that you have entered. The command history feature is particularly useful for recalling long or complex commands or entries, including access lists. You can customize this feature to suit your needs as described in these sections:

- Changing the Command History Buffer Size, page 12 (optional)
- Recalling Commands, page 13 (optional)
- Disabling the Command History Feature, page 13 (optional)

Changing the Command History Buffer Size

By default, the switch records ten command lines in its history buffer. You can alter this number for a current terminal session or for all sessions on a particular line. These procedures are optional.

Beginning in privileged EXEC mode, enter this command to change the number of command lines that the switch records during the current terminal session:

```
Switch# terminal history [size number-of-lines]
```

The range is from 0 to 256.

Beginning in line configuration mode, enter this command to configure the number of command lines the switch records for all sessions on a particular line:

```
Switch(config-line)# history [size number-of-lines]
```

The range is from 0 to 256.

Recalling Commands

To recall commands from the history buffer, perform one of the actions listed in Table 4 on page 13. These actions are optional.

Table 4 Recalling Commands

Action[1]	Result
Press **Ctrl-P** or the up arrow key.	Recall commands in the history buffer, beginning with the most recent command. Repeat the key sequence to recall successively older commands.
Press **Ctrl-N** or the down arrow key.	Return to more recent commands in the history buffer after recalling commands with **Ctrl-P** or the up arrow key. Repeat the key sequence to recall successively more recent commands.
show history	While in privileged EXEC mode, list the last several commands that you just entered. The number of commands that appear is controlled by the setting of the **terminal history** global configuration command and the **history** line configuration command.

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

Disabling the Command History Feature

The command history feature is automatically enabled. You can disable it for the current terminal session or for the command line. These procedures are optional.

To disable the feature during the current terminal session, enter the **terminal no history** privileged EXEC command.

To disable command history for the line, enter the **no history** line configuration command.

Using Editing Features

This section describes the editing features that can help you manipulate the command line. It contains these sections:

- Enabling and Disabling Editing Features, page 13 (optional)
- Editing Commands Through Keystrokes, page 14 (optional)
- Editing Command Lines That Wrap, page 15 (optional)

Enabling and Disabling Editing Features

Although enhanced editing mode is automatically enabled, you can disable it, reenable it, or configure a specific line to have enhanced editing. These procedures are optional.

To globally disable enhanced editing mode, enter this command in line configuration mode:

```
Switch (config-line)# no editing
```

To reenable the enhanced editing mode for the current terminal session, enter this command in privileged EXEC mode:

```
Switch# terminal editing
```

To reconfigure a specific line to have enhanced editing mode, enter this command in line configuration mode:

```
Switch(config-line)# editing
```

Editing Commands Through Keystrokes

Table 5 on page 14 shows the keystrokes that you need to edit command lines. These keystrokes are optional.

Table 5 Editing Commands through Keystrokes

Capability	Keystroke[1]	Purpose
Move around the command line to make changes or corrections.	Press **Ctrl-B**, or press the left arrow key.	Move the cursor back one character.
	Press **Ctrl-F**, or press the right arrow key.	Move the cursor forward one character.
	Press **Ctrl-A**.	Move the cursor to the beginning of the command line.
	Press **Ctrl-E**.	Move the cursor to the end of the command line.
	Press **Esc B**.	Move the cursor back one word.
	Press **Esc F**.	Move the cursor forward one word.
	Press **Ctrl-T**.	Transpose the character to the left of the cursor with the character located at the cursor.
Recall commands from the buffer and paste them in the command line. The switch provides a buffer with the last ten items that you deleted.	Press **Ctrl-Y.**	Recall the most recent entry in the buffer.
	Press **Esc Y**.	Recall the next buffer entry. The buffer contains only the last 10 items that you have deleted or cut. If you press **Esc Y** more than ten times, you cycle to the first buffer entry.
Delete entries if you make a mistake or change your mind.	Press the **Delete** or **Backspace** key.	Erase the character to the left of the cursor.
	Press **Ctrl-D**.	Delete the character at the cursor.
	Press **Ctrl-K**.	Delete all characters from the cursor to the end of the command line.
	Press **Ctrl-U** or **Ctrl-X**.	Delete all characters from the cursor to the beginning of the command line.
	Press **Ctrl-W**.	Delete the word to the left of the cursor.
	Press **Esc D**.	Delete from the cursor to the end of the word.
Capitalize or lowercase words or capitalize a set of letters.	Press **Esc C**.	Capitalize at the cursor.
	Press **Esc L**.	Change the word at the cursor to lowercase.
	Press **Esc U**.	Capitalize letters from the cursor to the end of the word.

Table 5 Editing Commands through Keystrokes (continued)

Capability	Keystroke[1]	Purpose
Designate a particular keystroke as an executable command, perhaps as a shortcut.	Press **Ctrl-V** or **Esc Q**.	
Scroll down a line or screen on displays that are longer than the terminal screen can display. **Note:** The More prompt is used for any output that has more lines than can be displayed on the terminal screen, including **show** command output. You can use the **Return** and **Space** bar keystrokes whenever you see the More prompt.	Press the **Return** key.	Scroll down one line.
	Press the **Space** bar.	Scroll down one screen.
Redisplay the current command line if the switch suddenly sends a message to your screen.	Press **Ctrl-L** or **Ctrl-R.**	Redisplay the current command line.

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

Editing Command Lines That Wrap

You can use a wraparound feature for commands that extend beyond a single line on the screen. When the cursor reaches the right margin, the command line shifts ten spaces to the left. You cannot see the first ten characters of the line, but you can scroll back and check the syntax at the beginning of the command. The keystroke actions are optional.

To scroll back to the beginning of the command entry, press **Ctrl-B** or the left arrow key repeatedly. You can also press **Ctrl-A** to immediately move to the beginning of the line.

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

In this example, the **access-list** global configuration command entry extends beyond one line. When the cursor first reaches the end of the line, the line is shifted ten spaces to the left and redisplayed. The dollar sign ($) shows that the line has been scrolled to the left. Each time the cursor reaches the end of the line, the line is again shifted ten spaces to the left.

```
Switch(config)# access-list 101 permit tcp 131.108.2.5 255.255.255.0 131.108.1
Switch(config)# $ 101 permit tcp 131.108.2.5 255.255.255.0 131.108.1.20 255.25
Switch(config)# $t tcp 131.108.2.5 255.255.255.0 131.108.1.20 255.255.255.0 eq
Switch(config)# $108.2.5 255.255.255.0 131.108.1.20 255.255.255.0 eq 45
```

After you complete the entry, press **Ctrl-A** to check the complete syntax before pressing the **Return** key to execute the command. The dollar sign ($) appears at the end of the line to show that the line has been scrolled to the right:

```
Switch(config)# access-list 101 permit tcp 131.108.2.5 255.255.255.0 131.108.1$
```

The software assumes you have a terminal screen that is 80 columns wide. If you have a different width, use the **terminal width** privileged EXEC command to set the width of your terminal.

Use line wrapping with the command history feature to recall and modify previous complex command entries. For information about recalling previous command entries, see Editing Commands Through Keystrokes, page 14.

Searching and Filtering Output of show and more Commands

You can search and filter the output for **show** and **more** commands. This is useful when you need to sort through large amounts of output or if you want to exclude output that you do not need to see. Using these commands is optional.

To use this functionality, enter a **show** or **more** command followed by the pipe character (|), one of the keywords **begin**, **include**, or **exclude**, and an expression that you want to search for or filter out:

command **|** {**begin** | **include** | **exclude**} *regular-expression*

Expressions are case sensitive. For example, if you enter **| exclude output**, the lines that contain **output** are not displayed, but the lines that contain **Output** appear.

This example shows how to include in the output display only lines where the expression **protocol** appears:

```
Switch# show interfaces | include protocol
Vlan1 is up, line protocol is up
Vlan10 is up, line protocol is down
```

Accessing the CLI

You can access the CLI through a console connection, through Telnet, or by using the browser.

Accessing the CLI through a Console Connection or through Telnet

To understand the boot process and the options available for assigning IP information, see Performing Switch Setup Configuration, page 59

If your switch is already configured, you can access the CLI through a local console connection or through a remote Telnet session, but your switch must first be configured for this type of access. For more information, see Setting the Telnet Password for a Terminal Line: Example, page 184.

You can use one of these methods to establish a connection with the switch:

- Connect the switch console port to a management station or dial-up modem. For information about connecting to the console port, see the *Hardware Installation Guide Hardware Technical Guide*.
- Use any Telnet TCP/IP or encrypted Secure Shell (SSH) package from a remote management station. The switch must have network connectivity with the Telnet or SSH client, and the switch must have an enable secret password configured.

  The switch supports up to 16 simultaneous Telnet sessions. Changes made by one Telnet user are reflected in all other Telnet sessions.

  For information about configuring the switch for SSH, see Configuring the SSH Server, page 179. The switch supports up to five simultaneous secure SSH sessions.

After you connect through the console port, through a Telnet session or through an SSH session, the user EXEC prompt appears on the management station.

Configuring Interfaces

This chapter defines the types of interfaces and describes how to configure them.

Understanding Interface Types

This section describes the different types of interfaces supported by the switch with references to chapters that contain more detailed information about configuring these interface types. The rest of the chapter describes configuration procedures for physical interface characteristics.

UNI, NNI, and ENI Port Types

The switch supports user-network interfaces (UNIs), network node interfaces (NNIs), and enhanced network interfaces (ENIs). UNIs are typically connected to a host, such as a PC or a Cisco IP phone. NNIs are typically connected to a router or to another switch. ENIs have the same functionality as UNIs, but can be configured to support protocol control packets for Cisco Discovery Protocol (CDP), Spanning-Tree Protocol (STP), Link Layer Discovery Protocol (LLDP), and EtherChannel Link Aggregation Control Protocol (LACP) or Port Aggregation Protocol (PAgP).

By default, all ports are enabled as NNI.

All ports on the switch can be configured as UNIs or ENIs.

The default state for a UNI or ENI is administratively down to prevent unauthorized users from gaining access to other ports as you configure the switch. Traffic is not switched between these ports, and all arriving traffic at UNIs or ENIs must leave on NNIs to prevent a user from gaining access to another user's private network. If it is appropriate for two or more UNIs or ENIs to exchange traffic within the switch, the UNIs and ENIs can be assigned to a community VLAN. See Chapter 20, "Configuring VLANs," for instructions on how to configure community VLANs.

Note: Even though the default state for a UNI or ENI is shutdown, entering the **default interface** *interface-id* command changes the port to the enabled state.

The default status for an NNI is administratively up to allow a service provider remote access to the switch during initial configuration.

A port can be reconfigured from UNI to NNI or ENI and the reverse. When a port is reconfigured as another interface type, it inherits all the characteristics of that interface type. When you reconfigure a UNI or ENI to be an NNI, you must enable the port before it becomes active.

Changing the port type from UNI to ENI does not affect the administrative state of the port. If the UNI status is shut down, it remains shut down when reconfigured as an ENI; if the port is in a no shutdown state, it remains in the no shutdown state. At any time, all ports on the switch are either UNI, NNI, or ENI.

Port-Based VLANs

A VLAN is a switched network that is logically segmented by function, team, or application, without regard to the physical location of the users. Packets received on a port are forwarded only to ports that belong to the same VLAN as the receiving port. Network devices in different VLANs cannot communicate with one another without a Layer 3 device to route traffic between the VLANs.

VLAN partitions provide hard firewalls for traffic in the VLAN, and each VLAN has its own MAC address table. A VLAN comes into existence when a local port is associated with the VLAN ID or when a user creates te VLAN ID.

To isolate VLANs of different customers in a service-provider network, the switch uses UNI-ENI VLANs. UNI-ENI VLANs isolate user network interfaces (UNIs) or enhanced network interfaces (ENIs) on the switch from UNIs or ENIs that belong to other customer VLANs. There are two types of UNI-ENI VLANs:

- UNI-ENI isolated VLAN—This is the default VLAN state for all VLANs created on the switch. Local switching does not occur among UNIs or ENIs on the switch that belong to the same UNI-ENI isolated VLAN.
- UNI-ENI community VLAN—Local switching is allowed among UNIs and ENIs on the switch that belong to the same UNI community VLAN. If UNIs or ENIs belong to the same customer, and you want to switch packets between the ports, you can configure the common VLAN as a UNI-ENI community VLAN.

  Note: Local switching takes place between ENIs and UNIs in the same community VLAN. Because you can enable spanning tree on ENIs, but not on UNIs, you should use caution when configuring ENIs and UNIs in the same community VLAN. UNIs are always in the forwarding state.

To configure VLANs, use the **vlan** *vlan-id* global configuration command to enter VLAN configuration mode. The VLAN configurations for VLAN IDs 1 to 1005 are saved in the VLAN database. Extended-range VLANs (VLAN IDs 1006 to 4094) are not added to the VLAN database. VLAN configuration is saved in the switch running configuration, and you can save it in the switch startup configuration file by entering the **copy running-config startup-config** privileged EXEC command.

Add ports to a VLAN by using the **switchport** interface configuration commands:

- Identify the interface.
- For a trunk port, set trunk characteristics, and if desired, define the VLANs to which it can belong.
- For an access port, set and define the VLAN to which it belongs.
- For a tunnel port, set and define the VLAN ID for the customer-specific VLAN tag.

Switch Ports

Switch ports are Layer 2 only interfaces associated with a physical port. Switch ports belong to one or more VLANs. A switch port can be an access port, a trunk port, a private-VLAN port, or a tunnel port. You can configure a port as an access port or trunk port. You configure a private VLAN port as a host or promiscuous port that belongs to a private-VLAN primary or secondary VLAN. (Only NNIs can be configured as promiscuous ports.) You must manually configure tunnel ports as part of an asymmetric link connected to an IEEE 802.1Q trunk port. Switch ports are used for managing the physical interface and associated Layer 2 protocols and do not handle routing or bridging.

Configure switch ports by using the **switchport** interface configuration commands. Use the **switchport** command with no keywords to put an interface that is in Layer 3 mode into Layer 2 mode.

Note: When you put an interface that is in Layer 3 mode into Layer 2 mode, the previous configuration information related to the affected interface might be lost, and the interface is returned to its default configuration.

Access Ports

An access port belongs to and carries the traffic of only one VLAN. Traffic is received and sent in native formats with no VLAN tagging. Traffic arriving on an access port is assumed to belong to the VLAN assigned to the port. If an access port receives an 802.1Q tagged packet, the packet is dropped, and the source address is not learned. 802.1x can also be used for VLAN assignment.

Two types of access ports are supported:

- Static access ports are manually assigned to a VLAN.
- VLAN membership of dynamic access ports is learned through incoming packets. By default, a dynamic access port is a member of no VLAN, and forwarding to and from the port is enabled only when the VLAN membership of the port is discovered. UNIs begin forwarding packets as soon as they are enabled. Dynamic access ports on the switch are assigned to a VLAN by a VLAN Membership Policy Server (VMPS). Dynamic access ports for VMPS are only supported on UNIs and ENIs.

Trunk Ports

An 802.1Q trunk port carries the traffic of multiple VLANs and by default is a member of all VLANs in the VLAN database. A trunk port supports simultaneous tagged and untagged traffic. An 802.1Q trunk port is assigned a default Port VLAN ID (PVID), and all untagged traffic travels on the port default PVID. All untagged traffic and tagged traffic with a NULL VLAN ID are assumed to belong to the port default PVID. A packet with a VLAN ID equal to the outgoing port default PVID is sent untagged. All other traffic is sent with a VLAN tag.

Although by default a trunk port is a member of multiple VLANs, you can limit VLAN membership by configuring an allowed list of VLANs for each trunk port. The list of allowed VLANs does not affect any other port but the associated trunk port. By default, all possible VLANs (VLAN ID 1 to 4094) are in the allowed list. A trunk port can become a member of a VLAN only if the VLAN is in the enabled state.

For more information about trunk ports, see Configuring VLANs, page 289

Tunnel Ports

Tunnel ports are used in 802.1Q tunneling to segregate the traffic of customers in a service-provider network from other customers who are using the same VLAN number. You configure an asymmetric link from a tunnel port on a service-provider edge switch to an 802.1Q trunk port on the customer switch. Packets entering the tunnel port on the edge switch, already IEEE 802.1Q-tagged with the customer VLANs, are encapsulated with another layer of an 802.1Q tag (called the metro tag), containing a VLAN ID unique in the service-provider network, for each customer. The double-tagged packets go through the service-provider network keeping the original customer VLANs separate from those of other customers. At the outbound interface, also a tunnel port, the metro tag is removed, and the original VLAN numbers from the customer network are retrieved.

Note: IEEE 802.1Q tunneling is only supported when the switch is running the IP Services license.

Tunnel ports cannot be trunk ports or access ports and must belong to a VLAN unique to each customer.

Routed Ports

A routed port is a physical port that acts like a port on a router; it does not have to be connected to a router. A routed port is not associated with a particular VLAN, as is an access port. A routed port behaves like a regular router interface, except that it does not support VLAN subinterfaces. Routed ports can be configured with a Layer 3 routing protocol. A routed port is a Layer 3 interface only and does not support Layer 2 protocols, such as STP.

Configure routed ports by putting the interface into Layer 3 mode with the **no switchport** interface configuration command. Then assign an IP address to the port, enable routing, and assign routing protocol characteristics by using the **ip routing** and **router** *protocol* global configuration commands.

Note: Entering a **no switchport** interface configuration command shuts down the interface and then re-enables it, which might generate messages on the device to which the interface is connected. When you put an interface that is in Layer 2 mode into Layer 3 mode, the previous configuration information related to the affected interface might be lost.

The number of routed ports that you can configure is not limited by software. However, the interrelationship between this number and the number of other features being configured might impact CPU performance because of hardware limitations. See Configuring Layer 3 Interfaces, page 42 for information about what happens when hardware resource limitations are reached.

Note: For full Layer 3 routing, you must have the IP services image installed on the switch

Switch Virtual Interfaces

A switch virtual interface (SVI) represents a VLAN of switch ports as one interface to the routing or bridging function in the system. Only one SVI can be associated with a VLAN, but you need to configure an SVI for a VLAN only when you wish to route between VLANs or to provide IP host connectivity to the switch. By default, an SVI is created for the default VLAN (VLAN 1) to permit remote switch administration. Additional SVIs must be explicitly configured.

Note: You cannot delete interface VLAN 1.

SVIs provide IP host connectivity only to the system; in Layer 3 mode, you can configure routing across SVIs.

Although the switch supports a total of 1005 VLANs (and SVIs), the interrelationship between the number of SVIs and routed ports and the number of other features being configured might impact CPU performance because of hardware limitations. See Configuring Layer 3 Interfaces, page 42 for information about what happens when hardware resource limitations are reached.

SVIs are created the first time that you enter the **vlan** interface configuration command for a VLAN interface. The VLAN corresponds to the VLAN tag associated with data frames on an IEEE 802.1Q encapsulated trunk or the VLAN ID configured for an access port. Configure a VLAN interface for each VLAN for which you want to route traffic, and assign it an IP address. For more information, see Manually Assigning IP Information to SVIs, page 71.

Note: When you create an SVI, it does not become active until it is associated with a physical port.

SVIs support routing protocols.

Note: Routed ports (or SVIs) are supported only when the IP services image is installed on the switch.

EtherChannel Port Groups

EtherChannel port groups treat multiple switch ports as one switch port. These port groups act as a single logical port for high-bandwidth connections between switches or between switches and servers. An EtherChannel balances the traffic load across the links in the channel. If a link within the EtherChannel fails, traffic previously carried over the failed link changes to the remaining links. You can group multiple trunk ports into one logical trunk port, group multiple access ports into one logical access port, group multiple tunnel ports into one logical tunnel port, or group multiple routed ports into one logical routed port. Most protocols operate over either single ports or aggregated switch ports and do not recognize the physical ports within the port group. Exceptions are the Cisco Discovery Protocol (CDP), Link Aggregation Control Protocol (LACP), and the Port Aggregation Protocol (PAgP), which operate only on physical NNI or ENI ports.

When you configure an EtherChannel, you create a port-channel logical interface and assign an interface to the EtherChannel. For Layer 3 interfaces, you manually create the logical interface by using the **interface port-channel** global configuration command. Then you manually assign an interface to the EtherChannel by using the **channel-group** interface configuration command. For Layer 2 interfaces, use the **channel-group** interface configuration command to dynamically create the port-channel logical interface. This command binds the physical and logical ports together. For more information, see Configuring EtherChannels, page 1069

Power over Ethernet Ports

PoE-capable switch ports automatically supply power to these connected devices (if the switch senses that there is no power on the circuit):

- Cisco pre-standard powered devices (such as Cisco IP Phones and Cisco Aironet access points)
- 802.3af/802.3at-compliant powered devices

A powered device can receive redundant power when it is connected only to a PoE switch port and to an AC power source.

After the switch detects a powered device, it determines the device power requirements and then grants or denies power to the device. The switch can also sense the real-time power consumption of the device by monitoring and policing the power usage.

This section has this PoE information:

- Supported Protocols and Standards, page 21
- Powered-Device Detection and Initial Power Allocation, page 22
- Power Management Modes, page 22

Supported Protocols and Standards

The switch uses these protocols and standards to support PoE:

- CDP with power consumption—The powered device notifies the switch of the amount of power it is consuming. The switch does not reply to the power-consumption messages. The switch can only supply power to or remove power from the PoE port.
- Cisco intelligent power management—The powered device and the switch negotiate through power-negotiation CDP messages for an agreed power-consumption level. The negotiation allows a high-power Cisco powered device, which consumes more than 7 W, to operate at its highest power mode. The powered device first boots up in low-power mode, consumes less than 7 W, and negotiates to obtain enough power to operate in high-power mode. The device changes to high-power mode only when it receives confirmation from the switch.

  High-power devices can operate in low-power mode on switches that do not support power-negotiation CDP.

  Cisco intelligent power management is backward-compatible with CDP with power consumption; the switch responds according to the CDP message that it receives. CDP is not supported on third-party powered devices; therefore, the switch uses the IEEE classification to determine the power usage of the device.
- IEEE 802.3af/802.3at—The major features of this standard are powered-device discovery, power administration, disconnect detection, and optional powered-device power classification. For more information, see the standard.

Powered-Device Detection and Initial Power Allocation

The switch detects a Cisco pre-standard or an IEEE-compliant powered device when the PoE-capable port is in the no-shutdown state, PoE is enabled (the default), and the connected device is not being powered by an AC adaptor.

After device detection, the switch determines the device power requirements based on its type:

- A Cisco pre-standard powered device does not provide its power requirement when the switch detects it, so the switch allocates 15.4 W as the initial allocation for power budgeting.

  The initial power allocation is the maximum amount of power that a powered device requires. The switch initially allocates this amount of power when it detects and powers the powered device. As the switch receives CDP messages from the powered device and as the powered device negotiates power levels with the switch through CDP power-negotiation messages, the initial power allocation might be adjusted.

- The switch classifies the detected IEEE device within a power consumption class. Based on the available power in the power budget, the switch determines if a port can be powered. Table 1 lists these levels.

Class	Maximum Power Level Required from the Switch
0 (class status unknown)	15.4 W
1	4 W
2	7 W
3	15.4 W
4 (POE+)	30 W (requires LLDP)

Note: Prior to release 15.2(6)E1, if a rack mounted, IE4010 or IE5000 series switch, was powered by 2 PWR-RGD-AC-DC-250 power supplies, the correct total PoE budget (385W) as indicated in the Data Sheet, would not be displayed. If the proper total power budget isn't displayed on your switch, upgrade to release 15.2(6)E1 or later.

The switch monitors and tracks requests for power and grants power only when it is available. The switch tracks its power budget (the amount of power available on the switch for PoE). The switch performs power-accounting calculations when a port is granted or denied power to keep the power budget up to date.

After power is applied to the port, the switch uses CDP to determine the *actual* power consumption requirement of the connected Cisco powered devices, and the switch adjusts the power budget accordingly. This does not apply to third-party PoE devices. The switch processes a request and either grants or denies power. If the request is granted, the switch updates the power budget. If the request is denied, the switch ensures that power to the port is turned off, generates a syslog message, and updates the LEDs. Powered devices can also negotiate with the switch for more power.

If the switch detects a fault caused by an undervoltage, overvoltage, overtemperature, oscillator-fault, or short-circuit condition, it turns off power to the port, generates a syslog message, and updates the power budget and LEDs.

Note: IE switches may show occasional PoE controller error messages on the console, for example:

```
%ILPOWER-3-CONTROLLER_ERR: Controller error, Controller number 0: accessing failed
```

This can occur when there are no powered devices connected and all ports continue to function normally. There are no workarounds. These messages can be ignored.

If these or any other errors seen cause performance issues, contact Cisco support.

Power Management Modes

To limit the overall PoE budget of DIN rail switches such as the IE-4000, use the global configuration command **power inline wattage max <4-125>**.

Note - This command does not apply to rack-mount switches with integrated power supplies, such as the IE-4010 and IE-5000.

The switch supports these PoE modes:

- **auto**—The switch automatically detects if the connected device requires power. If the switch discovers a powered device connected to the port and if the switch has enough power, it grants power, updates the power budget, turns on power to the port on a first-come, first-served basis, and updates the LEDs. For LED information, see the hardware installation guide.

  If the switch has enough power for all the powered devices, they all come up. If enough power is available for all powered devices connected to the switch, power is turned on to all devices. If there is not enough available PoE, or if a device is disconnected and reconnected while other devices are waiting for power, it cannot be determined which devices are granted or are denied power.

  If granting power would exceed the system power budget, the switch denies power, ensures that power to the port is turned off, generates a syslog message, and updates the LEDs. After power has been denied, the switch periodically rechecks the power budget and continues to attempt to grant the request for power.

  If a device being powered by the switch is then connected to wall power, the switch might continue to power the device. The switch might continue to report that it is still powering the device whether the device is being powered by the switch or receiving power from an AC power source.

  If a powered device is removed, the switch automatically detects the disconnect and removes power from the port. You can connect a nonpowered device without damaging it.

  You can specify the maximum wattage that is allowed on the port. If the IEEE class maximum wattage of the powered device is greater than the configured maximum value, the switch does not provide power to the port. If the switch powers a powered device, but the powered device later requests through CDP messages more than the configured maximum value, the switch removes power to the port. The power that was allocated to the powered device is reclaimed into the global power budget. If you do not specify a wattage, the switch delivers the maximum value. Use the **auto** setting on any PoE port. The auto mode is the default setting.

- **static**—The switch pre-allocates power to the port (even when no powered device is connected) and guarantees that power will be available for the port. The switch allocates the port configured maximum wattage, and the amount is never adjusted through the IEEE class or by CDP messages from the powered device. Because power is pre-allocated, any powered device that uses less than or equal to the maximum wattage is guaranteed to be powered when it is connected to the static port. The port no longer participates in the first-come, first-served model.

  However, if the powered-device IEEE class is greater than the maximum wattage, the switch does not supply power to it. If the switch learns through CDP messages that the powered device needs more than the maximum wattage, the powered device is shutdown.

  If you do not specify a wattage, the switch pre-allocates the maximum value. The switch powers the port only if it discovers a powered device. Use the **static** setting on a high-priority interface.

- **never**—The switch disables powered-device detection and never powers the PoE port even if an unpowered device is connected. Use this mode only when you want to make sure power is never applied to a PoE-capable port, making the port a data-only port.

For information on configuring a PoE port, see Configuring a Power Management Mode on a PoE Port, page 36.

Power Monitoring and Power Policing

When policing of the real-time power consumption is enabled, the switch takes action when a powered device consumes more power than the maximum amount allocated, also referred to as the *cutoff-power value*.

When PoE is enabled, the switch senses the real-time power consumption of the powered device. The switch monitors the real-time power consumption of the connected powered device; this is called *power monitoring* or *power sensing*. The switch also polices the power usage with the *power policing* feature.

Power monitoring is backward-compatible with Cisco intelligent power management and CDP-based power consumption. It works with these features to ensure that the PoE port can supply power to the powered device. For more information about these PoE features, see Supported Protocols and Standards, page 21.

The switch senses the real-time power consumption of the connected device as follows:

1. The switch monitors the real-time power consumption on individual ports.
2. The switch records the power consumption, including peak power usage. The switch reports the information through the CISCO-POWER-ETHERNET-EXT-MIB.
3. If power policing is enabled, the switch polices power usage by comparing the real-time power consumption to the maximum power allocated to the device. For more information about the maximum power consumption, also referred to as the *cutoff power*, on a PoE port, see Maximum Power Allocation (Cutoff Power) on a PoE Port, page 24.

   If the device uses more than the maximum power allocation on the port, the switch can either turn off power to the port, or the switch can generate a syslog message and update the LEDs (the port LED is now blinking amber) while still providing power to the device based on the switch configuration. By default, power-usage policing is disabled on all PoE ports.

   If error recovery from the PoE error-disabled state is enabled, the switch automatically takes the PoE port out of the error-disabled state after the specified amount of time.

   If error recovery is disabled, you can manually re-enable the PoE port by using the **shutdown** and **no shutdown** interface configuration commands.
4. If policing is disabled, no action occurs when the powered device consumes more than the maximum power allocation on the PoE port, which could adversely affect the switch.

Maximum Power Allocation (Cutoff Power) on a PoE Port

When power policing is enabled, the switch determines one of the these values as the cutoff power on the PoE port in this order:

1. Manually when you set the user-defined power level that the switch budgets for the port by using the **power inline consumption default** *wattage* global or interface configuration command
2. Manually when you set the user-defined power level that limits the power allowed on the port by using the **power inline auto max** *max-wattage* or the **power inline static max** *max-wattage* interface configuration command
3. Automatically when the switch sets the power usage of the device by using CDP power negotiation or by the IEEE classification
4. Automatically when the switch sets the power usage to be the default value of 15400 mW

Use the first or second method in the previous list to manually configure the cutoff-power value by entering the **power inline consumption default** *wattage* or the **power inline** [**auto** | **static max**] *max-wattage* command. If you are not manually configuring the cutoff-power value, the switch automatically determines the value by using CDP power negotiation or the device IEEE classification, which is the third method in the previous list. If the switch cannot determine the value by using one of these methods, it uses the default value of 15400 mW (the fourth method in the previous list).

Power Consumption Values

You can configure the initial power allocation and the maximum power allocation on a port. However, these values are only the configured values that determine when the switch should turn on or turn off power on the PoE port. The maximum power allocation is not the same as the actual power consumption of the powered device. The actual cutoff power value that the switch uses for power policing is not equal to the configured power value.

When power policing is enabled, the switch polices the power usage *at the switch port*, which is greater than the power consumption of the device. When you are manually set the maximum power allocation, you must consider the power loss over the cable from the switch port to the powered device. The cutoff power is the sum of the rated power consumption of the powered device and the worst-case power loss over the cable.

The actual amount of power consumed by a powered device on a PoE port is the cutoff-power value plus a calibration factor of 500 mW (0.5 W). The actual cutoff value is approximate and varies from the configured value by a percentage of the configured value. For example, if the configured cutoff power is 12 W, the actual cutoff-value is 11.4 W, which is 5% less than the configured value.

We recommend that you enable power policing when PoE is enabled on your switch. For example, if policing is disabled and you set the cutoff-power value by using the **power inline auto max 6300** interface configuration command, the configured maximum power allocation on the PoE port is 6.3 W (6300 mW). The switch provides power to the connected devices on the port if the device needs up to 6.3 W. If the CDP-power negotiated value or the IEEE classification value exceeds the configured cutoff value, the switch does not provide power to the connected device. After the switch turns on power on the PoE port, the switch does not police the real-time power consumption of the device, and the device can consume more power than the maximum allocate d amount, which could adversely affect the switch and the devices connected to the other PoE ports.

Because the switch supports internal power supplies and the Cisco Redundant Power System 2300 (also referred to as the RPS 2300), the total amount of power available for the powered devices varies depending on the power supply configuration.

The switch supports dual power supplies. If a power supply is removed or fails and the switch does not have enough power for the powered devices, the switch first denies power to low-priority ports in descending order of port numbers, and then to high priority ports in descending numbers. The total available PoE power is 65 watts per power supply.

- If a power supply is removed and replaced by a new power supply with less power and the switch does not have enough power for the powered devices, the switch denies power to the PoE ports in auto mode in descending order of the port numbers. If the switch still does not have enough power, the switch then denies power to the PoE ports in static mode in descending order of the port numbers.
- If the new power supply supports more power than the previous one and the switch now has more power available, the switch grants power to the PoE ports in static mode in ascending order of the port numbers. If it still has power available, the switch then grants power to the PoE ports in auto mode in ascending order of the port numbers.

Dual-Purpose Ports on IE 4000

Each dual-purpose port is considered a single interface with dual front ends (an RJ-45 connector and an SFP module connector). The dual front ends are not redundant interfaces; the switch activates only one connector of the pair.

By default, dual-purpose ports are user-network interfaces (UNIs) and SFP-only module ports are network node interfaces (NNIs). TBy default, the switch dynamically selects the dual-purpose port media type that first links up. However, you can use the media-type interface configuration command to manually select the RJ-45 connector or the SFP module connector.

Each dual-purpose port has two LEDs: one shows the status of the SFP module port, and one shows the status of the RJ-45 port. The port LED is on for whichever connector is active. For more information about the LEDs, see the hardware installation guide.

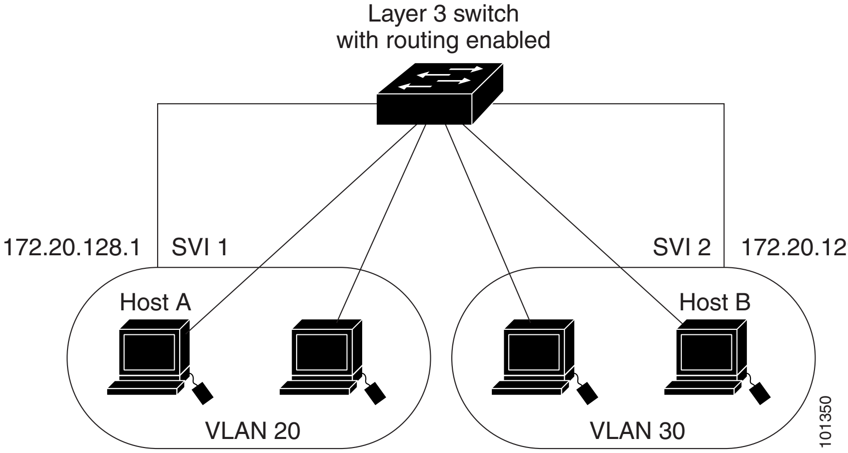
Connecting Interfaces

Devices within a single VLAN can communicate directly through any switch. Ports in different VLANs cannot exchange data without going through a routing device. With a standard Layer 2 switch, ports in different VLANs have to exchange information through a router.

By default, the switch provides VLAN isolation between UNIs or ENIs. UNIs and ENIs cannot exchange traffic unless they are changed to NNIs or assigned to a UNI-ENI community VLAN.

By using the switch with routing enabled, when you configure both VLAN 20 and VLAN 30 with an SVI to which an IP address is assigned, packets can be sent from Host A to Host B directly through the switch with no need for an external router (Figure 1 on page 26).

Figure 1 Connecting VLANs with the Switch



When the IP services image is running on the switch, routing can be enabled on the switch. Whenever possible, to maintain high performance, forwarding is done by the switch hardware. However, only IP Version 4 packets with Ethernet II encapsulation can be routed in hardware. The routing function can be enabled on all SVIs and routed ports. The switch routes only IP traffic. When IP routing protocol parameters and address configuration are added to an SVI or routed port, any IP traffic received from these ports is routed.

Using the Switch USB Port

Note: Windows PCs require a driver for the USB port. See the hardware installation guide for driver installation instructions.

Use the supplied USB Type A-to-USB mini-Type B cable to connect a PC or other device to the switch. The connected device must include a terminal emulation application. When the switch detects a valid USB connection to a powered-on device that supports host functionality (such as a PC), input from the RJ-45 console is immediately disabled, and input from the USB console is enabled. Removing the USB connection immediately reenables input from the RJ-45 console connection. A LED on the switch shows which console connection is in use.

Console Port Change Logs

At software startup, a log shows whether the USB or the RJ-45 console port is active. The switch first displays the RJ-45 media type.

In the sample output, the switch has a connected USB console cable. Because the bootloader did not change to the USB console, the first log from the switch shows the RJ-45 console. A short time later, the console changes and the USB console log appears.

```
switch
*Mar  1 00:01:00.171: %USB_CONSOLE-6-MEDIA_RJ45: Console media-type is RJ45.
*Mar  1 00:01:00.431: %USB_CONSOLE-6-MEDIA_USB: Console media-type is USB.
```

When the USB cable is removed or the PC de-activates the USB connection, the hardware automatically changes to the RJ-45 console interface:

```
switch
Mar  1 00:20:48.635: %USB_CONSOLE-6-MEDIA_RJ45: Console media-type is RJ45.
```

You can configure the console type to always be RJ-45, and you can configure an inactivity timeout for the USB connector.

Configuring the Console Media Type

Beginning in privileged EXEC mode, follow these steps to select the RJ-45 console media type. If you configure the RJ-45 console, USB console operation is disabled, and input always remains with the RJ-45 console.

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	line console 0	Configure the console. Enter line configuration mode.
3.	**media-type rj45**	Configure the console media type to always be RJ-45. If you do not enter this command and both types are connected, the default is USB.
4.	**end**	Return to privileged EXEC mode.
5.	**show running-configuration**	Verify your settings.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

This example disables the USB console media type and enables the RJ-45 console media type.

```
Switch# configure terminal
Switch(config)# line console 0
Switch(config-line)# media-type rj45
```

A log shows that this termination has occurred. This example shows that the console on switch reverted to RJ-45.

```
*Mar  1 00:25:36.860: %USB_CONSOLE-6-CONFIG_DISABLE: Console media-type USB disabled by system
configuration, media-type reverted to RJ45.
```

A log entry shows when a console cable is attached. If a USB console cable is connected to the switch, it is prevented from providing input.

```
*Mar  1 00:34:27.498: %USB_CONSOLE-6-CONFIG_DISALLOW: Console media-type USB is disallowed by system
configuration, media-type remains RJ45.
```

This example reverses the previous configuration and immediately activates the USB console that is connected.

```
Switch# configure terminal
Switch(config)# line console 0
Switch(config-line)# no media-type rj45
```

Using Interface Configuration Mode

The switch supports these interface types:

- Physical ports—switch ports, routed ports, UNIs, NNIs, and ENIs
- VLANs—switch virtual interfaces
- Port-channels—EtherChannel interfaces

You can also configure a range of interfaces (see Configuring a Range of Interfaces, page 29).

To configure a physical interface (port), specify the interface type, the module number, and the switch port number, and enter interface configuration mode.

- Type—10/100/1000 Mbps Ethernet ports, Gigabit Ethernet (gigabitethernet or gi), TenGigabitEthernet (tengigethernet or te) for or small form-factor pluggable (SFP) module Gigabit Ethernet interfaces.
- Module number—The module or slot number on the switch.
- Port number—The interface number on the switch. The port numbers always begin at 1, starting with the leftmost port when facing the front of the switch, for example, gigabitethernet 1/1. If there is more than one interface type (for example, 10/100 ports and SFP module ports), the port numbers restart with the second interface type: gigabitethernet 1/1.

You can identify physical interfaces by physically checking the interface location on the switch. You can also use the **show** privileged EXEC commands to display information about a specific interface or all the interfaces on the switch. The remainder of this chapter primarily provides physical interface configuration procedures.

Procedures for Configuring Interfaces

These general instructions apply to all interface configuration processes.

1. Enter the **configure terminal** command at the privileged EXEC prompt:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#
```

2. Enter the **interface** global configuration command. Identify the interface type and the number of the connector. In this example, Fast Ethernet port 1 is selected:

```
Switch(config)# interface fastethernet0/1
Switch(config-if)#
```

Note: You do not need to add a space between the interface type and interface number. For example, in the preceding line, you can specify either **fastethernet 0/1**, **fastethernet0/1**, **fa 0/1**, or **fa0/1**.

3. If you are configuring a UNI or ENI, enter the **no shutdown** interface configuration command to enable the interface:

```
Switch(config-if)# no shutdown
```

4. Follow each **interface** command with the interface configuration commands that the interface requires. The commands that you enter define the protocols and applications that will run on the interface. The commands are collected and applied to the interface when you enter another interface command or enter **end** to return to privileged EXEC mode.

You can also configure a range of interfaces by using the **interface range** or **interface range macro** global configuration commands. Interfaces configured in a range must be the same type and must be configured with the same feature options.

5. After you configure an interface, verify its status by using the **show** privileged EXEC commands listed in the Monitoring and Maintaining the Interfaces, page 45.

Enter the **show interfaces** privileged EXEC command to see a list of all interfaces on or configured for the switch. A report is provided for each interface that the device supports or for the specified interface.

Configuring a Range of Interfaces

You can use the **interface range** global configuration command to configure multiple interfaces with the same configuration parameters. When you enter the interface range configuration mode, all command parameters that you enter are attributed to all interfaces within that range until you exit this mode.

Beginning in privileged EXEC mode, follow these steps to configure a range of interfaces with the same parameters:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface range** {*port-range*}	Specify the range of interfaces (VLANs or physical ports) to be configured, and enter interface range configuration mode. ■ You can use the **interface range** command to configure up to five port ranges or a previously defined macro. ■ In a comma-separated *port-range*, you must enter the interface type for each entry and enter spaces before and after the comma. ■ In a hyphen-separated *port-range*, you do not need to re-enter the interface type, but you must enter a space before the hyphen.
3.	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
4.		Use the normal configuration commands to apply the configuration parameters to all interfaces in the range.
5.	**end**	Return to privileged EXEC mode.
6.	**show interfaces** [*interface-id*]	Verify the configuration of the interfaces in the range.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

When using the **interface range** global configuration command, note these guidelines:

- Valid entries for *port-range*:
  - **vlan** *vlan-ID* - *vlan-ID*, where the VLAN ID is 1 to 4094
  - **gigabitethernet** module/{*first port*} - {*last port*}, where the module is always 1
  - **tengigabitethernet** module/{*first port*} - {*last port*}, where the module is always 1
  - **port-channel** *port-channel-number* - *port-channel-number*, where the *port-channel-number* is 1 to 10.

  When you use the **interface range** command with port channels, the first and last port channel number must be active port channels.
- The **interface range** command only works with VLAN interfaces that have been configured with the **interface vlan** command. The **show running-config** privileged EXEC command displays the configured VLAN interfaces. VLAN interfaces not displayed by the **show running-config** command cannot be used with the **interface range** command.
- All interfaces defined as in a range must be the same type (all Fast Ethernet ports, all Gigabit Ethernet ports, all EtherChannel ports, or all VLANs), but you can enter multiple ranges in a command.

This example shows how to use the **interface range** global configuration command to set the speed on ports 1 and 2 to 100 Mbps:

```
Switch# configure terminal
```

```
Switch(config)# interface range fastethernet0/1 - 2
Switch(config-if-range)# no shutdown
Switch(config-if-range)# speed 100
```

This example shows how to use a comma to add different interface type strings to the range to enable Fast Ethernet ports 1 to 3 and Gigabit Ethernet ports 1 and 2 to receive 802.3x flow control pause frames:

```
Switch# configure terminal
Switch(config)# interface range fastethernet0/1 - 3 , GigabitEthernet1/17 - 2
Switch(config-if-range)# flowcontrol receive on
```

If you enter multiple configuration commands while you are in interface range mode, each command is executed as it is entered. The commands are not batched together and executed after you exit interface range mode. If you exit interface range configuration mode while the commands are being executed, some commands might not be executed on all interfaces in the range. Wait until the command prompt reappears before exiting interface range configuration mode.

Configuring and Using Interface Range Macros

You can create an interface range macro to automatically select a range of interfaces for configuration. Before you can use the **macro** keyword in the **interface range macro** global configuration command string, you must use the **define interface-range** global configuration command to define the macro.

Beginning in privileged EXEC mode, follow these steps to define an interface range macro:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**define interface-range** *macro_name interface-range*	Define the interface-range macro, and save it in NVRAM. ■ The *macro_name* is a 32-character maximum character string. ■ A macro can contain up to five comma-separated interface ranges. ■ Each *interface-range* must consist of the same port type.
3.	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
4.	**interface range macro** *macro_name*	Select the interface range to be configured using the values saved in the interface-range macro called *macro_name.* You can now use the normal configuration commands to apply the configuration to all interfaces in the defined macro.
5.	**end**	Return to privileged EXEC mode.
6.	**show running-config | include define**	Show the defined interface range macro configuration.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no define interface-range** *macro_name* global configuration command to delete a macro.

When using the **define interface-range** global configuration command, note these guidelines:

- Valid entries for *interface-range*:
  - **vlan** *vlan-ID* - *vlan-ID*, where the VLAN ID is 1 to 4094
  - **gigabitethernet** module/{*first port*} - {*last port*}, where the module is always 1
  - **tengigabitethernet** module/{*first port*} - {*last port*}, where the module is always 1

  - **port-channel** *port-channel-number* - *port-channel-number*, where the *port-channel-number* is 1 to 10.

    When you use the interface ranges with port channels, the first and last port channel number must be active port channels.

- You must add a space between the first interface number and the hyphen when entering an *interface-range*. For example, **GigabitEthernet1/17 - 18** is a valid range; **GigabitEthernet1/17-18** is not a valid range.
- The VLAN interfaces must have been configured with the **interface vlan** command. The **show running-config** privileged EXEC command displays the configured VLAN interfaces. VLAN interfaces not displayed by the **show running-config** command cannot be used as *interface-ranges*.
- All interfaces defined as in a range must be the same type (all Fast Ethernet ports, all Gigabit Ethernet ports, all EtherChannel ports, or all VLANs), but you can combine multiple interface types in a macro.

This example shows how to define an interface-range named *enet_list* to include ports 1 and 2 and to verify the macro configuration:

```
Switch# configure terminal
Switch(config)# define interface-range enet_list GigabitEthernet1/17 - 2
Switch(config)# end
Switch# show running-config | include define
define interface-range enet_list GigabitEthernet1/17 - 2
```

This example shows how to create a multiple-interface macro named *macro1* and assign all of the interfaces in the range to a VLAN:

```
Switch# configure terminal
Switch(config)# define interface-range macro1 fastethernet0/1 - 2, GigabitEthernet1/17 - 2
Switch(config)# interface range macro macro1
Switch(config-if-range)# switchport access vlan 20
Switch(config-if-range)# no shut
Switch(config-if-range)# end
```

This example shows how to enter interface range configuration mode for the interface-range macro *enet_list*:

```
Switch# configure terminal
Switch(config)# interface range macro enet_list
Switch(config-if-range)#
```

This example shows how to delete the interface-range macro *enet_list* and to verify that it was deleted.

```
Switch# configure terminal
Switch(config)# no define interface-range enet_list
Switch(config)# end
Switch# show run | include define
Switch#
```

Configuring Ethernet Interfaces

- Default Ethernet Interface Configuration, page 32
- Configuring the Port Type, page 33
- Configuring Interface Speed and Duplex Mode, page 34
- Configuring a Power Management Mode on a PoE Port, page 36
- Budgeting Power for Devices Connected to a PoE Port, page 37

- Configuring IEEE 802.3x Flow Control, page 39
- Configuring Auto-MDIX on an Interface, page 40
- Adding a Description for an Interface, page 41

Default Ethernet Interface Configuration

Table 6 on page 32 shows the Ethernet interface default configuration for NNIs, and Table 7 on page 33 shows the Ethernet interface default configuration for UNIs and ENIs. For more details on the VLAN parameters listed in the table, see Configuring VLANs, page 289

Note: To configure Layer 2 parameters, if the interface is in Layer 3 mode, you must enter the **switchport** interface configuration command without any parameters to put the interface into Layer 2 mode. This shuts down the interface and then re-enables it, which might generate messages on the device to which the interface is connected. When you put an interface that is in Layer 3 mode into Layer 2 mode, the previous configuration information related to the affected interface might be lost, and the interface is returned to its default configuration.

Table 6 Default Ethernet Configuration for NNIs

Feature	Default Setting
Operating mode	**Layer 2 or** switching mode (**switchport** command).
Allowed VLAN range	VLANs 1- 4094.
Default VLAN (for access ports)	VLAN 1 (Layer 2 interfaces only).
Native VLAN (for 802.1Q trunks)	VLAN 1 (Layer 2 interfaces only).
VLAN trunking	Switchport mode access (Layer 2 interfaces only).
Port enable state	Enabled.
Port description	None defined.
Speed	Autonegotiate.
Duplex mode	Full.
802.3x flow control	Flow control is set to **receive**: **off**. It is always off for sent packets.
EtherChannel	Disabled on all Ethernet ports. See Configuring EtherChannels, page 1069
Port blocking (unknown multicast and unknown unicast traffic)	Disabled (not blocked) (only Layer 2 interfaces).
Broadcast, multicast, and unicast storm control	Disabled.
Port security	Disabled (only Layer 2 interfaces).
Port Fast	Disabled.
Auto-MDIX	Enabled. **Note:** The switch might not support a pre-standard powered device—such as Cisco IP phones and access points that do not fully support 802.3af/802.3at—if that powered device is connected to the switch through a crossover cable. This is regardless of whether auto-MIDX is enabled on the switch port.
Power over Ethernet (PoE)	Enabled (auto).
Cisco Discovery Protocol (CDP)	Enabled.
VMPS	Not configured.

Table 7 Default Ethernet Configuration for UNIs and ENIs

Feature	Default Setting
Operating mode	**Layer 2 or** switching mode (**switchport** command).
Allowed VLAN range	VLANs 1- 4094.
Default VLAN (for access ports)	VLAN 1 (Layer 2 interfaces only).
Native VLAN (for 802.1Q trunks)	VLAN 1 (Layer 2 interfaces only).
VLAN trunking	Switchport mode access (Layer 2 interfaces only).
Dynamic VLAN	Enabled.
Port enable state	Disabled when no configuration file exists.
Port description	None defined.
Speed	Autonegotiate.
Duplex mode	Autonegotiate.
802.3x flow control	Flow control is set to **receive**: **off**. It is always off for sent packets.
EtherChannel	Disabled on all Ethernet ports. See Configuring EtherChannels, page 1069
Port blocking (unknown multicast and unknown unicast traffic)	Disabled (not blocked) (only Layer 2 interfaces).
Broadcast, multicast, and unicast storm control	Disabled.
Port security	Disabled (only Layer 2 interfaces).
Auto-MDIX	Enabled.

Configuring the Port Type

By default, all the 10/100 ports on the switch are configured as UNIs, and the SFP module ports are configured as NNIs.

You use the **port-type** interface configuration command to change the port types. An ENI has the same characteristics as a UNI, but it can be configured to support CDP, STP, LLDP, and Etherchannel LACP and PAgP.

When a port is changed from an NNI to a UNI or ENI, it inherits the configuration of the assigned VLAN, either in isolated or community mode.

When you change a port from NNI to UNI or ENI or the reverse, any features exclusive to the port type revert to the default configuration. For Layer 2 protocols, such as STP, CDP, and LLDP, the default for UNIs and ENIs is disabled (although they can be enabled on ENIs) and the default for NNIs is enabled.

Note: By default, the switch sends keepalive messages on UNI s and ENIs and does not send keepalive messages on NNIs. Changing the port type from UNI or ENI to NNI or from NNI to UNI or ENI has no effect on the keepalive status. You can change the keepalive state from the default setting by entering the [**no**] **keepalive** interface configuration command. If you enter the **keepalive** command with no arguments, keepalive packets are sent with the default time interval (10 seconds) and number of retries (5). Entering the **no keepalive** command disables keepalive packets on the interface.

Beginning in privileged EXEC mode, follow these steps to configure the port type on an interface:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode
2.	**interface** *interface-id*	Specify the interface to configure, and enter interface configuration mode.
3.	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
4.	**port-type** {**eni** | **nni** | **uni**}	Change a port to an ENI, NNI, or UNI.
5.	**end**	Return to privileged EXEC mode.
6.	**show interfaces** *interface-id*	Verify the interface 802.3x flow control settings.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Entering the **no port-type** or **default port-type** interface configuration command returns the port to the default state: UNI for Fast Ethernet ports and NNI for Gigabit Ethernet ports.

This example shows how to change a port from a UNI to an NNI and save it to the running configuration.

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface fastethernet0/1
Switch(config-if)# port-type nni
Switch(config-if)# no shutdown
5d20h: %SYS-5-CONFIG_I: Configured from console by console
Switch(config-if)# end
Switch# copy running-config startup-config
```

Configuring Interface Speed and Duplex Mode

Ethernet interfaces on the switch operate at 10, 100, or 1000 Mbps and in either full- or half-duplex mode. In full-duplex mode, two stations can send and receive traffic at the same time. Normally, 10-Mbps ports operate in half-duplex mode, which means that stations can either receive or send traffic.

Switch models include combinations of Fast Ethernet (10/100-Mbps) ports, Gigabit Ethernet (10/100/1000-Mbps) ports, and small form-factor pluggable (SFP) module slots supporting SFP modules.

These sections describe how to configure the interface speed and duplex mode:

- Speed and Duplex Configuration Guidelines, page 34
- Setting the Interface Speed and Duplex Parameters, page 35

Speed and Duplex Configuration Guidelines

When configuring an interface speed and duplex mode, note these guidelines:

- You can configure interface speed on Fast Ethernet (10/100-Mbps) and Gigabit Ethernet (10/100/1000-Mbps) ports. You can configure Fast Ethernet ports to full-duplex, half-duplex, or to autonegotiate mode. You can configure Gigabit Ethernet ports to full-duplex mode or to autonegotiate. You also can configure Gigabit Ethernet ports to half-duplex mode if the speed is 10 or 100 Mbps. Half-duplex mode is not supported on Gigabit Ethernet ports operating at 1000 Mbps.
- With the exception of when 1000BASE-T SFP modules are installed in the SFP module slots, you cannot configure speed on SFP module ports, but you can configure speed to not negotiate (**nonegotiate**) if connected to a device that does not support autonegotiation.

  However, when a 1000BASE-T SFP module is in the SFP module slot, you can configure speed as 10, 100, or 1000 Mbps, or auto, but not as **nonegotiate**.

On a 100BASE-FX SFP module, you cannot configure the speed as **nonegotiate**.

- You cannot configure duplex mode on SFP module ports; they operate in full-duplex mode except in these situations:
  - When a Cisco1000BASE-T SFP module is in the SFP module slot, you can configure duplex mode to **auto** or **full**. Half-duplex mode is supported with the **auto** setting.
  - When a Cisco100BASE-FX SFP module is in the SFP module slot, you can configure duplex mode to half or full (the default for this SFP module). Although the auto keyword is available, it puts the interface in full-duplex mode because the 100BASE-FX SFP module does not support autonegotiation.
- If both ends of the line support autonegotiation, we highly recommend the default setting of **auto** negotiation.
- If you configure the speed as **nonegotiate** on one device and configure **auto** negotiation on the remote device, the port may go down on some platforms. The IEEE specification does not define the expected behavior of an auto negotiation mismatch on a 1000BaseX link. The link may or may not come up.
- If one interface supports autonegotiation and the other end does not, configure duplex and speed on both interfaces; do not use the **auto** setting on the supported side.
- When STP is enabled and a port is reconfigured, the switch can take up to 30 seconds to check for loops. The port LED is amber while STP reconfigures.

Caution: Changing the interface speed and duplex mode configuration might shut down and re-enable the interface during the reconfiguration.

Setting the Interface Speed and Duplex Parameters

Beginning in privileged EXEC mode, follow these steps to set the speed and duplex mode for a physical interface.

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Specify the physical interface to be configured, and enter interface configuration mode.
3.	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
4.	**speed** {**10 | 100 | 1000 | auto** [**10** | **100** | **1000**] **| nonegotiate**}	Enter the appropriate speed parameter for the interface: ■ Enter **10**, **100**, or **1000** to set a specific speed for the interface. The **1000** keyword is available only for 10/100/1000 Mbps ports or SFP module ports with a 1000BASE-T SFP module. ■ Enter **auto** to enable the interface to autonegotiate speed with the connected device. If you use the **10**, **100**, or the **1000** keywords with the **auto** keyword, the port autonegotiates only at the specified speeds. ■ The **nonegotiate** keyword is available only for SFP module ports. SFP module ports operate only at 1000 Mbps but can be configured to not negotiate if connected to a device that does not support autonegotiation. **Note:** When a Cisco1000BASE-T SFP module is in the SFP module slot, the speed can be configured to **10**, **100**, **1000**, or to **auto**, but not to **nonegotiate**.

	Command	Purpose
5.	**duplex** {**auto** | **full** | **half**}	Enter the duplex parameter for the interface. **Note:** The default duplex mode is **full** when an FE SFP module is inserted. Enable half-duplex mode (for interfaces operating only at 10 or 100 Mbps). You cannot configure half-duplex mode for interfaces operating at 1000 Mbps. You can configure the duplex setting when the speed is set to **auto**. This command is not available on SFP module ports with these exceptions: ■ If a Cisco 1000BASE-T SFP module is inserted, you can configure duplex to **auto** or to **full**. ■ If a Cisco 100BASE-FX SFP module is inserted, you can configure duplex to **full** or to **half**. Although the **auto** keyword is available, it puts the interface in half-duplex mode (the default).
6.	**end**	Return to privileged EXEC mode.
7.	**show interfaces** *interface-id*	Display the interface speed and duplex mode configuration.
8.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no speed** and **no duplex** interface configuration commands to return the interface to the default speed and duplex settings (autonegotiate). To return all interface settings to the defaults, use the **default interface** *interface-id* interface configuration command.

This example shows how to set the interface speed to 10 Mbps and the duplex mode to half on a 10/100 Mbps port:

```
Switch# configure terminal
Switch(config)# interface fasttethernet0/3
Switch(config-if)# no shutdown
Switch(config-if)# speed 10
Switch(config-if)# duplex half
```

This example shows how to set the interface speed to 100 Mbps on a 10/100/1000 Mbps port:

```
Switch# configure terminal
Switch(config)# interface GigabitEthernet1/18
Switch(config-if)# speed 100
```

Configuring a Power Management Mode on a PoE Port

For most situations, the default configuration (auto mode) works well, providing plug-and-play operation. No further configuration is required. However, use the following procedure to give a PoE port higher priority, to make it data only, or to specify a maximum wattage to disallow high-power powered devices on a port.

Note: When you make PoE configuration changes, the port being configured drops power. Depending on the new configuration, the state of the other PoE ports, and the state of the power budget, the port might not be powered up again. For example, port 1 is in the auto and on state, and you configure it for static mode. The switch removes power from port 1, detects the powered device, and repowers the port. If port 1 is in the auto and on state and you configure it with a maximum wattage of 10 W, the switch removes power from the port and then redetects the powered device. The switch repowers the port only if the powered device is a Class 1, Class 2, or a Cisco-only powered device.

Beginning in privileged EXEC mode, follow these steps to configure a power management mode on a PoE-capable port:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Specify the physical port to be configured, and enter interface configuration mode.
3.	**power inline** {**auto** [**max** *max-wattage*] | **neve**r | **static** [**max** *max-wattage*]}	Configure the PoE mode on the port. The keywords have these meanings: ■ **auto**–Enable powered-device detection. If enough power is available, automatically allocate power to the PoE port after device detection. This is the default setting. (Optional) **max** *max-wattage*–Limit the power allowed on the port. The range is 4000 to 30000 mW. The default is 30000 mW. ■ **never**–Disable device detection, and disable power to the port. **Note:** If a port has a Cisco powered device connected to it, do not use the **power inline never** command to configure the port. A false link-up can occur, placing the port into an error-disabled state. ■ **static**–Enable powered-device detection. Pre-allocate (reserve) power for a port before the switch discovers the powered device. The switch reserves power for this port even when no device is connected and guarantees that power will be provided upon device detection. The switch allocates power to a port configured in static mode before it allocates power to a port configured in auto mode.
4.	**end**	Return to privileged EXEC mode.
5.	**show power inline** [i*nterface-id*]	Display PoE status for the switch or for the specified interface.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Budgeting Power for Devices Connected to a PoE Port

When Cisco powered devices are connected to PoE ports, the switch uses Cisco Discovery Protocol (CDP) to determine the *actual* power consumption of the devices, and the switch adjusts the power budget accordingly. The CDP protocol works with Cisco powered devices and does not apply to IEEE third-party powered devices. For these devices, when the switch grants a power request, the switch adjusts the power budget according to the powered-device IEEE classification. If the powered device is a Class 0 (class status unknown) or a Class 3, the switch budgets 30,000 milliwatts for the device, regardless of the actual amount of power needed. If the powered device reports a higher class than its actual consumption or does not support power classification (defaults to Class 0), the switch can power fewer devices because it uses the IEEE class information to track the global power budget.

By using the **power inline consumption** *wattage* configuration command, you can override the default power requirement specified by the IEEE classification. The difference between what is mandated by the IEEE classification and what is actually needed by the device is reclaimed into the global power budget for use by additional devices. You can then extend the switch power budget and use it more effectively.

Caution: You should carefully plan your switch power budget and make certain not to oversubscribe the power supply.

Note: When you manually configure the power budget, you must also consider the power loss over the cable between the switch and the powered device.

When you enter the **power inline consumption default** *wattage* or the **no power inline consumption default** global configuration command, or the **power inline consumption** *wattage* or the **no power inline consumption** interface configuration command this caution message appears:

```
%CAUTION: Interface interface-id: Misconfiguring the 'power inline consumption/allocation' command may
cause damage to the switch and void your warranty. Take precaution not to oversubscribe the power
supply.
 It is recommended to enable power policing if the switch supports it.
Refer to documentation.
```

If the power supply is over-subscribed to by up to 20 percent, the switch continues to operate but its reliability is reduced. If the power supply is subscribed to by more than 20 percent, the short-circuit protection circuitry triggers and shuts the switch down.

For more information about the IEEE power classifications, see Power over Ethernet Ports, page 21.

Beginning in privileged EXEC mode, follow these steps to configure the amount of power budgeted to a powered device connected to each PoE port on a switch:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**no cdp run**	(Optional) Disable CDP.
3.	**power inline consumption default** *wattage*	Configure the power consumption of powered devices connected to each the PoE port on the switch. The range for each device is 4000 to 15400 mW. The default is 15400 mW.
4.	**end**	Return to privileged EXEC mode.
5.	**show power inline consumption**	Display the power consumption status.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no power inline consumption default** global configuration command.

Beginning in privileged EXEC mode, follow these steps to configure amount of power budgeted to a powered device connected to a specific PoE port:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**no cdp run**	(Optional) Disable CDP.
3.	**interface** *interface-id*	Specify the physical port to be configured, and enter interface configuration mode.
4.	**power inline consumption** *wattage*	Configure the power consumption of a powered device connected to a PoE port on the switch. The range for each device is 4000 to 15400 mW. The default is 15400 mW. **Note:** When you use this command, we recommend you also enable power policing.
5.	**end**	Return to privileged EXEC mode.
6.	**show power inline consumption**	Display the power consumption status.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no power inline consumption** interface configuration command.

Beginning in privileged EXEC mode, follow these steps to configure amount of power budgeted to a powered device connected to a specific PoE port:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**no cdp run**	(Optional) Disable CDP.
3.	**interface** *interface-id*	Specify the physical port to be configured, and enter interface configuration mode.
4.	**power inline consumption** *wattage*	Configure the power consumption of a powered device connected to a PoE port on the switch. The range for each device is 4000 to 15400 mW. The default is 15400 mW.
5.	**end**	Return to privileged EXEC mode.
6.	**show power inline consumption**	Display the power consumption status.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no power inline consumption** interface configuration command.

Configuring IEEE 802.3x Flow Control

802.3x flow control enables connected Ethernet ports to control traffic rates during congestion by allowing congested nodes to pause link operation at the other end. If one port experiences congestion and cannot receive any more traffic, it notifies the other port by sending a pause frame to stop sending until the condition clears. Upon receipt of a pause frame, the sending device stops sending any data packets, which prevents any loss of data packets during the congestion period.

Note: Ports can receive, but not send, pause frames.

You use the **flowcontrol** interface configuration command to set the interface’s ability to **receive** pause frames to **on**, **off**, or **desired**. The default state is **off**.

When set to **desired**, an interface can operate with an attached device that is required to send flow-control packets or with an attached device that is not required to but can send flow-control packets.

These rules apply to 802.3x flow control settings on the device:

- **receive on** (or **desired**): The port cannot send pause frames but can operate with an attached device that is required to or can send pause frames; the port can receive pause frames.
- **receive off**: 802.3x flow control does not operate in either direction. In case of congestion, no indication is given to the link partner, and no pause frames are sent or received by either device.

Beginning in privileged EXEC mode, follow these steps to configure 802.3x flow control on an interface:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode
2.	**interface** *interface-id*	Specify the physical interface to be configured, and enter interface configuration mode.
3.	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
4.	**flowcontrol** {**receive**} {**on** | **off** | **desired**}	Configure the 802.3x flow control mode for the port.

	Command	Purpose
5.	**end**	Return to privileged EXEC mode.
6.	**show interfaces** *interface-id*	Verify the interface 802.3x flow control settings.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable 802.3x flow control, use the **flowcontrol receive off** interface configuration command.

This example shows how to enable 802.3x flow control on a port:

```
Switch# configure terminal
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# flowcontrol receive on
Switch(config-if)# end
```

Configuring Auto-MDIX on an Interface

When automatic medium-dependent interface crossover (auto-MDIX) is enabled on an interface, the interface automatically detects the required cable connection type (straight through or crossover) and configures the connection appropriately. When connecting switches without the auto-MDIX feature, you must use straight-through cables to connect to devices such as servers, workstations, or routers and crossover cables to connect to other switches or repeaters. With auto-MDIX enabled, you can use either type of cable to connect to other devices, and the interface automatically corrects for any incorrect cabling. For more information about cabling requirements, see the hardware installation guide.

When you enable auto-MDIX, you must also set the speed and duplex on the interface to **auto** so that the feature operates correctly. Auto-MDIX is supported on all 10/100 and 10/100/1000 Mbps interfaces and on Cisco 10/100/1000 BASE-T/TX SFP module interfaces. It is not supported on 1000 BASE-SX or -LX SFP module interfaces.

Table 4 shows the link states that result from auto-MDIX settings and correct and incorrect cabling.

Local Side Auto-MDIX	Remote Side Auto-MDIX	With Correct Cabling	With Incorrect Cabling
On	On	Link up	Link up
On	Off	Link up	Link up
Off	On	Link up	Link up
Off	Off	Link up	Link down

Beginning in privileged EXEC mode, follow these steps to configure auto-MDIX on an interface:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode
2.	**interface** *interface-id*	Specify the physical interface to be configured, and enter interface configuration mode.
3.	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
4.	**speed auto**	Configure the interface to autonegotiate speed with the connected device.
5.	**duplex auto**	Configure the interface to autonegotiate duplex mode with the connected device.
6.	**mdix auto force**	Enable auto-MDIX on the interface.

	Command	Purpose
7.	**end**	Return to privileged EXEC mode.
8.	**show controllers ethernet-controller** *interface-id* **phy**	Verify the operational state of the auto-MDIX feature on the interface.
9.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable auto-MDIX, use the **no mdix auto force** interface configuration command.

This example shows how to enable auto-MDIX on a port:

```
Switch# configure terminal
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# no shutdown
Switch(config-if)# speed auto
Switch(config-if)# duplex auto
Switch(config-if)# mdix auto force
Switch(config-if)# end
```

Adding a Description for an Interface

You can add a description about an interface to help you remember its function. The description appears in the output of these privileged EXEC commands: **show configuration, show running-config,** and **show interfaces**.

Beginning in privileged EXEC mode, follow these steps to add a description for an interface:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Specify the interface for which you are adding a description, and enter interface configuration mode.
3.	**description** *string*	Add a description (up to 240 characters) for an interface.
4.	**end**	Return to privileged EXEC mode.
5.	**show interfaces** *interface-id* **description** or **show running-config**	Verify your entry.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no description** interface configuration command to delete the description.

This example shows how to add a description on a port and how to verify the description:

```
Switch# config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface GigabitEthernet1/18
Switch(config-if)# description Connects to Marketing
Switch(config-if)# end
Switch# show interfaces GigabitEthernet1/18 description
Interface Status         Protocol Description
Gi 0/2    admin down     down     Connects to Marketing
```

Configuring Layer 3 Interfaces

The switch must be running the IP services image to support Layer 3 interfaces:

- SVIs: You should configure SVIs for any VLANs for which you want to route traffic. SVIs are created when you enter a VLAN ID following the **interface vlan** global configuration command. To delete an SVI, use the **no interface vlan** global configuration command. You cannot delete interface VLAN 1.

  When you create an SVI, it does not become active until it is associated with a physical port.

- Routed ports: Routed ports are physical ports configured to be in Layer 3 mode by using the **no switchport** interface configuration command.

- Layer 3 EtherChannel ports: EtherChannel interfaces made up of routed ports.

A Layer 3 switch can have an IP address assigned to each routed port and SVI.

There is no defined limit to the number of SVIs and routed ports that can be configured in a switch. However, the interrelationship between the number of SVIs and routed ports and the number of other features being configured might have an impact on CPU usage because of hardware limitations. If the switch is using maximum hardware resources, attempts to create a routed port or SVI have these results:

- If you try to create a new routed port, the switch generates a message that there are not enough resources to convert the interface to a routed port, and the interface remains as a switch port.

- If you try to create an extended-range VLAN, an error message is generated, and the extended-range VLAN is rejected.

- If the switch attempts to boot up with a configuration that has more VLANs and routed ports than hardware can support, the VLANs are created, but the routed ports are shut down, and the switch sends a message that this was due to insufficient hardware resources.

All Layer 3 interfaces require an IP address to route traffic. This procedure shows how to configure an interface as a Layer 3 interface and how to assign an IP address to an interface.

Note: If the physical port is in Layer 2 mode (the default), you must enter the **no switchport** interface configuration command to put the interface into Layer 3 mode. Entering a **no switchport** command disables and then re-enables the interface, which might generate messages on the device to which the interface is connected. Furthermore, when you put an interface that is in Layer 2 mode into Layer 3 mode, the previous configuration information related to the affected interface might be lost, and the interface is returned to its default configuration

Beginning in privileged EXEC mode, follow these steps to configure a Layer 3 interface:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** {{**fastethernet** | **gigabitethernet**} *interface-id*} | {**vlan** *vlan-id*} | {**port-channel** *port-channel-number*}	Specify the interface to be configured as a Layer 3 interface, and enter interface configuration mode.
3.	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
4.	**no switchport**	For physical ports only, enter Layer 3 mode.
5.	**ip address** *ip_address subnet_mask*	Configure the IP address and IP subnet.
6.	**no shutdown**	Enable the interface.

	Command	Purpose
7.	**end**	Return to privileged EXEC mode.
8.	**show interfaces** [*interface-id*] **show ip interface** [*interface-id*] **show running-config interface** [*interface-id*]	Verify the configuration.
9.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove an IP address from an interface, use the **no ip address** interface configuration command.

This example shows how to configure a port as a routed port and to assign it an IP address:

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface GigabitEthernet1/18
Switch(config-if)# no switchport
Switch(config-if)# ip address 192.20.135.21 255.255.255.0
```

Configuring the System MTU

The default maximum transmission unit (MTU) size for frames received and sent on all interfaces on the switch is 1500 bytes. You can increase the MTU size for all interfaces operating at 10 or 100 Mbps by using the **system mtu** global configuration command. You can increase the MTU size to support jumbo frames on all Gigabit Ethernet interfaces by using the **system mtu jumbo** global configuration command. You can change the MTU size for routed ports by using the **system mtu routing** global configuration command.

Note: You cannot configure a routing MTU size that exceeds the system MTU size. If you change the system MTU size to a value smaller than the currently configured routing MTU size, the configuration change is accepted, but not applied until the next switch reset. When the configuration change takes effect, the routing MTU size automatically defaults to the new system MTU size.

Gigabit Ethernet ports are not affected by the **system mtu** command. Fast Ethernet ports are not affected by the **system mtu jumbo** command because jumbo frames are not supported on 10/100 interfaces, including 100BASE-FX and 100BASE-BX SFP modules. If you do not configure the **system mtu jumbo** command, the setting of the **system mtu** command applies to all Gigabit Ethernet interfaces.

You cannot set the MTU size for an individual interface; you set it for all 10/100 or all Gigabit Ethernet interfaces on the switch. When you change the system MTU size, you must reset the switch before the new configuration takes effect. The **system mtu routing** command does not require a switch reset to take effect.

Note: The system MTU setting is saved in the switch environmental variable in NVRAM and becomes effective when the switch reloads. The MTU settings you enter with the **system mtu** and **system mtu jumbo** commands are not saved in the switch IOS configuration file, even if you enter the **copy running-config startup-config** privileged EXEC command. Therefore, if you use TFTP to configure a new switch by using a backup configuration file and want the system MTU to be other than the default, you must explicitly configure the **system mtu** and **system mtu jumbo** settings on the new switch and then reload the switch.

Frames sizes that can be received by the switch CPU are limited to 1998 bytes, no matter what value was entered with the **system mtu** or **system mtu jumbo** commands. Although frames that are forwarded or routed are typically not received by the CPU, in some cases packets are sent to the CPU, such as traffic sent to control traffic, SNMP, Telnet, or routing protocols.

Because the switch does not fragment packets, it drops:

- switched packets larger than the packet size supported on the *egress* interface

- routed packets larger than the routing MTU value

For example, if the **system mtu** value is 1998 bytes and the **system mtu jumbo** value is 5000 bytes, packets up to 5000 bytes can be received on interfaces operating at 1000 Mbps. However, although a packet larger than 1998 bytes can be received on an interface operating at 1000 Mbps, if its destination interface is operating at 10 or 100 Mbps, the packet is dropped.

Routed packets are subjected to MTU checks on the sending ports. The MTU value used for routed ports is derived from the configured **system mtu** value (not the **system mtu jumbo** value). That is, the routed MTU is never greater than the system MTU for any VLAN. The routing protocols use the system MTU value when negotiating adjacencies and the MTU of the link. For example, the Open Shortest Path First (OSPF) protocol uses this MTU value before setting up an adjacency with a peer router. To view the MTU value for routed packets for a specific VLAN, use the **show platform port-asic mvid** privileged EXEC command.

Note: If Layer 2 Gigabit Ethernet interfaces are configured to accept frames greater than the 10/100 interfaces, jumbo frames received on a Layer 2 Gigabit Ethernet interface and sent on a Layer 2 10/100 interface are dropped.

Beginning in privileged EXEC mode, follow these steps to change the MTU size for all 10/100 or Gigabit Ethernet interfaces:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**system mtu** *bytes*	(Optional) Change the MTU size for all interfaces on the switch that are operating at 10 or 100 Mbps. The range is 1500 to 1998 bytes; the default is 1500 bytes.
3.	**system mtu jumbo** *bytes*	(Optional) Change the MTU size for all Gigabit Ethernet interfaces on the switch. The range is 1500 to 9000 bytes; the default is 1500 bytes.
4.	**system mtu routing** *bytes*	(Optional) Change the system MTU for routed ports. The range is 1500 to the system MTU value, the maximum MTU that can be routed for all ports. Although larger packets can be accepted, they cannot be routed.
5.	**end**	Return to privileged EXEC mode.
6.	**copy running-config startup-config**	Save your entries in the configuration file.
7.	**reload**	Reload the operating system.

If you enter a value that is outside the allowed range for the specific type of interface, the value is not accepted.

Once the switch reloads, you can verify your settings by entering the **show system mtu** privileged EXEC command.

This example shows how to set the maximum packet size for a Gigabit Ethernet port to 1800 bytes:

```
Switch(config)# system mtu jumbo 1800
Switch(config)# exit
Switch# reload
```

This example shows the response when you try to set Gigabit Ethernet interfaces to an out-of-range number:

```
Switch(config)# system mtu jumbo 25000
                                   ^

% Invalid input detected at '^' marker.
```

Monitoring and Maintaining the Interfaces

These sections contain interface monitoring and maintenance information:

- Monitoring Interface Status, page 45
- Using FEFI to Maintain the Fiber FE Interfaces, page 46
- Clearing and Resetting Interfaces and Counters, page 47
- Shutting Down and Restarting the Interface, page 47

Monitoring Interface Status

Commands entered at the privileged EXEC prompt display information about the interface, including the versions of the software and the hardware, the configuration, and statistics about the interfaces. Table 8 on page 45 lists some of these interface monitoring commands. (You can display the full list of **show** commands by using the **show ?** command at the privileged EXEC prompt.)

Table 8 Show Commands for Interfaces

Command	Purpose
show interfaces [*interface-id*]	Display the status and configuration of all interfaces or a specific interface.
show interfaces *interface-id* **status** [**err-disabled**]	Display interface status or a list of interfaces in an error-disabled state.
show interfaces [*interface-id*] **switchport**	Display administrative and operational status of switching mode. You can use this command to find out if a port is in routing or in switching mode.
show interfaces [*interface-id*] **description**	Display the description configured on an interface or all interfaces and the interface status.
show ip interface [*interface-id*]	Display the usability status of all interfaces configured for IP routing or the specified interface.
show interface [*interface-id*] **stats**	Display the input and output packets by the switching path for the interface.

Table 8 Show Commands for Interfaces (continued)

Command	Purpose
show interfaces [*interface-id]* **transceiver** [**detail** | **dom-supported-list | module** *number* **| properties | threshold-table**]	Display these physical and operational status about an SFP module: - *interface-id*-(Optional) Display configuration and status for a specified physical interface. - **detail-**(Optional) Display calibration properties, including high and low numbers and any alarm information for any Digital Optical Monitoring (DoM)-capable transceiver if one is installed in the switch. - **dom-supported-list-**(Optional) List all supported DoM transceivers. - **module** *number*-(Optional) Limit display to interfaces on module on the switch. The range is 1 to 9. This option is not available if you entered a specific interface ID. - **properties-**(Optional) Display speed, duplex, and inline power settings on an interface - **threshold-table-**(Optional) Display alarm and warning threshold table
show interfaces [*interface-id*] [{**transceiver properties** | **detail**}] *module number*]	Display physical and operational status about an SFP module.
show port-type [**eni** | **nni** / **uni**]	Display interface type information for the Cisco ME switch.
show running-config interface [*interface-id*]	Display the running configuration in RAM for the interface.
show version	Display the hardware configuration, software version, the names and sources of configuration files, and the boot images.
show controllers ethernet-controller *interface-id* **phy**	Display the operational state of the auto-MDIX feature on the interface.

Using FEFI to Maintain the Fiber FE Interfaces

A far end fault is an error in the link that one station detects but the other does not, such as a disconnected Tx wire. In this example, the sending station still receives valid data and detects that the link is good through the link integrity monitor. The sending station does not detect that its own transmission is not being received by the other station. A 100BASE-FX station that detects a remote fault like this modifies its transmitted IDLE stream to send a special bit pattern (FEFI IDLE pattern) to inform the neighbor of the remote fault. The FEFI-IDLE pattern then triggers a shutdown of the remote port (notconnect).

Fiber FastEthernet hardware uses far end fault indication (FEFI) to bring the link down on both sides of the link in these situations. A similar function is provided by link negotiation for Gigabit Ethernet. FEFI is not supported on copper ports, which do not usually have issues in which one station can detect while the other cannot. Copper ports use Ethernet link pulses to monitor the link.

With FEFI, no forwarding loop occurs because there is no connectivity between the ports. If the link is up on one side and down on the other, however, blackholing of traffic might occur. Use Unidirectional Link Detection (UDLD) to prevent traffic blackholing.

Default FEFI Configuration

FEFI is enabled globally and not configurable on the switch, however it applies only to the fiber Fast Ethernet SFP interfaces on the switch.

Using FEFI on GE SFP Ports

FEFI can be used on the switch Gigabit Ethernet (GE) SFP ports when the GE ports are connected with 100FX/LX SFP transceiver type. However, using these SFP transceivers limits the GE interfaces to 100 MB/s.

Clearing and Resetting Interfaces and Counters

Table 9 on page 47 lists the privileged EXEC mode **clear** commands that you can use to clear counters and reset interfaces.

Table 9 Clear Commands for Interfaces

Command	Purpose
clear counters [*interface-id*]	Clear interface counters.
clear interface *interface-id*	Reset the hardware logic on an interface.
clear line [*number* | **console 0** | **vty** *number*]	Reset the hardware logic on an asynchronous serial line.

To clear the interface counters shown by the **show interfaces** privileged EXEC command, use the **clear counters** privileged EXEC command. The **clear counters** command clears all current interface counters from the interface unless you specify optional arguments that clear only a specific interface type from a specific interface number.

Note: The **clear counters** privileged EXEC command does not clear counters retrieved by using Simple Network Management Protocol (SNMP), but only those seen with the **show interface** privileged EXEC command.

Shutting Down and Restarting the Interface

Shutting down an interface disables all functions on the specified interface and marks the interface as unavailable on all monitoring command displays. This information is communicated to other network servers through all dynamic routing protocols. The interface is not mentioned in any routing updates.

Beginning in privileged EXEC mode, follow these steps to shut down an interface:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** {**vlan** *vlan-id*} | {{**fastethernet** | **gigabitethernet**} *interface-id*} | {**port-channel** *port-channel-number*}	Select the interface to be configured.
3.	**shutdown**	Shut down an interface.
4.	**end**	Return to privileged EXEC mode.
5.	**show running-config**	Verify your entry.

Use the **no shutdown** interface configuration command to enable an interface.

To verify that an interface is disabled, enter the **show interfaces** privileged EXEC command. A disabled interface is shown as *administratively down* in the display.

Configuring Switch Alarms

Information About Switch Alarms

The switch software monitors switch conditions on a per-port or a switch basis. If the conditions present on the switch or a port do not match the set parameters, the switch software triggers an alarm or a system message. By default, the switch software sends the system messages to a system message logging facility, or a *syslog* facility. You can also configure the switch to send Simple Network Management Protocol (SNMP) traps to an SNMP server.

Global Status Monitoring Alarms

The switch processes alarms related to temperature and power supply conditions, referred to as global or facility alarms.

Table 10 Global Status Monitoring Alarms

Alarm	Description
Power supply alarm	The switch monitors dual power supply levels. If there are two power supplies installed in the switch, an alarm triggers if a power supply fails. The alarm is automatically cleared when both power supplies are working. You can configure the power supply alarm to be connected to the hardware relays. For more information, see Configuring the Power Supply Alarms, page 51.
Temperature alarms	The switch contains one temperature sensor with a primary and secondary temperature setting. The sensor monitors the environmental conditions inside the switch. The primary and secondary temperature alarms can be set as follows: ■ The primary alarm is enabled automatically to trigger both at a low temperature, -4° F (-20° C) and a high temperature, 203° F (95° C). It cannot be disabled. By default, the primary temperature alarm is associated with the major relay. ■ The secondary alarm triggers when the system temperature is higher or lower than the configured high and low temperature thresholds. The secondary alarm is disabled by default. For more information, see Configuring the Switch Temperature Alarms, page 52.
SD-Card	By default the alarm is disabled.

FCS Error Hysteresis Threshold

The Ethernet standard calls for a maximum bit-error rate of 10^{-8}. The bit error-rate range is from 10^{-6} to 10^{-11}. The bit error-rate input to the switch is a positive exponent. If you want to configure the bit error-rate of 10^{-9}, enter the value 9 for the exponent. By default, the FCS bit error-rate is 10^{-8}.

You can set the FCS error hysteresis threshold to prevent the toggle of the alarm when the actual bit-error rate fluctuates near the configured rate. The hysteresis threshold is defined as the ratio between the alarm clear threshold to the alarm set threshold, expressed as a percentage value.

For example, if the FCS bit error-rate alarm value is configured to 10^{-8}, that value is the alarm set threshold. To set the alarm clear threshold at $5*10^{-10}$, the hysteresis, value *h*, is determined as follows:

h = alarm clear threshold / alarm set threshold

$h = 5*10^{-10} / 10^{-8} = 5*10^{-2} = 0.05 = 5$ percent

The FCS hysteresis threshold is applied to all ports on the switch. The allowable range is from 1 to 10 percent. The default value is 10 percent. See Configuring the FCS Bit Error Rate Alarm, page 52 for more information.

Port Status Monitoring Alarms

The switch can also monitor the status of the Ethernet ports and generate alarm messages based on the alarms listed in Table 11 on page 50. To save user time and effort, it supports changeable alarm configurations by using alarm profiles. You can create a number of profiles and assign one of these profiles to each Ethernet port.

Alarm profiles provide a mechanism for you to enable or disable alarm conditions for a port and associate the alarm conditions with one or both alarm relays. You can also use alarm profiles to set alarm conditions to send alarm traps to an SNMP server and system messages to a syslog server. The alarm profile *defaultPort* is applied to all interfaces in the factory configuration (by default).

Note: You can associate multiple alarms to one relay or one alarm to both relays.

Table 11 on page 50 lists the port status monitoring alarms and their descriptions and functions. Each fault condition is assigned a severity level based on the Cisco IOS System Error Message Severity Level.

Table 11 Port Status Monitoring Alarms

Alarm List ID	Alarm	Description
1	Link Fault alarm	The switch generates a link fault alarm when problems with a port physical layer cause unreliable data transmission. A typical link fault condition is loss of signal or clock. The link fault alarm is cleared automatically when the link fault condition is cleared. The severity for this alarm is *error condition*, level 3.
2	Port not Forwarding alarm	The switch generates a port not-forwarding alarm when a port is not forwarding packets. This alarm is cleared automatically when the port begins to forward packets. The severity for this alarm is *warning*, level 4.
3	Port not Operating alarm	The switch generates a port not-operating alarm when a port fails during the startup self-test. When triggered, the port not-operating alarm is only cleared when the switch is restarted and the port is operational. The severity for this alarm is *error condition*, level 3.
4	FCS Bit Error Rate alarm	The switch generates an FCS bit error-rate alarm when the actual FCS bit error-rate is close to the configured rate. You can set the FCS bit error-rate by using the interface configuration CLI for each of the ports. See Configuring the FCS Bit Error Rate Alarm, page 52 for more information. The severity for this alarm is *error condition*, level 3.

Triggering Alarm Options

The switch supports these methods for triggering alarms:

- Configurable Relay

  The switch is equipped with one independent alarm relay that can be triggered by alarms for global, port status and SD flash card conditions. You can configure the relay to send a fault signal to an external alarm device, such as a bell, light, or other signaling device. You can associate any alarm condition with the alarm relay. Each fault condition is assigned a severity level based on the Cisco IOS System Error Message Severity Level.

  See Configuring the Power Supply Alarms, page 51 for more information on configuring the relay.

- SNMP Traps

SNMP is an application-layer protocol that provides a message format for communication between managers and agents. The SNMP system consists of an SNMP manager, an SNMP agent, and a management information base (MIB).

The **snmp-server enable traps** command can be changed so that the user can send alarm traps to an SNMP server. You can use alarm profiles to set environmental or port status alarm conditions to send SNMP alarm traps. See Enabling SNMP Traps, page 54 for more information.

- Syslog Messages

  You can use alarm profiles to send system messages to a syslog server. See Configuring the Power Supply Alarms, page 51 for more information.

Default Switch Alarm Settings

Table 12 Default Switch Alarm Settings

	Alarm	Default Setting
Global	Power supply alarm	Enabled in switch single power mode. No alarm. In dual-power supply mode, the default alarm notification is a system message to the console.
	Primary temperature alarm	Enabled for switch temperature range of 203ºF (95ºC) maximum to -4° F (-20ºC) minimum. The primary switch temperature alarm is associated with the major relay.
	Secondary temperature alarm	Disabled.
	Output relay mode alarm	Normally deenergized. The alarm output has switched off or is in an off state.
Port	Link fault alarm	Disabled on all interfaces.
	Port not forwarding alarm	Disabled on all interfaces.
	Port not operating alarm	Enabled on all interfaces.
	FCS bit error rate alarm	Disabled on all interfaces.

How to Configure Switch Alarms

Configuring the Power Supply Alarms

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**power-supply dual**	Configures dual power supplies.
3.	**alarm facility power-supply disable**	Disables the power supply alarm.
4.	**alarm facility power-supply relay major**	Associates the power supply alarm to the relay.
5.	**alarm facility power-supply notifies**	Sends power supply alarm traps to an SNMP server.
6.	**alarm facility power-supply syslog**	Sends power supply alarm traps to a syslog server.
7.	**end**	Returns to privileged EXEC mode.

	Command	Purpose
8.	show env power	Displays the switch power status.
9.	show facility-alarm status	Displays all generated alarms for the switch.
10.	**show alarm settings**	Verifies the configuration.
11.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring the Switch Temperature Alarms

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**alarm facility temperature** {**primary** | **secondary**} **high** *threshold*	Sets the high temperature threshold value. Set the threshold from –238° F (–150° C) to 572° F (300° C).
3.	**alarm facility temperature primary low** *threshold*	Sets the low temperature threshold value. Set the threshold from –328° F (–200° C) to 482° F (250° C).
4.	**end**	Returns to privileged EXEC mode.
5.	**show alarm settings**	Verifies the configuration.
6.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Associating the Temperature Alarms to a Relay

By default, the primary temperature alarm is associated to the relay. You can use the **alarm facility temperature** global configuration command to associate the primary temperature alarm to an SNMP trap, or a syslog message, or to associate the secondary temperature alarm to the relay, an SNMP trap, or a syslog message.

Note: The single relay on the switch is called the major relay.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**alarm facility temperature** {**primary** | **secondary**} **relay major**	Associates the primary or secondary temperature alarm to the relay.
3.	**alarm facility temperature** {**primary** | **secondary**} **notifies**	Sends primary or secondary temperature alarm traps to an SNMP server.
4.	**alarm facility temperature** {**primary** | **secondary**} **syslog**	Sends primary or secondary temperature alarm traps to a syslog server. Uses the **no alarm facility temperature secondary** command to disable the secondary temperature alarm.
5.	**end**	Returns to privileged EXEC mode.
6.	**show alarm settings**	Verifies the configuration.
7.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring the FCS Bit Error Rate Alarm

Setting the FCS Error Threshold

The switch generates an FCS bit error-rate alarm when the actual rate is close to the configured rate.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Enters the interface to be configured, and enters interface configuration mode.
3.	**fcs-threshold** ***value***	Sets the FCS error rate. For *value*, the range is 6 to 11 to set a maximum bit error rate of 10^{-6} to 10^{-11}. By default, the FCS bit error rate is 10^{-8}.
4.	**end**	Returns to privileged EXEC mode.
5.	**show fcs-threshold**	Verifies the setting.
6.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Setting the FCS Error Hysteresis Threshold

The hysteresis setting prevents the toggle of an alarm when the actual bit error-rate fluctuates near the configured rate. The FCS hysteresis threshold is applied to all ports of a switch.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**alarm facility fcs-hysteresis** *percentage*	Sets the hysteresis percentage for the switch. For *percentage*, the range is 1 to 10. The default value is 10 percent.
3.	**end**	Returns to privileged EXEC mode.
4.	**show running config**	Verifies the configuration.
5.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Alarm Profiles

Creating an Alarm Profile

You can use the **alarm profile** global configuration command to create an alarm profile or to modify an existing profile. When you create a new alarm profile, none of the alarms are enabled.

Note: The only alarm enabled in the *defaultPort* profile is the Port not operating alarm.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**alarm profile** ***name***	Creates the new profile or identifies an existing profile, and enters alarm profile configuration mode.
3.	**end**	Returns to privileged EXEC mode.
4.	**show alarm profile** ***name***	Verifies the configuration.
5.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Modifying an Alarm Profile

You can modify an alarm profile from alarm profile configuration mode.

You can enter more than one alarm type separated by a space.

Command	Purpose
alarm {fcs-error | link-fault | not-forwarding | not-operating}	(Optional) Adds or modifies alarm parameters for a specific alarm.
notifies {fcs-error | link-fault | not-forwarding | not-operating}	(Optional) Configures the alarm to send an SNMP trap to an SNMP server.
relay-major {fcs-error | link-fault | not-forwarding | not-operating}	(Optional) Configures the alarm to send an alarm trap to the relay.
syslog {fcs-error | link-fault | not-forwarding | not-operating}	(Optional) Configures the alarm to send an alarm trap to a syslog server.

Attaching an Alarm Profile to a Specific Port

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *port interface*	Enters interface configuration mode.
3.	**alarm-profile** ***name***	Attaches the specified profile to the interface.
4.	**end**	Returns to privileged EXEC mode.
5.	**show alarm profile**	Verifies the configuration.
6.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Enabling SNMP Traps

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**snmp-server enable traps alarms**	Enables the switch to send SNMP traps.
3.	**end**	Returns to privileged EXEC mode.
4.	**show alarm settings**	Verifies the configuration.
5.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Monitoring and Maintaining Switch Alarms Status

Table 13 Commands for Displaying Global and Port Alarm Status

Command	Purpose
show alarm description ports	Displays an alarm number and its text description.
show alarm profile [*name*]	Displays all alarm profiles in the system or a specified profile.

Table 13 Commands for Displaying Global and Port Alarm Status (continued)

Command	Purpose
show alarm settings	Displays all global alarm settings on the switch.
show env {**alarm-contact** | **all** | **power** | **temperature**}	Displays the status of environmental facilities on the switch.
show facility-alarm status [**critical** | **info** | **major | minor**]	Displays generated alarms on the switch.

Configuration Examples for Switch Alarms

Configuring External Alarms: Example

This example configures alarm input 1 named *door sensor* to assert a major alarm when the door circuit is closed and then displays the status and configuration for all alarms:

```
Switch(config)# alarm contact 1 description door sensor
Switch(config)# alarm contact 1 severity major
Switch(config)# alarm contact 1 trigger closed
Switch(config)# end
Switch(config)# show env alarm-contact
Switch# show env alarm-contact

ALARM CONTACT 1
   Status:      not asserted
   Description: door sensor
   Severity:    major
   Trigger:     closed
ALARM CONTACT 2
   Status:      not asserted
   Description: external alarm contact 2
   Severity:    minor
   Trigger:     closed
```

Associating Temperature Alarms to a Relay: Examples

This example sets the secondary temperature alarm to the major relay, with a high temperature threshold value of 113ºF (45ºC). All alarms and traps associated with this alarm are sent to a syslog server and an SNMP server.

```
Switch(config) # alarm facility temperature secondary high 45
Switch(config) # alarm facility temperature secondary relay major
Switch(config) # alarm facility temperature secondary syslog
Switch(config) # alarm facility temperature secondary notifies
```

This example sets the first (primary) temperature alarm to the major relay. All alarms and traps associated with this alarm are sent to a syslog server.

```
Switch(config) # alarm facility temperature primary syslog
Switch(config) # alarm facility temperature primary relay major
```

Configuring a Dual Power Supply: Examples

This example shows how to configure two power supplies:

```
Switch# configure terminal
```

```
Switch(config)# power-supply dual
```

These examples show how to display information when two power supplies are not present which results in a triggered alarm.

```
Switch# show facility-alarm status
Source Severity Description Relay Time
Switch MAJOR 5 Redundant Pwr missing or failed NONE Mar 01
1993 00:23:52

Switch# show env power
POWER SUPPLY A is DC OK
POWER SUPPLY B is DC FAULTY <--

Switch# show hard led
SWITCH: 1
SYSTEM: GREEN
ALARM : ALT_RED_BLACK <--
```

Displaying Alarm Settings: Example

```
Switch# show alarm settings
Alarm relay mode: De-energized
Power Supply
     Alarm      Enabled
     Relay
     Notifies  Disabled
     Syslog     Enabled
Temperature-Primary
     Alarm      Enabled
     Thresholds  MAX: 95C  MIN: -20C
     Relay      MAJ
     Notifies  Enabled
     Syslog     Enabled
Temperature-Secondary
     Alarm      Disabled
     Threshold
     Relay
     Notifies  Disabled
     Syslog     Disabled
License-File-Corrupt
     Alarm      Enabled
     Relay
     Notifies  Enabled
     Syslog     Enabled

Switch# show alarm settings
Alarm relay mode: De-energized
Power Supply
         Alarm                   Enabled
         Relay
         Notifies                Disabled
         Syslog                  Enabled
Temperature-Primary
         Alarm                   Enabled
         Thresholds              MAX: 95C                MIN:  -20C
         Relay                   MAJ
         Notifies                Enabled
         Syslog                  Enabled
Temperature-Secondary
         Alarm                   Disabled
         Threshold
         Relay
```

```
          Notifies                 Disabled
          Syslog                   Disabled
SD-Card
          Alarm                    Disabled
          Relay
          Notifies                 Disabled
          Syslog                   Enabled
Input-Alarm 1
          Alarm                    Enabled
          Relay
          Notifies                 Disabled
          Syslog                   Enabled
Input-Alarm 2
          Alarm                    Enabled
          Relay
          Notifies                 Disabled
          Syslog                   Enabled
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Alarm input and output ports.	*Hardware Installation Guide Hardware Technical Guide*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Performing Switch Setup Configuration

Restrictions for Performing Switch Setup Configuration

- The DHCP-based autoconfiguration with a saved configuration process stops if there is not at least one Layer 3 interface in an up state without an assigned IP address in the network.
- Unless you configure a timeout, the DHCP-based autoconfiguration with a saved configuration feature tries indefinitely to download an IP address.
- The auto-install process stops if a configuration file cannot be downloaded or it the configuration file is corrupted.

Note: The configuration file that is downloaded from TFTP is merged with the existing configuration in the running configuration but is not saved in the NVRAM unless you enter the **write memory** or **copy running-configuration startup-configuration** privileged EXEC command. Note that if the downloaded configuration is saved to the startup configuration, the feature is not triggered during subsequent system restarts.

Information About Performing Switch Setup Configuration

This chapter describes how to perform your initial switch configuration tasks that include IP address assignments and DHCP autoconfiguration.

Switch Boot Process

To start your switch, you need to follow the procedures in the *Hardware Installation Guide Hardware Technical Guide* for installing and powering on the switch and for setting up the initial switch configuration (IP address, subnet mask, default gateway, secret and Telnet passwords, and so forth).

The normal boot process involves the operation of the boot loader software, which performs these activities:

- Performs low-level CPU initialization—Initializes the CPU registers, which control where physical memory is mapped, its quantity and its speed.
- Performs power-on self-test (POST) for the CPU subsystem—Tests the CPU DRAM and the portion of the flash device that makes up the flash file system.
- Initializes the flash memory card file system on the system board.
- Loads a default operating system software image into memory and boots up the switch.

The boot loader provides access to the flash file system before the operating system is loaded. Normally, the boot loader is used only to load, uncompress, and launch the operating system. After the boot loader gives the operating system control of the CPU, the boot loader is not active until the next system reset or power-on.

The switch supports a flash memory card that makes it possible to replace a failed switch without reconfiguring the new switch. The slot for the flash memory card is hot swappable and front-accessed. A cover protects the flash card and holds the card firmly in place. The cover is hinged and closed with a captive screw, which prevents the card from coming loose and protects against shock and vibration.

Use the **show flash:** privileged EXEC command to display the flash memory card file settings. For information about how to remove or replace the flash memory card on the switch, see the *Hardware Installation Guide.*

The boot loader also provides trap-door access into the system if the operating system has problems serious enough that it cannot be used. The trap-door mechanism provides enough access to the system so that if it is necessary, you can format the flash file system, reinstall the operating system software image by using the Xmodem Protocol, recover from a lost or forgotten password, and finally restart the operating system.

Note: You can disable password recovery.

Before you can assign switch information, make sure you have connected a PC or terminal to the console port, and configured the PC or terminal-emulation software baud rate and character format to match these of the switch console port:

- Baud rate default is 9600.
- Data bits default is 8.

  If the data bits option is set to 8, set the parity option to none.
- Stop bits default is 1.
- Parity settings default is none.

Default Switch Boot Settings

Feature	Default Setting
Operating system software image	The switch attempts to automatically boot up the system using information in the BOOT environment variable. If the variable is not set, the switch attempts to load and execute the first executable image it can by performing a recursive, depth-first search throughout the flash file system. The Cisco IOS image is stored in a directory that has the same name as the image file (excluding the .bin extension). In a depth-first search of a directory, each encountered subdirectory is completely searched before continuing the search in the original directory.
Configuration file	Configured switches use the *config.text* file stored on the system board in flash memory. A new switch has no configuration file.

Switch Boot Optimization

The normal switch boot process involves a memory test, file system check (FSCK), and power-on self-test (POST).

The **boot fast** command in global configuration mode is enabled by default to permit switch boot optimization, which disables these tests and minimizes the bootup time. However, after a system crash this feature is automatically disabled.

Note - With boot fast enabled, the expected boot time for the switch is from 2 to 3 minutes, depending on image and configuration size.

Reload sequences occur immediately if your switch is set up to automatically bring up the system by using information in the BOOT environment variable. Otherwise, these reload sequences occur after you enter the manual **boot** command in bootloader configuration mode.

First Reload

The switch disables the boot fast feature and displays the following warning message:

```
“Reloading with boot fast feature disabled”
```

After the system message appears, the system saves the crash information and automatically resets itself for the next reload cycle.

Second Reload

The boot loader performs its normal full memory test and FSCK check with LED status progress. If the memory and FSCK tests are successful, the system performs additional POST tests and the results are displayed on the console.

The boot fast feature is reenabled after the system comes up successfully.

Switch Information Assignment

You can assign IP information through the switch setup program, through a DHCP server, or manually.

Use the switch setup program if you want to be prompted for specific IP information. With this program, you can also configure a hostname and an enable secret password. The program gives you the option of assigning a Telnet password (to provide security during remote management) and configuring your switch as a command or member switch of a cluster or as a standalone switch. For more information about the setup program, see the *Hardware Installation Guide Hardware Technical Guide*.

Use a DHCP server for centralized control and automatic assignment of IP information after the server is configured.

Note: If you are using DHCP, do not respond to any of the questions in the setup program until the switch receives the dynamically assigned IP address and reads the configuration file.

If you are an experienced user familiar with the switch configuration steps, manually configure the switch. Otherwise, use the setup program.

Switch Default Settings

Feature	Default Setting
IP address and subnet mask	No IP address or subnet mask is defined.
Default gateway	No default gateway is defined.
Enable secret password	No password is defined.
Hostname	The factory-assigned default hostname is *Switch*.
Telnet password	No password is defined.
Cluster command switch functionality	Disabled.
Cluster name	No cluster name is defined.
Manual boot	No.
Boot optimization	Enabled.

DHCP-Based Autoconfiguration Overview

DHCP provides configuration information to Internet hosts and internetworking devices. This protocol consists of two components: one for delivering configuration parameters from a DHCP server to a device and a mechanism for allocating network addresses to devices. DHCP is built on a client-server model, in which designated DHCP servers allocate network addresses and deliver configuration parameters to dynamically configured devices. The switch can act as both a DHCP client and a DHCP server.

During DHCP-based autoconfiguration, your switch (DHCP client) is automatically configured at startup with IP address information and a configuration file.

With DHCP-based autoconfiguration, no DHCP client-side configuration is needed on your switch. However, you need to configure the DHCP server for various lease options associated with IP addresses. If you are using DHCP to relay the configuration file location on the network, you might also need to configure a Trivial File Transfer Protocol (TFTP) server and a Domain Name System (DNS) server.

The DHCP server for your switch can be on the same LAN or on a different LAN than the switch. If the DHCP server is running on a different LAN, you should configure a DHCP relay device between your switch and the DHCP server. A relay device forwards broadcast traffic between two directly connected LANs. A router does not forward broadcast packets, but it forwards packets based on the destination IP address in the received packet.

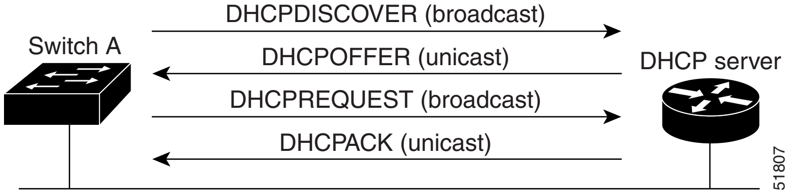
DHCP-based autoconfiguration replaces the BOOTP client functionality on your switch.

DHCP Client Request Process

When you boot up your switch, the DHCP client is invoked and requests configuration information from a DHCP server when the configuration file is not present on the switch. If the configuration file is present and the configuration includes the **ip address dhcp** interface configuration command on specific routed interfaces, the DHCP client is invoked and requests the IP address information for those interfaces.

Figure 2 on page 63 shows the sequence of messages that are exchanged between the DHCP client and the DHCP server.

Figure 2 DHCP Client and Server Message Exchange



The client, Switch A, broadcasts a DHCPDISCOVER message to locate a DHCP server. The DHCP server offers configuration parameters (such as an IP address, subnet mask, gateway IP address, DNS IP address, a lease for the IP address, and so forth) to the client in a DHCPOFFER unicast message.

In a DHCPREQUEST broadcast message, the client returns a formal request for the offered configuration information to the DHCP server. The formal request is broadcast so that all other DHCP servers that received the DHCPDISCOVER broadcast message from the client can reclaim the IP addresses that they offered to the client.

The DHCP server confirms that the IP address has been allocated to the client by returning a DHCPACK unicast message to the client. With this message, the client and server are bound, and the client uses configuration information received from the server. The amount of information the switch receives depends on how you configure the DHCP serverd in conjunction with the TFTP server. For more information, see TFTP Server, page 65.

If the configuration parameters sent to the client in the DHCPOFFER unicast message are invalid (a configuration error exists), the client returns a DHCPDECLINE broadcast message to the DHCP server.

The DHCP server sends the client a DHCPNAK denial broadcast message, which means that the offered configuration parameters have not been assigned, that an error has occurred during the negotiation of the parameters, or that the client has been slow in responding to the DHCPOFFER message. (The DHCP server assigned the parameters to another client.)

A DHCP client might receive offers from multiple DHCP or BOOTP servers and can accept any of the offers; however, the client usually accepts the first offer it receives. The offer from the DHCP server is not a guarantee that the IP address is allocated to the switch. However, the server usually reserves the address until the client has had a chance to formally request the address. If the switch accepts replies from a BOOTP server and configures itself, the switch broadcasts, instead of unicasts, TFTP requests to obtain the switch configuration file.

The DHCP hostname option allows a group of switches to obtain hostnames and a standard configuration from the central management DHCP server. A client (switch) includes in its DCHPDISCOVER message an option 12 field used to request a hostname and other configuration parameters from the DHCP server. The configuration files on all clients are identical except for their DHCP-obtained hostnames.

If a client has a default hostname (the **hostname** *name* global configuration command is not configured or the **no hostname** global configuration command is entered to remove the hostname), the DHCP hostname option is not included in the packet when you enter the **ip address dhcp** interface configuration command. In this case, if the client receives the DCHP hostname option from the DHCP interaction while acquiring an IP address for an interface, the client accepts the DHCP hostname option and sets the flag to show that the system now has a hostname configured.

DHCP-Based Autoconfiguration and Image Update

You can use the DHCP image upgrade features to configure a DHCP server to download both a new image and a new configuration file to one or more switches in a network. This helps ensure that each new switch added to a network receives the same image and configuration.

There are two types of DHCP image upgrades: DHCP autoconfiguration and DHCP auto-image update.

DHCP Autoconfiguration

DHCP autoconfiguration downloads a configuration file to one or more switches in your network from a DHCP server. The downloaded configuration file becomes the running configuration of the switch. It does not over write the bootup configuration saved in the flash, until you reload the switch.

DHCP Auto-Image Update

You can use DHCP auto-image upgrade with DHCP autoconfiguration to download both a configuration *and* a new image to one or more switches in your network. The switch (or switches) downloading the new configuration and the new image can be blank (or only have a default factory configuration loaded).

If the new configuration is downloaded to a switch that already has a configuration, the downloaded configuration is appended to the configuration file stored on the switch. (Any existing configuration is not overwritten by the downloaded one.)

Note: To enable a DHCP auto-image update on the switch, the TFTP server where the image and configuration files are located must be configured with the correct option 67 (the configuration filename), option 66 (the DHCP server hostname) option 150 (the TFTP server address), and option 125 (description of the file) settings.

For procedures to configure the switch as a DHCP server, see DHCP Server Configuration Guidelines, page 64 and the “Configuring DHCP” section of the “IP addressing and Services” section of the *Cisco IOS IP DHCP Configuration Guide*, Release 15.0.

After you install the switch in your network, the auto-image update feature starts. The downloaded configuration file is saved in the running configuration of the switch, and the new image is downloaded and installed on the switch. When you reboot the switch, the configuration is stored in the saved configuration on the switch.

DHCP Server Configuration Guidelines

Follow these guidelines if you are configuring a device as a DHCP server:

- Configure the DHCP server with reserved leases that are bound to each switch by the switch hardware address.
- If you want the switch to receive IP address information, you must configure the DHCP server with these lease options:
  - IP address of the client (required)
  - Subnet mask of the client (required)
  - Router IP address (default gateway address to be used by the switch) (required)
  - DNS server IP address (optional)
- If you want the switch to receive the configuration file from a TFTP server, you must configure the DHCP server with these lease options:
  - TFTP server name (required)
  - Boot filename (the name of the configuration file that the client needs) (recommended)
  - Hostname (optional)
- Depending on the settings of the DHCP server, the switch can receive IP address information, the configuration file, or both.
- If you do not configure the DHCP server with the lease options described previously, it replies to client requests with only those parameters that are configured.

If the IP address and the subnet mask are not in the reply, the switch is not configured. If the router IP address or the TFTP server name are not found, the switch might send broadcast, instead of unicast, TFTP requests. Unavailability of other lease options does not affect autoconfiguration.

- The switch can act as a DHCP server. By default, the Cisco IOS DHCP server and relay agent features are enabled on your switch but are not configured. These features are not operational. If your DHCP server is a Cisco device, for additional information about configuring DHCP, see the “Configuring DHCP” section of the “IP Addressing and Services” section of the *Cisco IOS IP Configuration Guide* on Cisco.com.

TFTP Server

Based on the DHCP server configuration, the switch attempts to download one or more configuration files from the TFTP server. If you configured the DHCP server to respond to the switch with all the options required for IP connectivity to the TFTP server, and if you configured the DHCP server with a TFTP server name, address, and configuration filename, the switch attempts to download the specified configuration file from the specified TFTP server.

If you did not specify the configuration filename, the TFTP server, or if the configuration file could not be downloaded, the switch attempts to download a configuration file by using various combinations of filenames and TFTP server addresses. The files include the specified configuration filename (if any) and these files: network-config, cisconet.cfg, and *hostname*.config (or *hostname*.cfg), where *hostname* is the switch’s current hostname. The TFTP server addresses used include the specified TFTP server address (if any) and the broadcast address (255.255.255.255).

For the switch to successfully download a configuration file, the TFTP server must contain one or more configuration files in its base directory. The files can include these files:

- The configuration file named in the DHCP reply (the actual switch configuration file).
- The network-confg or the cisconet.cfg file (known as the default configuration files).
- The router-confg or the ciscortr.cfg file (These files contain commands common to all switches. Normally, if the DHCP and TFTP servers are properly configured, these files are not accessed.)

If you specify the TFTP server name in the DHCP server-lease database, you must also configure the TFTP server name-to-IP-address mapping in the DNS-server database.

If the TFTP server to be used is on a different LAN from the switch, or if it is to be accessed by the switch through the broadcast address (which occurs if the DHCP server response does not contain all the required information described previously), a relay must be configured to forward the TFTP packets to the TFTP server. For more information, see Relay Device, page 65. The preferred solution is to configure the DHCP server with all the required information.

DNS Server

The DHCP server uses the DNS server to resolve the TFTP server name to an IP address. You must configure the TFTP server name-to-IP address map on the DNS server. The TFTP server contains the configuration files for the switch.

You can configure the IP addresses of the DNS servers in the lease database of the DHCP server from where the DHCP replies will retrieve them. You can enter up to two DNS server IP addresses in the lease database.

The DNS server can be on the same or on a different LAN as the switch. If it is on a different LAN, the switch must be able to access it through a router.

Relay Device

You must configure a relay device, also referred to as a *relay agent*, when a switch sends broadcast packets that require a response from a host on a different LAN. Examples of broadcast packets that the switch might send are DHCP, DNS, and in some cases, TFTP packets. You must configure this relay device to forward received broadcast packets on an interface to the destination host.

If the relay device is a Cisco router, enable IP routing (**ip routing** global configuration command), and configure helper addresses by using the **ip helper-address** interface configuration command.

For example, in Figure 3 on page 66, configure the router interfaces as follows:

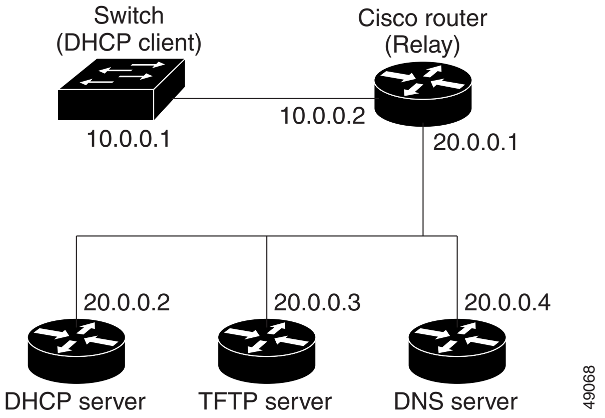
On interface 10.0.0.2:

```
router(config-if)# ip helper-address 20.0.0.2
router(config-if)# ip helper-address 20.0.0.3
router(config-if)# ip helper-address 20.0.0.4
```

On interface 20.0.0.1:

```
router(config-if)# ip helper-address 10.0.0.1
```

Figure 3 Relay Device Used in Autoconfiguration



How to Obtain Configuration Files

Depending on the availability of the IP address and the configuration filename in the DHCP reserved lease, the switch obtains its configuration information in these ways:

- The IP address and the configuration filename is reserved for the switch and provided in the DHCP reply (one-file read method).

  The switch receives its IP address, subnet mask, TFTP server address, and the configuration filename from the DHCP server. The switch sends a unicast message to the TFTP server to retrieve the named configuration file from the base directory of the server and upon receipt, it completes its boot-up process.

- The IP address and the configuration filename is reserved for the switch, but the TFTP server address is not provided in the DHCP reply (one-file read method).

  The switch receives its IP address, subnet mask, and the configuration filename from the DHCP server. The switch sends a broadcast message to a TFTP server to retrieve the named configuration file from the base directory of the server, and upon receipt, it completes its boot-up process.

- Only the IP address is reserved for the switch and provided in the DHCP reply. The configuration filename is not provided (two-file read method).

  The switch receives its IP address, subnet mask, and the TFTP server address from the DHCP server. The switch sends a unicast message to the TFTP server to retrieve the network-confg or cisconet.cfg default configuration file. (If the network-confg file cannot be read, the switch reads the cisconet.cfg file.)

The default configuration file contains the hostnames-to-IP-address mapping for the switch. The switch fills its host table with the information in the file and obtains its hostname. If the hostname is not found in the file, the switch uses the hostname in the DHCP reply. If the hostname is not specified in the DHCP reply, the switch uses the default *Switch* as its hostname.

After obtaining its hostname from the default configuration file or the DHCP reply, the switch reads the configuration file that has the same name as its hostname (*hostname*-confg or *hostname*.cfg, depending on whether network-confg or cisconet.cfg was read earlier) from the TFTP server. If the cisconet.cfg file is read, the filename of the host is truncated to eight characters.

If the switch cannot read the network-confg, cisconet.cfg, or the hostname file, it reads the router-confg file. If the switch cannot read the router-confg file, it reads the ciscortr.cfg file.

Note: The switch broadcasts TFTP server requests if the TFTP server is not obtained from the DHCP replies, if all attempts to read the configuration file through unicast transmissions fail, or if the TFTP server name cannot be resolved to an IP address.

How to Control Environment Variables

With a normally operating switch, you enter the boot loader mode only through a switch console connection configured for 9600 b/s. Unplug the switch power cord, and press the switch **Mode** button while reconnecting the power cord. You can release the **Mode** button a second or two after the LED above port 1 turns off. Then the boot loader *switch:* prompt appears.

The switch boot loader software provides support for nonvolatile environment variables, which can be used to control how the boot loader or any other software running on the system behaves. Boot loader environment variables are similar to environment variables that can be set on UNIX or DOS systems.

Environment variables that have values are stored in flash memory outside of the flash file system.

Each line in these files contains an environment variable name and an equal sign followed by the value of the variable. A variable has no value if it is not listed in this file; it has a value if it is listed in the file even if the value is a null string. A variable that is set to a null string (for example, “ ”) is a variable with a value. Many environment variables are predefined and have default values.

Environment variables store two kinds of data:

- Data that controls code, which does not read the Cisco IOS configuration file. For example, the name of a boot loader helper file, which extends or patches the functionality of the boot loader can be stored as an environment variable.
- Data that controls code, which is responsible for reading the Cisco IOS configuration file. For example, the name of the Cisco IOS configuration file can be stored as an environment variable.

You can change the settings of the environment variables by accessing the boot loader or by using Cisco IOS commands. Under normal circumstances, it is not necessary to alter the setting of the environment variables.

Common Environment Variables

Table 14 on page 68 describes the function of the most common environment variables.

Table 14 Environment Variables

Variable	Boot Loader Command	Cisco IOS Global Configuration Command
BOOT	**set BOOT** *filesystem***:/***file-url ...* A semicolon-separated list of executable files to try to load and execute when automatically booting. If the BOOT environment variable is not set, the system attempts to load and execute the first executable image it can find by using a recursive, depth-first search through the flash file system. If the BOOT variable is set but the specified images cannot be loaded, the system attempts to boot the first bootable file that it can find in the flash file system.	**boot system** f*ilesystem:/file-url ...* Specifies the Cisco IOS image to load during the next boot cycle. This command changes the setting of the BOOT environment variable.
MANUAL_BOOT	**set MANUAL_BOOT yes** Decides whether the switch automatically or manually boots up. Valid values are 1, yes, 0, and no. If it is set to no or 0, the boot loader attempts to automatically boot up the system. If it is set to anything else, you must manually boot up the switch from the boot loader mode.	**boot manual** Enables manually booting up the switch during the next boot cycle and changes the setting of the MANUAL_BOOT environment variable. The next time you reboot the system, the switch is in boot loader mode. To boot up the system, use the **boot flash:***filesystem***:/***file-url* boot loader command, and specify the name of the bootable image.
CONFIG_FILE	**set CONFIG_FILE flash:/***file-url* Changes the filename that Cisco IOS uses to read and write a nonvolatile copy of the system configuration.	**boot config-file flash:/***file-url* Specifies the filename that Cisco IOS uses to read and write a nonvolatile copy of the system configuration. This command changes the CONFIG_FILE environment variable.

Scheduled Reload of the Software Image

You can schedule a reload of the software image to occur on the switch at a later time (for example, late at night or during the weekend when the switch is used less), or you can synchronize a reload network-wide (for example, to perform a software upgrade on all switches in the network).

Note: A scheduled reload must take place within approximately 24 days.

You have these reload options:

- Software reload to take effect in the specified minutes or hours and minutes. The reload must take place within approximately 24 days. You can specify the reason for the reload in a string up to 255 characters in length.
- Software reload to take place at the specified time (using a 24-hour clock). If you specify the month and day, the reload is scheduled to take place at the specified time and date. If you do not specify the month and day, the reload takes place at the specified time on the current day (if the specified time is later than the current time) or on the next day (if the specified time is earlier than the current time). Specifying 00:00 schedules the reload for midnight.

The **reload** command halts the system. If the system is not set to manually boot up, it reboots itself.

If your switch is configured for manual booting, do not reload it from a virtual terminal. This restriction prevents the switch from entering the boot loader mode and thereby taking it from the remote user’s control.

If you modify your configuration file, the switch prompts you to save the configuration before reloading. During the save operation, the system requests whether you want to proceed with the save if the CONFIG_FILE environment variable points to a startup configuration file that no longer exists. If you proceed in this situation, the system enters setup mode upon reload.

To cancel a previously scheduled reload, use the **reload cancel** privileged EXEC command.

How to Perform Switch Setup Configuration

Using DHCP to download a new image and a new configuration to a switch requires that you configure at least two switches. One switch acts as a DHCP and TFTP server and the second switch (client) is configured to download either a new configuration file or a new configuration file and a new image file.

Configuring DHCP Autoconfiguration (Only Configuration File)

This task describes how to configure DHCP autoconfiguration of the TFTP and DHCP settings on a new switch to download a new configuration file.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**ip dhcp** *poolname*	Creates a name for the DHCP Server address pool, and enters DHCP pool configuration mode.
3.	**bootfile** *filename*	Specifies the name of the configuration file that is used as a boot image.
4.	**network** *network-number mask prefix-length*	Specifies the subnet network number and mask of the DHCP address pool. **Note:** The prefix length specifies the number of bits that comprise the address prefix. The prefix is an alternative way of specifying the network mask of the client. The prefix length must be preceded by a forward slash (/).
5.	**default-router** *address*	Specifies the IP address of the default router for a DHCP client.
6.	**option 150** *address*	Specifies the IP address of the TFTP server.
7.	**exit**	Returns to global configuration mode.
8.	**tftp-server flash**:*filename.text*	Specifies the configuration file on the TFTP server.
9.	**interface** *interface-id*	Specifies the address of the client that will receive the configuration file.
10.	**no switchport**	Puts the interface into Layer 3 mode.
11.	**ip address** *address mask*	Specifies the IP address and mask for the interface.
12.	**end**	Returns to privileged EXEC mode.
13.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring DHCP Auto-Image Update (Configuration File and Image)

This task describes DHCP autoconfiguration to configure TFTP and DHCP settings on a new switch to download a new image and a new configuration file.

Before You Begin

You must create a text file (for example, autoinstall_dhcp) that will be uploaded to the switch. In the text file, put the name of the image that you want to download. This image must be a tar and not a bin file.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**ip dhcp pool** *name*	Creates a name for the DHCP server address pool and enters DHCP pool configuration mode.
3.	**bootfile** *filename*	Specifies the name of the file that is used as a boot image.
4.	**network** *network-number mask prefix-length*	Specifies the subnet network number and mask of the DHCP address pool. **Note:** The prefix length specifies the number of bits that comprise the address prefix. The prefix is an alternative way of specifying the network mask of the client. The prefix length must be preceded by a forward slash (/).
5.	**default-router** *address*	Specifies the IP address of the default router for a DHCP client.
6.	**option 150** *address*	Specifies the IP address of the TFTP server.
7.	**option 125** *hex*	Specifies the path to the text file that describes the path to the image file.
8.	**copy tftp flash** *filename.txt*	Uploads the text file to the switch.
9.	**copy tftp flash** *imagename.tar*	Uploads the tar file for the new image to the switch.
10.	**exit**	Returns to global configuration mode.
11.	**tftp-server flash:***config.text*	Specifies the Cisco IOS configuration file on the TFTP server.
12.	**tftp-server flash:***imagename.tar*	Specifies the image name on the TFTP server.
13.	**tftp-server flash:***filename.txt*	Specifies the text file that contains the name of the image file to download.
14.	**interface** *interface-id*	Specifies the address of the client that will receive the configuration file.
15.	**no switchport**	Puts the interface into Layer 3 mode.
16.	**ip address** *address mask*	Specifies the IP address and mask for the interface.
17.	**end**	Returns to privileged EXEC mode.
18.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring the Client

You should only configure and enable the Layer 3 interface. Do not assign an IP address or DHCP-based autoconfiguration with a saved configuration.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**boot host dhcp**	Enables autoconfiguration with a saved configuration.
3.	**boot host retry timeout** *timeout-value*	(Optional) Sets the amount of time the system tries to download a configuration file. **Note:** If you do not set a timeout, the system tries indefinitely to obtain an IP address from the DHCP server.

	Command	Purpose
4.	**banner config-save ^C** *warning-message* **^C**	(Optional) Creates warning messages to be displayed when you try to save the configuration file to NVRAM.
5.	**end**	Returns to privileged EXEC mode.
6.	**show boot**	Verifies the configuration.

Manually Assigning IP Information on a Routed Port

This task describes how to manually assign IP information on a Layer 3 routed port.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *type id*	Enters interface configuration mode.
3.	**no switchport**	Configures an interface into Layer 3 mode.
4.	**ip address** *address mask*	Specifies the IP address and mask for the interface.
5.	**exit**	Returns to global configuration mode.
6.	**ip default-gateway** *ip-address*	Enters the IP address of the next-hop router interface that is directly connected to the switch where a default gateway is being configured. The default gateway receives IP packets with unresolved destination IP addresses from the switch. Once the default gateway is configured, the switch has connectivity to the remote networks with which a host needs to communicate. **Note:** When your switch is configured to route with IP, it does not need to have a default gateway set.
7.	**end**	Returns to privileged EXEC mode.
8.	**show ip redirects**	Verifies the configured default gateway.
9.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Manually Assigning IP Information to SVIs

This task describes how to manually assign IP information to multiple switched virtual interfaces (SVIs).

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface vlan** *vlan-id*	Enters interface configuration mode, and enters the VLAN to which the IP information is assigned. The VLAN range is 1 to 4096.
3.	**ip address** *ip-address subnet-mask*	Enters the IP address and subnet mask.
4.	**exit**	Returns to global configuration mode.

	Command	Purpose
5.	**ip default-gateway** *ip-address*	Enters the IP address of the next-hop router interface that is directly connected to the switch where a default gateway is being configured. The default gateway receives IP packets with unresolved destination IP addresses from the switch. Once the default gateway is configured, the switch has connectivity to the remote networks with which a host needs to communicate. **Note:** When your switch is configured to route with IP, it does not need to have a default gateway set.
6.	**end**	Returns to privileged EXEC mode.
7.	**show interfaces vlan** *vlan-id*	Verifies the configured IP address.
8.	**show ip redirects**	Verifies the configured default gateway.
9.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Modifying the Startup Configuration

Specifying the Filename to Read and Write the System Configuration

By default, the Cisco IOS software uses the *config.text* file to read and write a nonvolatile copy of the system configuration. However, you can specify a different filename, which will be loaded during the next boot-up cycle.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**boot config-file flash:/***file-url*	Specifies the configuration file to load during the next boot-up cycle. For *file-url*, specify the path (directory) and the configuration filename. Filenames and directory names are case sensitive.
3.	**end**	Returns to privileged EXEC mode.
4.	**show boot**	Verifies your entries. The **boot config-file** global configuration command changes the setting of the CONFIG_FILE environment variable.
5.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Manually Booting the Switch

By default, the switch automatically boots up; however, you can configure it to manually boot up.

Before You Begin

Use a standalone switch for this task.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**boot manual**	Enables the switch to manually boot up during the next boot cycle.
3.	**end**	Returns to privileged EXEC mode.
4.	**show boot**	Verifies your entries. The **boot manual** global command changes the setting of the MANUAL_BOOT environment variable. The next time you reboot the system, the switch is in boot loader mode, shown by the switch: prompt. To boot up the system, use the **boot** *filesystem*:**/***file-url* boot loader command. ■ For *filesystem*:, use **flash:** for the system board flash device. ■ For *file-url*, specify the path (directory) and the name of the bootable image. Filenames and directory names are case sensitive.
5.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Booting a Specific Software Image

By default, the switch attempts to automatically boot up the system using information in the BOOT environment variable. If this variable is not set, the switch attempts to load and execute the first executable image it can by performing a recursive, depth-first search throughout the flash file system. In a depth-first search of a directory, each encountered subdirectory is completely searched before continuing the search in the original directory. However, you can specify a specific image to boot up.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**boot system** *filesystem*:**/***file-url*	Configures the switch to boot a specific image in flash memory during the next boot cycle. ■ For *filesystem*:, use **flash:** for the system board flash device. ■ For *file-url*, specify the path (directory) and the name of the bootable image. Filenames and directory names are case sensitive.
3.	**end**	Returns to privileged EXEC mode.
4.	**show boot**	Verifies your entries. The **boot system** global command changes the setting of the BOOT environment variable. During the next boot cycle, the switch attempts to automatically boot up the system using information in the BOOT environment variable.
5.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Monitoring Switch Setup Configuration

Verifying the Switch Running Configuration

You can check the configuration settings that you entered or changes that you made by entering this privileged EXEC command:

```
Switch# show running-config
Building configuration...

Current configuration: 1363 bytes
!
version 12.2
no service pad
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Switch A
!
enable secret 5 $1$ej9.$DMUvAUnZOAmvmgqBEzIxE0
!
.
<output truncated>
.
interface GigabitEthernet1/17
no switchport
ip address 172.20.137.50 255.255.255.0
!
interface GigabitEthernet1/18
mvr type source

<output truncated>

...!
interface VLAN1
 ip address 172.20.137.50 255.255.255.0
 no ip directed-broadcast
!
ip default-gateway 172.20.137.1 !
!
snmp-server community private RW
snmp-server community public RO
snmp-server community private@es0 RW
snmp-server community public@es0 RO
snmp-server chassis-id 0x12
!
end
```

To store the configuration or changes you have made to your startup configuration in flash memory, enter this privileged EXEC command:

```
Switch# copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
```

This command saves the configuration settings that you made. If you fail to do this, your configuration will be lost the next time you reload the system. To display information stored in the NVRAM section of flash memory, use the **show startup-config** or **more startup-config** privileged EXEC command.

For more information about alternative locations from which to copy the configuration file, see Working with the Cisco IOS File System, Configuration Files, and Software Images, page 1035

Configuration Examples for Performing Switch Setup Configuration

Retrieving IP Information Using DHCP-Based Autoconfiguration: Example

Switch A reads its configuration file as follows:

- It obtains its IP address 10.0.0.21 from the DHCP server.
- If no configuration filename is given in the DHCP server reply, Switch A reads the network-confg file from the base directory of the TFTP server.
- It adds the contents of the network-confg file to its host table.
- It reads its host table by indexing its IP address 10.0.0.21 to its hostname (switcha).
- It reads the configuration file that corresponds to its hostname; for example, it reads *switch1-confg* from the TFTP server.

Switches B through D retrieve their configuration files and IP addresses in the same way.

Figure 4 on page 75 shows a sample network for retrieving IP information by using DHCP-based autoconfiguration.

Figure 4 DHCP-Based Autoconfiguration Network Example

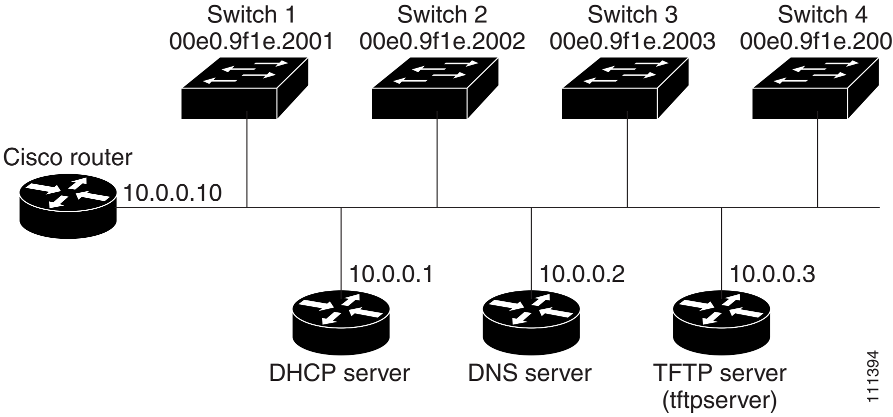


Table 15 on page 75 shows the configuration of the reserved leases on the DHCP server.

Table 15 DHCP Server Configuration

	Switch A	Switch B	Switch C	Switch D
Binding key (hardware address)	00e0.9f1e.2001	00e0.9f1e.2002	00e0.9f1e.2003	00e0.9f1e.2004
IP address	10.0.0.21	10.0.0.22	10.0.0.23	10.0.0.24
Subnet mask	255.255.255.0	255.255.255.0	255.255.255.0	255.255.255.0
Router address	10.0.0.10	10.0.0.10	10.0.0.10	10.0.0.10
DNS server address	10.0.0.2	10.0.0.2	10.0.0.2	10.0.0.2

Table 15 DHCP Server Configuration (continued)

	Switch A	Switch B	Switch C	Switch D
TFTP server name	*tftpserver* or *10.0.0.3*	*tftpserver* or *10.0.0.3*	*tftpserver* or *10.0.0.3*	*tftpserver* or *10.0.0.3*
Boot filename (configuration file) (optional)	switcha-confg	switchb-confg	switchc-confg	switchd-confg
Hostname (optional)	switcha	switchb	switchc	switchd

DNS Server Configuration

The DNS server maps the TFTP server name *tftpserver* to IP address 10.0.0.3.

TFTP Server Configuration (on UNIX)

The TFTP server base directory is set to /tftpserver/work/. This directory contains the network-confg file used in the two-file read method. This file contains the hostname to be assigned to the switch based on its IP address. The base directory also contains a configuration file for each switch (*switcha-confg*, *switchb-confg*, and so forth) as shown in this display:

```
prompt> cd /tftpserver/work/
prompt> ls
network-confg
switcha-confg
switchb-confg
switchc-confg
switchd-confg
prompt> cat network-confg
ip host switcha 10.0.0.21
ip host switchb 10.0.0.22
ip host switchc 10.0.0.23
ip host switchd 10.0.0.24
```

DHCP Client Configuration

No configuration file is present on Switch A through Switch D.

Scheduling Software Image Reload: Examples

This example shows how to reload the software on the switch on the current day at 7:30 p.m:

```
Switch# reload at 19:30
Reload scheduled for 19:30:00 UTC Wed Jun 5 1996 (in 2 hours and 25 minutes)
Proceed with reload? [confirm]
```

This example shows how to reload the software on the switch at a future time:

```
Switch# reload at 02:00 jun 20
Reload scheduled for 02:00:00 UTC Thu Jun 20 1996 (in 344 hours and 53 minutes)
Proceed with reload? [confirm]
```

To cancel a previously scheduled reload, use the **reload cancel** privileged EXEC command.

Configuring DHCP Auto-Image Update: Example

```
Switch# configure terminal
Switch(config)# ip dhcp pool pool1
Switch(dhcp-config)# network 10.10.10.0 255.255.255.0
Switch(dhcp-config)# bootfile config-boot.text
Switch(dhcp-config)# default-router 10.10.10.1
Switch(dhcp-config)# option 150 10.10.10.1
Switch(dhcp-config)# exit
Switch(config)# tftp-server flash:config-boot.text
Switch(config)# interface GigabitEthernet1/18
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.10.10.1 255.255.255.0
Switch(config-if)# end
```

Configuring a Switch as a DHCP Server: Example

This example shows how to configure a switch as a DHCP server so it downloads a configuration file:

```
Switch# config terminal
Switch(config)# ip dhcp pool pool1
Switch(dhcp-config)# network 10.10.10.0 255.255.255.0
Switch(dhcp-config)# bootfile config-boot.text
Switch(dhcp-config)# default-router 10.10.10.1
Switch(dhcp-config)# option 150 10.10.10.1
Switch(dhcp-config)# option 125 hex 0000.0009.0a05.08661.7574.6f69.6e73.7461.6c6c.5f64.686370
Switch(dhcp-config)# exit
Switch(config)# tftp-server flash:config-boot.text
Switch(config)# tftp-server flash:c-ipservices-mz.122-44.3.SE.tar
Switch(config)# tftp-server flash:ies-lanbase-tar.122-44.EX.tar
Switch(config)# tftp-server flash:boot-config.text
Switch(config)# tftp-server flash: autoinstall_dhcp
Switch(config)# interface GigabitEthernet1/18
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.10.10.1 255.255.255.0
Switch(config-if)# end
```

Configuring Client to Download Files from DHCP Server

This example uses a Layer 3 SVI interface on VLAN 99 to enable DHCP-based autoconfiguration with a saved configuration:

```
Switch# configure terminal
Switch(conf)# boot host dhcp
Switch(conf)# boot host retry timeout 300
Switch(conf)# banner config-save ^C Caution - Saving Configuration File to NVRAM May Cause You to
Nolonger Automatically Download Configuration Files at Reboot^C
Switch(config)# vlan 99
Switch(config-vlan)# interface vlan 99
Switch(config-if)# no shutdown
Switch(config-if)# end
Switch# show boot
BOOT path-list:
Config file:          flash:/config.text
Private Config file:  flash:/private-config.text
Enable Break:         no
Manual Boot:          no
HELPER path-list:
NVRAM/Config file
      buffer size:    32768
Timeout for Config
          Download:    300 seconds
Config Download
       via DHCP:       enabled (next boot: enabled)
Switch#
```

Additional References

The following sections provide references related to switch administration:

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Configuring Cisco IOS Configuration Engine

Prerequisites for Configuring Cisco IOS Configuration Engine

Set the CNS DeviceID

- When using the Cisco Configuration Engine user interface, you must first set the DeviceID field to the hostname value that the switch acquires after, not before, you use the **cns config initial** global configuration command at the switch. Otherwise, subsequent **cns config partial** global configuration command operations malfunction.

Enable Automated CNS Configuration

- To enable automated CNS configuration of the switch, you must first complete the prerequisites in Table 1. When you complete them, power on the switch. At the **setup** prompt, you do not need to enter a command. The switch begins the initial configuration as described in the Initial Configuration, page 83. When the full configuration file is loaded on your switch, you do not need to do anything else.

Device	Required Configuration
Access switch	Factory default (no configuration file)
Distribution switch	■ IP helper address ■ Enable DHCP relay agent ■ IP routing (if used as default gateway)
DHCP server	■ IP address assignment ■ TFTP server IP address ■ Path to bootstrap configuration file on the TFTP server ■ Default gateway IP address
TFTP server	■ A bootstrap configuration file that includes the CNS configuration commands that enable the switch to communicate with the Configuration Engine ■ The switch configured to use either the switch MAC address or the serial number (instead of the default hostname) to generate the ConfigID and EventID ■ The CNS event agent configured to push the configuration file to the switch
CNS Configuration Engine	One or more templates for each type of device, with the ConfigID of the device mapped to the template

Information About Configuring Cisco IOS Configuration Engine

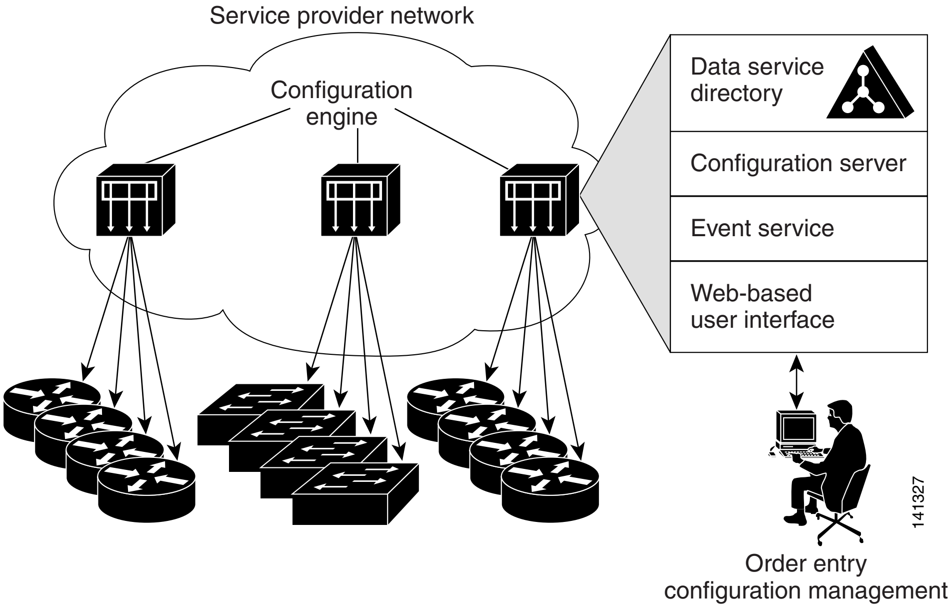
Cisco Configuration Engine is network management software that acts as a configuration service for automating the deployment and management of network devices and services (see Figure 5 on page 81). Each Cisco Configuration Engine service manages a group of Cisco devices (switches and routers) and the services that they deliver, storing their configurations and delivering them as needed. Cisco Configuration Engine automates initial configurations and configuration updates by generating device-specific configuration changes, sending them to the device, executing the configuration change, and logging the results.

Cisco Configuration Engine supports standalone and server modes and has these CNS components:

- Configuration service (web server, file manager, and namespace mapping server)
- Event service (event gateway)
- Data service directory (data models and schema)

In standalone mode, Cisco Configuration Engine supports an embedded directory service. In this mode, no external directory or other data store is required. In server mode, Cisco Configuration Engine supports a user-defined external directory.

Figure 5 Configuration Engine Architectural Overview



Configuration Service

Configuration Service is the core component of Cisco Configuration Engine. It consists of a configuration server that works with Cisco IOS CNS agents on the switch. Configuration Service delivers device and service configurations to the switch for initial configuration and mass reconfiguration by logical groups. Switches receive their initial configuration from the Configuration Service when they start up on the network for the first time.

Configuration Service uses CNS Event Service to send and receive configuration change events and to send success and failure notifications.

The configuration server is a web server that uses configuration templates and the device-specific configuration information stored in the embedded (standalone mode) or remote (server mode) directory.

Configuration templates are text files containing static configuration information in the form of CLI commands. In the templates, variables are specified using Lightweight Directory Access Protocol (LDAP) URLs that reference the device-specific configuration information stored in a directory.

The Cisco IOS agent can perform a syntax check on received configuration files and publish events to show the success or failure of the syntax check. The configuration agent can either apply configurations immediately or delay the application until receipt of a synchronization event from the configuration server.

Event Service

Cisco Configuration Engine uses Event Service for receipt and generation of configuration events. The event agent is on the switch and facilitates the communication between the switch and the event gateway on Configuration Engine.

Event Service is a highly capable publish-and-subscribe communication method. Event Service uses subject-based addressing to send messages to their destinations. Subject-based addressing conventions define a simple, uniform namespace for messages and their destinations.

NameSpace Mapper

Configuration Engine includes NameSpace Mapper (NSM), which provides a lookup service for managing logical groups of devices based on application, device or group ID, and event.

Cisco IOS devices recognize only event subject names that match those configured in Cisco IOS software; for example, cisco.cns.config.load. You can use the namespace mapping service to designate events by using any desired naming convention. When you have populated your data store with your subject names, NSM changes your event subject-name strings to those known by Cisco IOS.

For a subscriber, when given a unique device ID and event, the namespace mapping service returns a set of events to which to subscribe. Similarly, for a publisher, when given a unique group ID, device ID, and event, the mapping service returns a set of events on which to publish.

CNS IDs and Device Hostnames

Configuration Engine assumes that a unique identifier is associated with each configured switch. This unique identifier can take on multiple synonyms, where each synonym is unique within a particular namespace. The event service uses namespace content for subject-based addressing of messages.

Configuration Engine intersects two namespaces, one for the event bus and the other for the configuration server. Within the scope of the configuration server namespace, the term *ConfigID* is the unique identifier for a device. Within the scope of the event bus namespace, the term *DeviceID* is the CNS unique identifier for a device.

Because Configuration Engine uses both the event bus and the configuration server to provide configurations to devices, you must define both ConfigID and Device ID for each configured switch.

Within the scope of a single instance of the configuration server, no two configured switches can share the same value for ConfigID. Within the scope of a single instance of the event bus, no two configured switches can share the same value for DeviceID.

ConfigID

Each configured switch has a unique ConfigID, which serves as the key into the Configuration Engine directory for the corresponding set of switch CLI attributes. The ConfigID defined on the switch must match the ConfigID for the corresponding switch definition on Configuration Engine.

The ConfigID is fixed at startup time and cannot be changed until the device restarts, even if the switch hostname is reconfigured.

DeviceID

Each configured switch participating on the event bus has a unique DeviceID, which is analogous to the switch source address so that the switch can be targeted as a specific destination on the bus. All switches configured with the **cns config partial** global configuration command must access the event bus. Therefore, the DeviceID, as originated on the switch, must match the DeviceID of the corresponding switch definition in Configuration Engine.

The origin of the DeviceID is defined by the Cisco IOS hostname of the switch. However, the DeviceID variable and its usage reside within the event gateway adjacent to the switch.

The logical Cisco IOS termination point on the event bus is embedded in the event gateway, which in turn functions as a proxy on behalf of the switch. The event gateway represents the switch and its corresponding DeviceID to the event bus.

The switch declares its hostname to the event gateway immediately after the successful connection to the event gateway. The event gateway couples the DeviceID value to the Cisco IOS hostname each time this connection is established. The event gateway caches this DeviceID value for the duration of its connection to the switch.

Hostname and DeviceID Interaction

The DeviceID is fixed at the time of the connection to the event gateway and does not change even when the switch hostname is reconfigured.

When changing the switch hostname on the switch, the only way to refresh the DeviceID is to break the connection between the switch and the event gateway. Enter the **no cns event** global configuration command followed by the **cns event** global configuration command.

When the connection is reestablished, the switch sends its modified hostname to the event gateway. The event gateway redefines the DeviceID to the new value.

Using Hostname, DeviceID, and ConfigID

In standalone mode, when a hostname value is set for a switch, the configuration server uses the hostname as the DeviceID when an event is sent on hostname. If the hostname has not been set, the event is sent on the cn=<*value*> of the device.

In server mode, the hostname is not used. In this mode, the unique DeviceID attribute is always used for sending an event on the bus. If this attribute is not set, you cannot update the switch.

These and other associated attributes (tag value pairs) are set when you run **Setup** on Configuration Engine.

Cisco IOS Agents

The CNS event agent feature allows the switch to publish and subscribe to events on the event bus and works with the Cisco IOS agent.

Initial Configuration

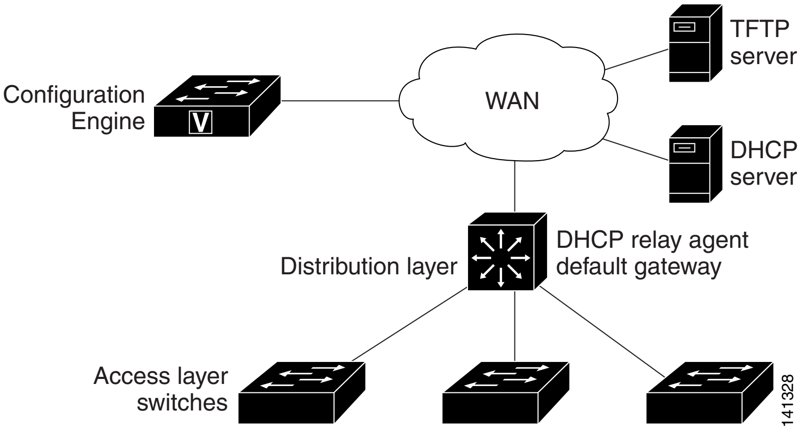
When the switch first comes up, it attempts to get an IP address by broadcasting a DHCP request on the network. Assuming there is no DHCP server on the subnet, the distribution switch acts as a DHCP relay agent and forwards the request to the DHCP server. Upon receiving the request, the DHCP server assigns an IP address to the new switch and includes the TFTP server IP address, the path to the bootstrap configuration file, and the default gateway IP address in a unicast reply to the DHCP relay agent. The DHCP relay agent forwards the reply to the switch.

The switch automatically configures the assigned IP address on interface VLAN 1 (the default) and downloads the bootstrap configuration file from the TFTP server. Upon successful download of the bootstrap configuration file, the switch loads the file in its running configuration.

The Cisco IOS agents initiate communication with Configuration Engine by using the appropriate ConfigID and EventID. Configuration Engine maps the ConfigID to a template and downloads the full configuration file to the switch.

Figure 6 on page 84 shows a sample network configuration for retrieving the initial bootstrap configuration file by using DHCP-based autoconfiguration.

Figure 6 Initial Configuration Overview



Incremental (Partial) Configuration

After the network is running, new services can be added by using the Cisco IOS agent. Incremental (partial) configurations can be sent to the switch. The actual configuration can be sent as an event payload by way of the event gateway (push operation) or as a signal event that triggers the switch to initiate a pull operation.

The switch can check the syntax of the configuration before applying it. If the syntax is correct, the switch applies the incremental configuration and publishes an event that signals success to the configuration server. If the switch does not apply the incremental configuration, it publishes an event showing an error status. When the switch has applied the incremental configuration, it can write it to NVRAM or wait until signaled to do so.

Synchronized Configuration

When the switch receives a configuration, it can defer application of the configuration upon receipt of a write-signal event. The write-signal event tells the switch not to save the updated configuration into its NVRAM. The switch uses the updated configuration as its running configuration. This ensures that the switch configuration is synchronized with other network activities before saving the configuration in NVRAM for use at the next reboot.

How to Configure Cisco IOS Configuration Engine

Configuring Cisco IOS Agents

CNS Event Agent and Cisco IOS CNS Agent embedded in the Cisco IOS software on the switch allows the switch to be connected and automatically configured. Both agents must be enabled and the CNS configuration can be **initial** or **partial**. The partial configuration allows you to use Configuration Engine to remotely send incremental configuration to the switch.

Enabling CNS Event Agent

Before You Begin

You must enable CNS Event Agent on the switch before you enable Cisco IOS CNS Agent.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**cns event** {*hostname* | *ip-address*} [*port-number*] [**backup**] [**failover-time** *seconds*] [**keepalive** *seconds retry-count*] [**reconnect** *time*] [**source** *ip-address*]	Enables the event agent, and enters the gateway parameters. ■ {*hostname* | *ip-address*}—Enters either the hostname or the IP address of the event gateway. ■ (Optional) *port number*—Enters the port number for the event gateway. The default port number is 11011. ■ (Optional) **backup**—Shows that this is the backup gateway. (If omitted, this is the primary gateway.) ■ (Optional) **failover-time** *seconds*—Enters how long the switch waits for the primary gateway route after the route to the backup gateway is established. ■ (Optional) **keepalive** *seconds*—Enters how often the switch sends keepalive messages. For *retry-count*, enters the number of unanswered keepalive messages that the switch sends before the connection is terminated. The default for each is 0. ■ (Optional) **reconnect** *time*—Enters the maximum time interval that the switch waits before trying to reconnect to the event gateway. ■ (Optional) **source** *ip-address*—Enters the source IP address of this device. **Note:** Though visible in the command-line help string, the **encrypt** and the **clock-timeout** *time* keywords are not supported.
3.	**end**	Returns to privileged EXEC mode.
4.	**show cns event connections**	Verifies information about the event agent.

Enabling Cisco IOS CNS Agent and an Initial Configuration

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**cns template connect** *name*	Enters CNS template connect configuration mode, and specifies the name of the CNS connect template.
3.	**cli** *config-text*	Enters a command line for the CNS connect template. Repeat this step for each command line in the template.
4.		Repeat Steps 2 to 3 to configure another CNS connect template.
5.	**exit**	Returns to global configuration mode.

	Command	Purpose
6.	**cns connect** *name* [**retries** *number*] [**retry-interval** *seconds*] [**sleep** *seconds*] [**timeout** *seconds*]	Enters CNS connect configuration mode, specifies the name of the CNS connect profile, and defines the profile parameters. The switch uses the CNS connect profile to connect to Configuration Engine. - (Optional) **retries** *number*–Enters the number of connection retries. The range is 1 to 30. The default is 3. - (Optional) **retry-interva**l *seconds*–Enters the interval between successive connection attempts to the Configuration Engine. The range is 1 to 40 seconds. The default is 10 seconds. - (Optional) **sleep** *seconds*–Enters the amount of time before which the first connection attempt occurs. The range is 0 to 250 seconds. The default is 0. - (Optional) **timeout** *seconds*–Enters the amount of time after which the connection attempts end. The range is 10 to 2000 seconds*.* The default is 120.
7.	**discover** {**controller** *controller-type* | **dlci** [**subinterface** *subinterface-number*] | **interface** [*interface-type*] | **line** *line-type*}	Specifies the interface parameters in the CNS connect profile. - **controller** *controller-type*–Enters the controller type. - **dlci**–Enters the active data-link connection identifiers (DLCIs). (Optional) **subinterface** *subinterface-number*–Specifies the point-to-point subinterface number that is used to search for active DLCIs. - **interface** [*interface-type*]–Enters the type of interface. - **line** *line-type*–Enters the line type.
8.	**template** *name* [... *name*]	Specifies the list of CNS connect templates in the CNS connect profile to be applied to the switch configuration. You can specify more than one template.
9.		Repeat Steps 7 to 8 to specify more interface parameters and CNS connect templates in the CNS connect profile.
10.	**exit**	Returns to global configuration mode.
11.	**hostname** *name*	Enters the hostname for the switch.
12.	**ip route** *network-number*	(Optional) Establishes a static route to Configuration Engine whose IP address is *network-number*.

	Command	Purpose
13.	**cns id** *interface num* {**dns-reverse** | **ipaddress** | **mac-address**} [**event**] [**image**] or **cns id** {**hardware-serial** | **hostname** | **string** *string* | **udi**} [**event**] [**image**]	(Optional) Sets the unique EventID or ConfigID used by the Configuration Engine. ■ *interface num*—Enters the type of interface for example, ethernet, group-async, loopback, or virtual-template. This setting specifies from which interface the IP or MAC address should be retrieved to define the unique ID. ■ **dns-reverse**—Retrieves the hostname and assigns it as the unique ID. ■ **ipaddress**—Uses the IP address. ■ **mac-address**—Uses the MAC address as the unique ID. ■ (Optional) **event**—Sets the ID to be the eventID value used to identify the switch. ■ (Optional) **image**—Sets the ID to be the imageID value used to identify the switch. **Note:** If the **event** and **image** keywords are omitted, the imageID value is used to identify the switch. ■ **hardware-serial**—Sets the switch serial number as the unique ID. ■ **hostname** (the default)—Selects the switch hostname as the unique ID, uses an arbitrary text string **string** *string* as the unique ID and **udi** sets the unique device identifier (UDI) as the unique ID.

	Command	Purpose
14.	**cns config initial** {*hostname* | *ip-address*} [*port-number*] [**event**] [**no-persist**] [**page** *page*] [**source** *ip-address*] [**syntax-check**]	Enables the Cisco IOS agent and initiates an initial configuration. ■ {*hostname* | *ip-address*}–Enters the hostname or the IP address of the configuration server. ■ (Optional) *port-number*–Enters the port number of the configuration server. The default port number is 80. ■ (Optional) **event**–Enables configuration success, failure, or warning messages when the configuration is finished. ■ (Optional) **no-persist**–Suppresses the automatic writing to NVRAM of the configuration pulled as a result of entering the **cns config initial** global configuration command. If the **no-persist** keyword is not entered, using the **cns config initial** command causes the resultant configuration to be automatically written to NVRAM. ■ (Optional) **page** *page*–Enters the web page of the initial configuration. The default is /Config/config/asp. ■ (Optional) **source** *ip-address*–Enters the source IP address. ■ (Optional) **syntax-check**–Checks the syntax when this parameter is entered. **Note:** Though visible in the command-line help string, the **encrypt**, **status** *url*, and **inventory** keywords are not supported.
15.	**end**	Returns to privileged EXEC mode.

Enabling a Partial Configuration

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**cns config partial** {*ip-address* | *hostname*} [*port-number*] [**source** *ip-address*]	Enables the configuration agent, and initiates a partial configuration. ■ {*ip-address* | *hostname*}–Enters the IP address or the hostname of the configuration server. ■ (Optional) *port-number*–Enters the port number of the configuration server. The default port number is 80. ■ (Optional) **source** *ip-address*–Enters the source IP address. **Note:** Though visible in the command-line help string, the **encrypt** keyword is not supported.
3.	**end**	Returns to privileged EXEC mode.

Monitoring and Maintaining Cisco IOS Configuration Engine

Command	Purpose
show cns config connections	Displays the status of the CNS Cisco IOS agent connections.
show cns config outstanding	Displays information about incremental (partial) CNS configurations that have started but are not yet completed.
show cns config stats	Displays statistics about the Cisco IOS agent.
show cns event connections	Displays the status of the CNS event agent connections.
show cns event stats	Displays statistics about the CNS event agent.
show cns event subject	Displays a list of event agent subjects that are subscribed to by applications.

Configuration Examples for Cisco IOS Configuration Engine

Enabling the CNS Event Agent: Example

This example shows how to enable the CNS event agent, set the IP address gateway to 10.180.1.27, set 120 seconds as the keepalive interval, and set 10 as the retry count.

```
Switch(config)# cns event 10.180.1.27 keepalive 120 10
```

Configuring an Initial CNS Configuration: Examples

This example shows how to configure an initial configuration on a remote switch when the switch configuration is unknown (the CNS Zero Touch feature).

```
Switch(config)# cns template connect template-dhcp
Switch(config-tmpl-conn)# cli ip address dhcp
Switch(config-tmpl-conn)# exit
Switch(config)# cns template connect ip-route
Switch(config-tmpl-conn)# cli ip route 0.0.0.0 0.0.0.0 ${next-hop}
Switch(config-tmpl-conn)# exit
Switch(config)# cns connect dhcp
Switch(config-cns-conn)# discover interface gigabitethernet
Switch(config-cns-conn)# template template-dhcp
Switch(config-cns-conn)# template ip-route
Switch(config-cns-conn)# exit
Switch(config)# hostname RemoteSwitch
RemoteSwitch(config)# cns config initial 10.1.1.1 no-persist
```

This example shows how to configure an initial configuration on a remote switch when the switch IP address is known. The Configuration Engine IP address is 172.28.129.22.

```
Switch(config)# cns template connect template-dhcp
Switch(config-tmpl-conn)# cli ip address dhcp
Switch(config-tmpl-conn)# exit
Switch(config)# cns template connect ip-route
Switch(config-tmpl-conn)# cli ip route 0.0.0.0 0.0.0.0 ${next-hop}
Switch(config-tmpl-conn)# exit
Switch(config)# cns connect dhcp
Switch(config-cns-conn)# discover interface gigabitethernet
Switch(config-cns-conn)# template template-dhcp
Switch(config-cns-conn)# template ip-route
Switch(config-cns-conn)# exit
Switch(config)# hostname RemoteSwitch
RemoteSwitch(config)# ip route 172.28.129.22 255.255.255.255 11.11.11.1
RemoteSwitch(config)# cns id ethernet 0 ipaddress
RemoteSwitch(config)# cns config initial 172.28.129.22 no-persist
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Network management commands	*Cisco IOS Network Management Command Reference*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	–

MIBs

MIBs	MIBs Link
–	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	–

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Configuring Switch Clusters

This chapter provides the concepts and procedures to create and manage switch clusters on your switch. You can create and manage switch clusters by using the command-line interface (CLI), Cisco Network Assistant (CNA) or SNMP. For information about CNA, see the online help for CNA.

This chapter provides information about switch clusters. It also includes guidelines and limitations for clusters mixed with other cluster-capable Catalyst switches, but it does not provide complete descriptions of the cluster features for switches in the cluster. For complete cluster information for a specific Catalyst platform, refer to the software configuration guide for that switch.

Cluster Command Switch Characteristics

A cluster command switch must meet these requirements:

- Has an IP address.
- Has Cisco Discovery Protocol (CDP) version 2 enabled (the default).
- Is not a command or cluster member switch of another cluster.
- Is connected to the standby cluster command switches through the management VLAN and to the cluster member switches through a common VLAN.

Standby Cluster Command Switch Characteristics

A standby cluster command switch must meet these requirements:

- Has an IP address.
- Has CDP version 2 enabled.
- Is connected to the command switch and to other standby command switches through its management VLAN.
- Is connected to all other cluster member switches (except the cluster command and standby command switches) through a common VLAN.
- Is redundantly connected to the cluster so that connectivity to cluster member switches is maintained.
- Is not a command or member switch of another cluster.

Candidate Switch and Cluster Member Switch Characteristics

Candidate switches are cluster-capable switches that have not yet been added to a cluster. Cluster member switches are switches that have actually been added to a switch cluster. Although not required, a candidate or cluster member switch can have its own IP address and password (for related considerations, see IP Addresses, page 99 and Passwords, page 100).

To join a cluster, a candidate switch must meet these requirements:

- Is running cluster-capable software.

- Has CDP version 2 enabled.
- Is not a command or cluster member switch of another cluster.
- If a cluster standby group exists, the switch is connected to every standby cluster command switch through at least one common VLAN. The VLAN to each standby cluster command switch can be different.
- Is connected to the cluster command switch through at least one common VLAN.

  Catalyst 1900, Catalyst 2820, Catalyst 2900 XL, Catalyst 2950, and Catalyst 3500 XL candidate and cluster member switches must be connected through their management VLAN to the cluster command switch and standby cluster command switches. For complete information about these switches in a switch-cluster environment, refer to the software configuration guide for that specific switch.

  This requirement does not apply if you have a Catalyst 2970, Catalyst 3550, Catalyst 3560, or Catalyst 3750 cluster command switch. Candidate and cluster member switches can connect through any VLAN in common with the cluster command switch.

Restrictions for Configuring Switch Clusters

We do not recommend using the **ip http access-class** global configuration command to limit access to specific hosts or networks. Access should be controlled through the cluster command switch or by applying access control lists (ACLs) on interfaces that are configured with IP address. For more information on ACLs, see Configuring Network Security with ACLs, page 575

Information About Configuring Switch Clusters

A *switch cluster* is a set of up to 16 connected, cluster-capable Catalyst switches that are managed as a single entity. The switches in the cluster use the switch clustering technology so that you can configure and troubleshoot a group of different Catalyst desktop switch platforms through a single IP address.

In a switch cluster, one switch must be the *cluster command switch* and up to 15 other switches can be *cluster member switches*. The total number of switches in a cluster cannot exceed 16 switches. The cluster command switch is the single point of access used to configure, manage, and monitor the cluster member switches. Cluster members can belong to only one cluster at a time.

Benefits of Clustering Switches

- Management of switches regardless of their interconnection media and their physical locations. The switches can be in the same location, or they can be distributed across a Layer 2 or Layer 3 (if your cluster is using a Catalyst 3550, Catalyst 3560, or Catalyst 3750 switch as a Layer 3 router between the Layer 2 switches in the cluster) network.

  Cluster members are connected to the cluster command switch according to the connectivity guidelines described in the Automatic Discovery of Cluster Candidates and Members, page 94. This section includes management VLAN considerations for the Catalyst 1900, Catalyst 2820, Catalyst 2900 XL, Catalyst 2950, and Catalyst 3500 XL switches. For complete information about these switches in a switch-cluster environment, refer to the software configuration guide for that specific switch.
- Command-switch redundancy if a cluster command switch fails. One or more switches can be designated as *standby cluster command switches* to avoid loss of contact with cluster members. A *cluster standby group* is a group of standby cluster command switches.
- Management of a variety of switches through a single IP address. This preserves IP addresses, especially if you have a limited number of them. All communication with the switch cluster is through the cluster command switch IP address.

Eligible Cluster Switches

Table 1 lists the switches eligible for switch clustering, including which ones can be cluster command switches and which ones can only be cluster member switches, and the required software versions.

Table 16 Eligible Switch Clusters

Switch	Cisco IOS Release	Cluster Capability
IE 2000	15.0(2)EA1 or later	Member or command switch
IE 3010	12.2(53)EZ or later	Member or command switch
IE 3000	12.2(40)EX or later	Member or command switch
IE 4000	15.2(2)EA or later	Member or command switch
IE 4010	15.2(4)EC or later	Member or command switch
IE 5000	15.2(2)EB or later	Member or command switch
Catalyst 3750-E or Catalyst 3560-E	12.2(35)SE2 or later	Member or command switch
Catalyst 3750	12.1(11)AX or later	Member or command switch
Catalyst 3560	12.1(19)EA1b or later	Member or command switch
Catalyst 3550	12.1(4)EA1 or later	Member or command switch
Catalyst 2975	12.2(46)EX or later	Member or command switch
Catalyst 2970	12.1(11)AX or later	Member or command switch
Catalyst 2960-S	12.2(53)SE or later	Member or command switch
Catalyst 2960	12.2(25)FX or later	Member or command switch
Catalyst 2955	12.1(12c)EA1 or later	Member or command switch
Catalyst 2950	12.0(5.2)WC(1) or later	Member or command switch
Catalyst 2950 LRE	12.1(11)JY or later	Member or command switch
Catalyst 2940	12.1(13)AY or later	Member or command switch
Catalyst 3500 XL	12.0(5.1)XU or later	Member or command switch
Catalyst 2900 XL (8-MB switches)	12.0(5.1)XU or later	Member or command switch
Catalyst 2900 XL (4-MB switches)	11.2(8.5)SA6 (recommended)	Member switch only
Catalyst 1900 and 2820	9.00(-A or -EN) or later	Member switch only

How to Plan for Switch Clustering

Anticipating conflicts and compatibility issues is a high priority when you manage several switches through a cluster. This section describes the guidelines, requirements, and caveats that you should understand before you create the cluster:

- Automatic Discovery of Cluster Candidates and Members, page 94
- IP Addresses, page 99
- Hostnames, page 99
- Passwords, page 100
- SNMP Community Strings, page 100
- TACACS+ and RADIUS, page 100

- LRE Profiles, page 100

Refer to the release notes for the list of Catalyst switches eligible for switch clustering, including which ones can be cluster command switches and which ones can only be cluster member switches, and for the required software versions and browser and Java plug-in configurations.

Automatic Discovery of Cluster Candidates and Members

The cluster command switch uses Cisco Discovery Protocol (CDP) to discover cluster member switches, candidate switches, neighboring switch clusters, and edge devices across multiple VLANs and in star or cascaded topologies.

Note: Do not disable CDP on the cluster command switch, on cluster members, or on any cluster-capable switches that you might want a cluster command switch to discover. For more information about CDP, see Configuring CDP, page 529

Following these connectivity guidelines ensures automatic discovery of the switch cluster, cluster candidates, connected switch clusters, and neighboring edge devices:

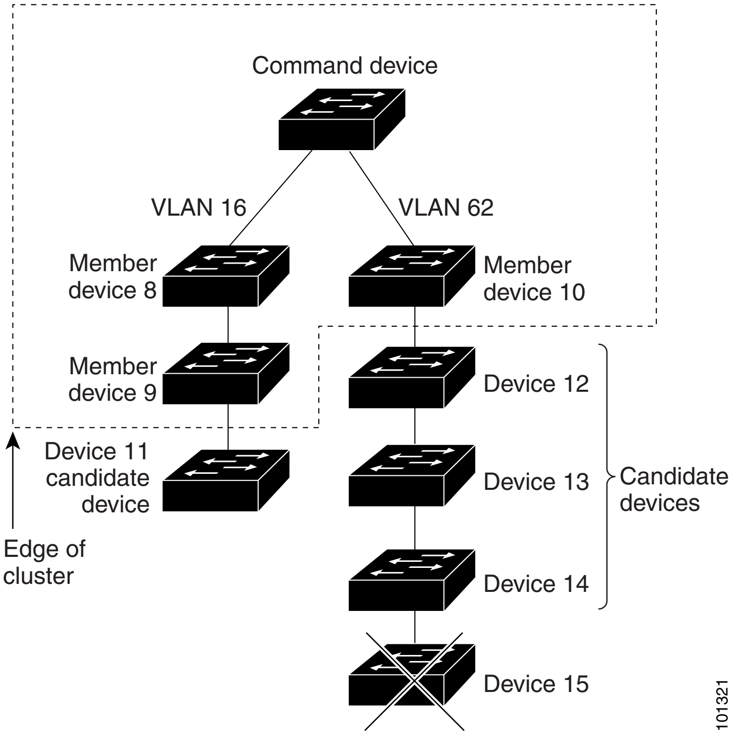
- Discovery Through CDP Hops, page 94
- Discovery Through Non-CDP-Capable and Noncluster-Capable Devices, page 95
- Discovery Through Different VLANs, page 96
- Discovery Through Different Management VLANs, page 97
- Discovery Through Routed Ports, page 97
- Discovery of Newly Installed Switches, page 98

Discovery Through CDP Hops

By using CDP, a cluster command switch can discover switches up to seven CDP hops away (the default is three hops) from the edge of the cluster. The edge of the cluster is where the last cluster member switches are connected to the cluster and to candidate switches. For example, cluster member switches 9 and 10 in Figure 7 on page 95 are at the edge of the cluster.

In Figure 7 on page 95, the cluster command switch has ports assigned to VLANs 16 and 62. The CDP hop count is three. The cluster command switch discovers switches 11, 12, 13, and 14 because they are within three hops from the edge of the cluster. It does not discover switch 15 because it is four hops from the edge of the cluster.

Figure 7 Discovery Through CDP Hops

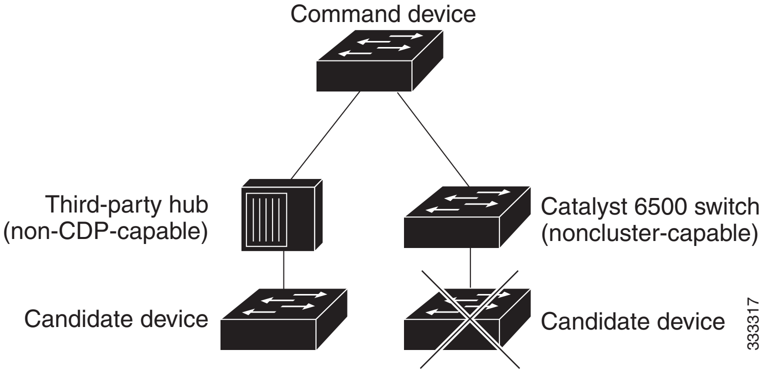


Discovery Through Non-CDP-Capable and Noncluster-Capable Devices

If a cluster command switch is connected to a *non-CDP-capable third-party hub* (such as a non-Cisco hub), it can discover cluster-enabled devices connected to that third-party hub. However, if the cluster command switch is connected to a *noncluster-capable Cisco device*, it cannot discover a cluster-enabled device connected beyond the noncluster-capable Cisco device.

Figure 8 on page 96 shows that the cluster command switch discovers the switch that is connected to a third-party hub. However, the cluster command switch does not discover the switch that is connected to a Catalyst 5000 switch.

Figure 8 Discovery Through Non-CDP-Capable and Noncluster-Capable Devices

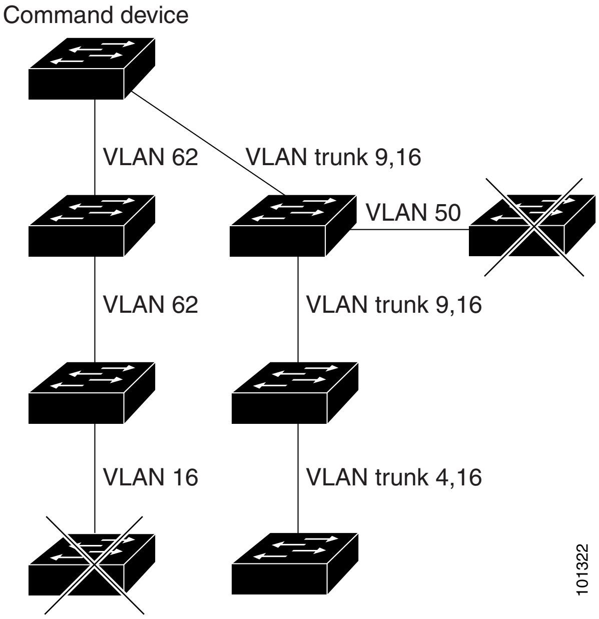


Discovery Through Different VLANs

If the cluster command switch is a Catalyst 2970, Catalyst 3550, Catalyst 3560, or Catalyst 3750 switch, the cluster can have cluster member switches in different VLANs. As cluster member switches, they must be connected through at least one VLAN in common with the cluster command switch. The cluster command switch in Figure 9 on page 97 has ports assigned to VLANs 9, 16, and 62 and therefore discovers the switches in those VLANs. It does not discover the switch in VLAN 50. It also does not discover the switch in VLAN 16 in the first column because the cluster command switch has no VLAN connectivity to it.

Catalyst 2900 XL, Catalyst 2950, and Catalyst 3500 XL cluster member switches must be connected to the cluster command switch through their management VLAN. For information about discovery through management VLANs, see Discovery Through Different Management VLANs, page 97.

Figure 9 Discovery Through Different VLANs



Discovery Through Different Management VLANs

Catalyst 2970, Catalyst 3550, Catalyst 3560, or Catalyst 3750 cluster command switches can discover and manage cluster member switches in different VLANs and different management VLANs. As cluster member switches, they must be connected through at least one VLAN in common with the cluster command switch. They do not need to be connected to the cluster command switch through their management VLAN. The default management VLAN is VLAN 1.

Note: If the switch cluster has a Catalyst 3750 or 2975 switch or has a switch stack, that switch or switch stack must be the cluster command switch.

The cluster command switch and standby command switch in Figure 11 on page 98 (assuming they are Catalyst 2960, Catalyst 2970, Catalyst 2975, Catalyst 3550, Catalyst 3560, or Catalyst 3750 cluster command switches) have ports assigned to VLANs 9, 16, and 62. The management VLAN on the cluster command switch is VLAN 9. Each cluster command switch discovers the switches in the different management VLANs except these:

- Switches 7 and 10 (switches in management VLAN 4) because they are not connected through a common VLAN (meaning VLANs 62 and 9) with the cluster command switch
- Switch 9 because automatic discovery does not extend beyond a noncandidate device, which is switch 7

Discovery Through Routed Ports

Note: The LAN Base image supports static routing.

If the cluster command switch has a routed port (RP) configured, it discovers only candidate and cluster member switches in the *same* VLAN as the routed port.

The Layer 3 cluster command switch in Figure 10 on page 98 can discover the switches in VLANs 9 and 62 but not the switch in VLAN 4. If the routed port path between the cluster command switch and cluster member switch 7 is lost, connectivity with cluster member switch 7 is maintained because of the redundant path through VLAN 9.

Figure 10 Discovery Through Routed Ports

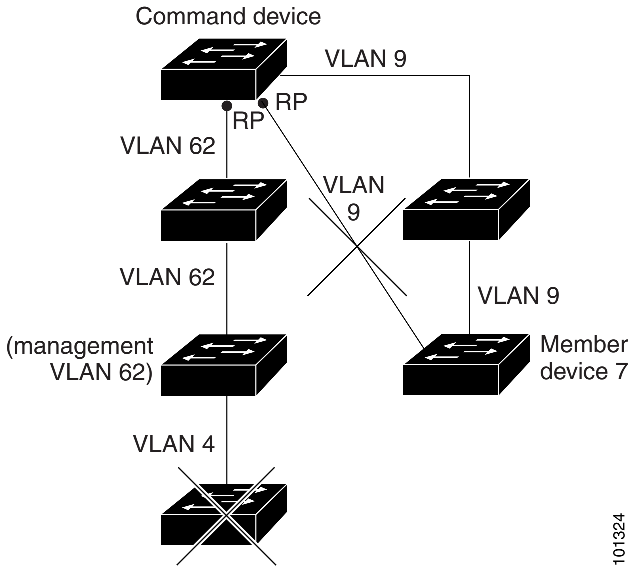
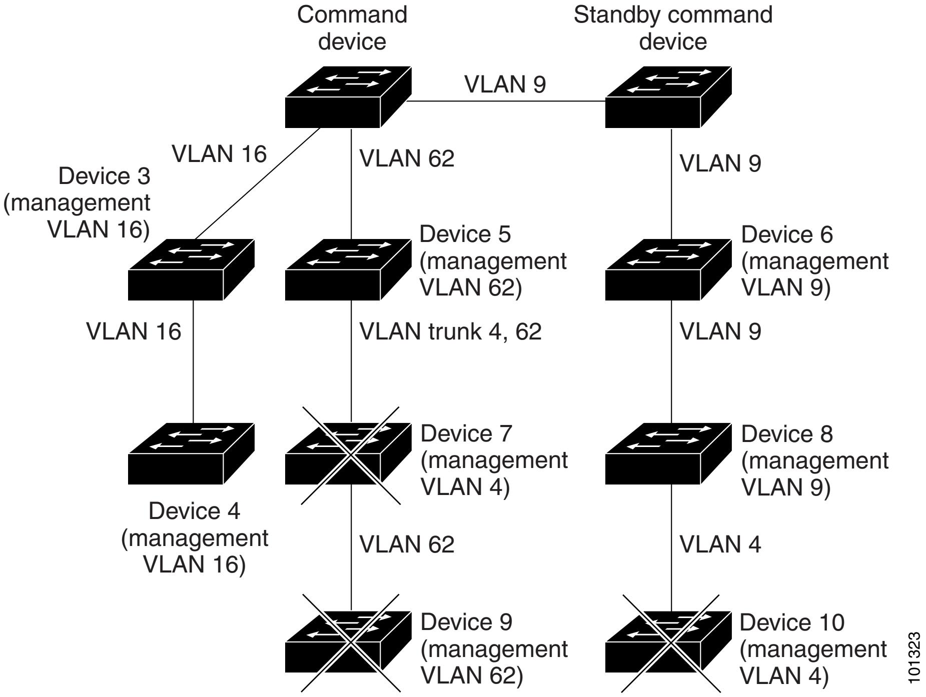


Figure 11 Discovery Through Different Management VLANs with a Layer 3 Cluster Command Switch



Discovery of Newly Installed Switches

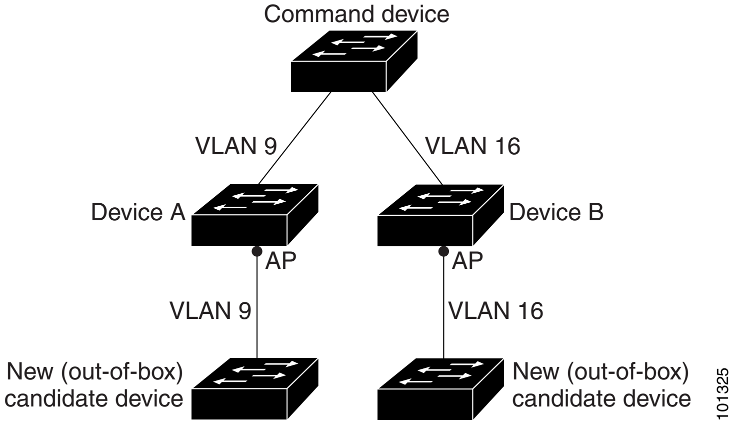
To join a cluster, the new, out-of-the-box switch must be connected to the cluster through one of its access ports. An access port (AP) carries the traffic of and belongs to only one VLAN. By default, the new switch and its access ports are assigned to VLAN 1.

When the new switch joins a cluster, its default VLAN changes to the VLAN of the immediately upstream neighbor. The new switch also configures its access port to belong to the VLAN of the immediately upstream neighbor.

The cluster command switch in Figure 12 on page 99 belongs to VLANs 9 and 16. When new cluster-capable switches join the cluster:

- One cluster-capable switch and its access port are assigned to VLAN 9.
- The other cluster-capable switch and its access port are assigned to management VLAN 16.

Figure 12 Discovery of Newly Installed Switches



IP Addresses

You must assign IP information to a cluster command switch. You can assign more than one IP address to the cluster command switch, and you can access the cluster through any of the command-switch IP addresses. If you configure a cluster standby group, you must use the standby-group virtual IP address to manage the cluster from the active cluster command switch. Using the virtual IP address ensures that you retain connectivity to the cluster if the active cluster command switch fails and that a standby cluster command switch becomes the active cluster command switch.

If the active cluster command switch fails and the standby cluster command switch takes over, you must either use the standby-group virtual IP address or any of the IP addresses available on the new active cluster command switch to access the cluster.

You can assign an IP address to a cluster-capable switch, but it is not necessary. A cluster member switch is managed and communicates with other cluster member switches through the command-switch IP address. If the cluster member switch leaves the cluster and it does not have its own IP address, you must assign an IP address to manage it as a standalone switch.

For more information about IP addresses, see Performing Switch Setup Configuration, page 59

Hostnames

You do not need to assign a hostname to either a cluster command switch or an eligible cluster member. However, a hostname assigned to the cluster command switch can help to identify the switch cluster. The default hostname for the switch is *Switch*.

If a switch joins a cluster and it does not have a hostname, the cluster command switch appends a unique member number to its own hostname and assigns it sequentially as each switch joins the cluster. The number means the order in which the switch was added to the cluster. For example, a cluster command switch named *eng-cluster* could name the fifth cluster member *eng-cluster-5*.

If a switch has a hostname, it retains that name when it joins a cluster and when it leaves the cluster.

If a switch received its hostname from the cluster command switch, was removed from a cluster, was then added to a new cluster, and kept the same member number (such as *5*), the switch overwrites the old hostname (such as *eng-cluster-5*) with the hostname of the cluster command switch in the new cluster (such as *mkg-cluster-5*). If the switch member number changes in the new cluster (such as *3*), the switch retains the previous name (*eng-cluster-5*).

Passwords

You do not need to assign passwords to an individual switch if it will be a cluster member. When a switch joins a cluster, it inherits the command-switch password and retains it when it leaves the cluster. If no command-switch password is configured, the cluster member switch inherits a null password. Cluster member switches only inherit the command-switch password.

If you change the member-switch password to be different from the command-switch password and save the change, the switch is not manageable by the cluster command switch until you change the member-switch password to match the command-switch password. Rebooting the member switch does not revert the password back to the command-switch password. We recommend that you do not change the member-switch password after it joins a cluster.

For more information about passwords, see Prevention for Unauthorized Switch Access, page 143.

For password considerations specific to the Catalyst 1900 and Catalyst 2820 switches, refer to the installation and configuration guides for those switches.

SNMP Community Strings

A cluster member switch inherits the command-switch first read-only (RO) and read-write (RW) community strings with *@esN* appended to the community strings:

- *command-switch-readonly-community-string@esN*, where *N* is the member-switch number.
- *command-switch-readwrite-community-string@esN*, where *N* is the member-switch number.

If the cluster command switch has multiple read-only or read-write community strings, only the first read-only and read-write strings are propagated to the cluster member switch.

The switches support an unlimited number of community strings and string lengths. For more information about SNMP and community strings, see Configuring SNMP, page 557

For SNMP considerations specific to the Catalyst 1900 and Catalyst 2820 switches, refer to the installation and configuration guides specific to those switches.

TACACS+ and RADIUS

If TACACS+ is configured on a cluster member, it must be configured on all cluster members. Similarly, if RADIUS is configured on a cluster member, it must be configured on all cluster members.The same switch cluster cannot have some members configured with TACACS+ and other members configured with RADIUS.

For more information about TACACS+, see Switch Access with TACACS+, page 145. For more information about RADIUS, see Configuring Radius Server Communication, page 172.

LRE Profiles

A configuration conflict occurs if a switch cluster has Long-Reach Ethernet (LRE) switches that use both private and public profiles. If one LRE switch in a cluster is assigned a public profile, all LRE switches in that cluster must have that same public profile. Before you add an LRE switch to a cluster, make sure that you assign it the same public profile used by other LRE switches in the cluster.

A cluster can have a mix of LRE switches that use different private profiles.

Managing Switch Clusters

Using the CLI to Manage Switch Clusters

You can configure cluster member switches from the CLI by first logging into the cluster command switch. Enter the **rcommand** user EXEC command and the cluster member switch number to start a Telnet session (through a console or Telnet connection) and to access the cluster member switch CLI. The command mode changes, and the Cisco IOS commands operate as usual. Enter the **exit** privileged EXEC command on the cluster member switch to return to the command-switch CLI.

This example shows how to log into member-switch 3 from the command-switch CLI:

```
switch# rcommand 3
```

If you do not know the member-switch number, enter the **show cluster members** privileged EXEC command on the cluster command switch.

The Telnet session accesses the member-switch CLI at the same privilege level as on the cluster command switch. The Cisco IOS commands then operate as usual.

Catalyst 1900 and Catalyst 2820 CLI Considerations

If your switch cluster has Catalyst 1900 and Catalyst 2820 switches running standard edition software, the Telnet session accesses the management console (a menu-driven interface) if the cluster command switch is at privilege level 15. If the cluster command switch is at privilege level 1 to 14, you are prompted for the password to access the menu console.

Command-switch privilege levels map to the Catalyst 1900 and Catalyst 2820 cluster member switches running standard and Enterprise Edition Software as follows:

- If the command-switch privilege level is 1 to 14, the cluster member switch is accessed at privilege level 1.
- If the command-switch privilege level is 15, the cluster member switch is accessed at privilege level 15.

Note: The Catalyst 1900 and Catalyst 2820 CLI is available only on switches running Enterprise Edition Software.

For more information about the Catalyst 1900 and Catalyst 2820 switches, refer to the installation and configuration guides for those switches.

Using SNMP to Manage Switch Clusters

When you first power on the switch, SNMP is enabled if you enter the IP information by using the setup program and accept its proposed configuration.

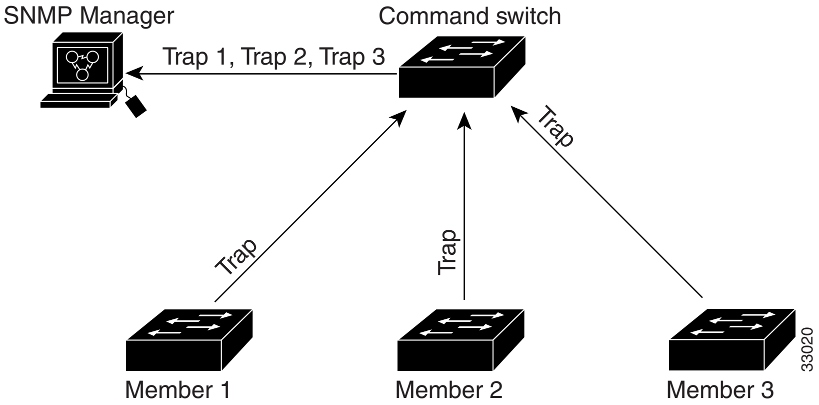
When you create a cluster, the cluster command switch manages the exchange of messages between cluster member switches and an SNMP application. The cluster software on the cluster command switch appends the cluster member switch number (*@esN*, where *N* is the switch number) to the first configured read-write and read-only community strings on the cluster command switch and propagates them to the cluster member switch. The cluster command switch uses this community string to control the forwarding of gets, sets, and get-next messages between the SNMP management station and the cluster member switches.

Note: When a cluster standby group is configured, the cluster command switch can change without your knowledge. Use the first read-write and read-only community strings to communicate with the cluster command switch if there is a cluster standby group configured for the cluster.

If the cluster member switch does not have an IP address, the cluster command switch redirects traps from the cluster member switch to the management station, as shown in Figure 13 on page 102. If a cluster member switch has its own IP address and community strings, the cluster member switch can send traps directly to the management station, without going through the cluster command switch.

If a cluster member switch has its own IP address and community strings, they can be used in addition to the access provided by the cluster command switch.

Figure 13 SNMP Management for a Cluster



Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Performing Switch Administration

This chapter describes how to perform one-time operations to administer your switch.

Information About Performing Switch Administration

System Time and Date Management

You can manage the system time and date on your switch using automatic configuration, such as the Network Time Protocol (NTP), or manual configuration methods.

System Clock

The basis of time service is the system clock. This clock runs from the moment the system starts up and keeps track of the date and time.

The system clock can then be set from these sources:

- NTP
- Manual configuration

The system clock can provide time to these services:

- User **show** commands
- Logging and debugging messages

The system clock keeps track of time internally based on Universal Time Coordinated (UTC), also known as Greenwich Mean Time (GMT). You can configure information about the local time zone and summer time (daylight saving time) so that the time appears correctly for the local time zone.

The system clock keeps track of whether the time is *authoritative* or not (that is, whether it has been set by a time source considered to be authoritative). If it is not authoritative, the time is available only for display purposes and is not redistributed. For configuration information, see Configuring Time and Date Manually, page 111.

Network Time Protocol

NTP is designed to time-synchronize a network of devices. NTP runs over User Datagram Protocol (UDP), which runs over IP. NTP is documented in RFC 1305.

An NTP network usually gets its time from an authoritative time source, such as a radio clock or an atomic clock attached to a time server. NTP then distributes this time across the network. NTP is extremely efficient; no more than one packet per minute is necessary to synchronize two devices to within a millisecond of one another.

NTP uses the concept of a *stratum* to describe how many NTP hops away a device is from an authoritative time source. A stratum 1 time server has a radio or atomic clock directly attached, a stratum 2 time server receives its time through NTP from a stratum 1 time server, and so on. A device running NTP automatically chooses as its time source the device with the lowest stratum number with which it communicates through NTP. This strategy effectively builds a self-organizing tree of NTP speakers.

NTP avoids synchronizing to a device whose time might not be accurate by never synchronizing to a device that is not synchronized. NTP also compares the time reported by several devices and does not synchronize to a device whose time is significantly different than the others, even if its stratum is lower.

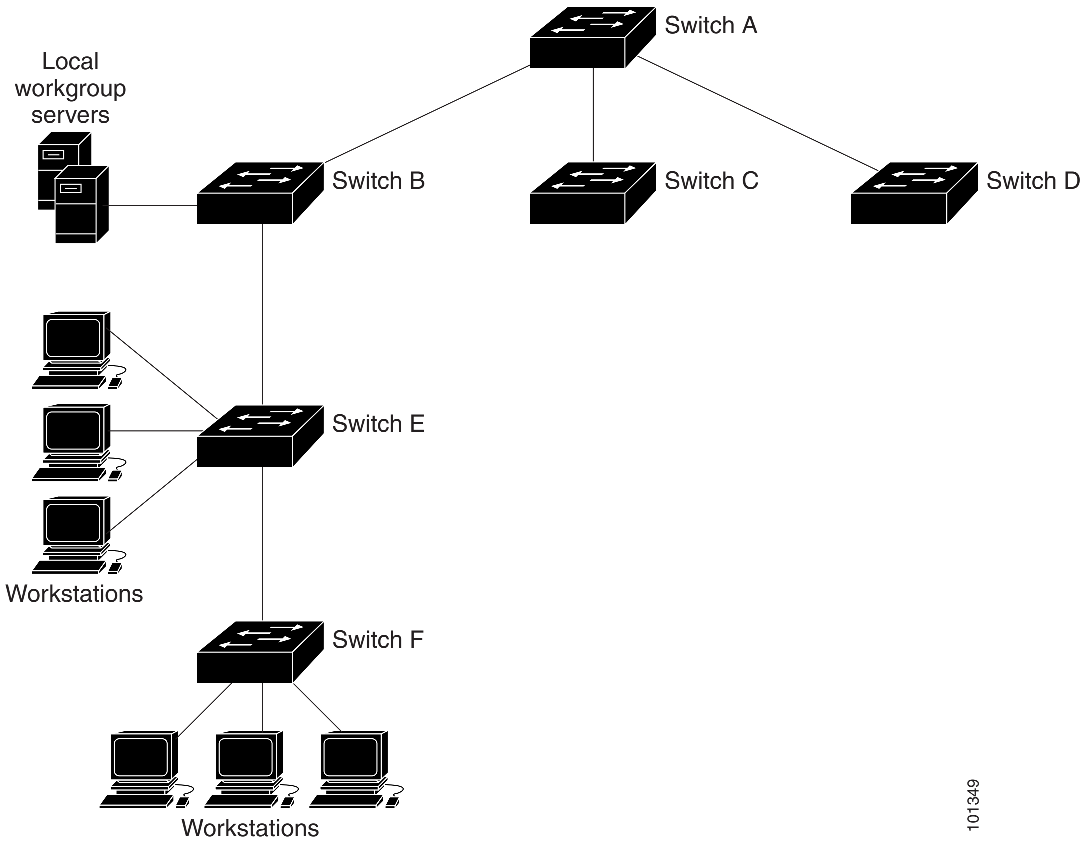
The communications between devices running NTP (known as *associations*) are usually statically configured; each device is given the IP address of all devices with which it should form associations. Accurate timekeeping is possible by exchanging NTP messages between each pair of devices with an association. However, in a LAN environment, NTP can be configured to use IP broadcast messages instead. This alternative reduces configuration complexity because each device can simply be configured to send or receive broadcast messages. However, in that case, information flow is one-way only.

The time kept on a device is a critical resource; you should use the security features of NTP to avoid the accidental or malicious setting of an incorrect time. Two mechanisms are available: an access list-based restriction scheme and an encrypted authentication mechanism.

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 the time service for your network be derived from the public NTP servers available on the IP Internet.

Figure 14 on page 106 shows a typical network example using NTP. Switch A is the NTP master, with Switches B, C, and D configured in NTP server mode, in server association with Switch A. Switch E is configured as an NTP peer to the upstream and downstream switches, Switch B and Switch F.

Figure 14 Typical NTP Network Configuration



If the network is isolated from the Internet, Cisco's implementation of NTP allows a device to act as if it is synchronized through NTP, when in fact it has learned the time by using other means. Other devices then synchronize to that device through NTP.

When multiple sources of time are available, NTP is always considered to be more authoritative. NTP time overrides the time set by any other method.

Several 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 host systems to be time-synchronized as well.

NTP Version 4

NTP version 4 is implemented on the switch. NTPv4 is an extension of NTP version 3. NTPv4 supports both IPv4 and IPv6 and is backward-compatible with NTPv3.

NTPv4 provides these capabilities:

- Support for IPv6.
- Improved security compared to NTPv3. The NTPv4 protocol provides a security framework based on public key cryptography and standard X509 certificates.
- Automatic calculation of the time-distribution hierarchy for a network. Using specific multicast groups, NTPv4 automatically configures the hierarchy of the servers to achieve the best time accuracy for the lowest bandwidth cost. This feature leverages site-local IPv6 multicast addresses.

For details about configuring NTPv4, see *Cisco IOS IPv6 Configuration Guide* on Cisco.com.

DNS

The DNS protocol controls the Domain Name System (DNS), a distributed database with which you can map hostnames to IP addresses. When you configure DNS on your switch, you can substitute the hostname for the IP address with all IP commands, such as **ping**, **telnet**, **connect**, and related Telnet support operations.

IP defines a hierarchical naming scheme that allows a device to be identified by its location or domain. Domain names are pieced together with periods (.) as the delimiting characters. For example, Cisco Systems is a commercial organization that IP identifies by a *com* domain name, so its domain name is *cisco.com*. A specific device in this domain, for example, the File Transfer Protocol (FTP) system is identified as *ftp.cisco.com*.

To keep track of domain names, IP has defined the concept of a domain name server, which holds a cache (or database) of names mapped to IP addresses. To map domain names to IP addresses, you must first identify the hostnames, specify the name server that is present on your network, and enable the DNS.

Default DNS Configuration

Feature	Default Setting
DNS enable state	Enabled.
DNS default domain name	None configured.
DNS servers	No name server addresses are configured.

Login Banners

You can configure a message-of-the-day (MOTD) and a login banner. The MOTD banner displays on all connected terminals at login and is useful for sending messages that affect all network users (such as impending system shutdowns).

The login banner also displays on all connected terminals. It appears after the MOTD banner and before the login prompts.

The MOTD and login banners are not configured.

System Name and Prompt

You configure the system name on the switch to identify it. By default, the system name and prompt are *Switch*.

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.

MAC Address Table

The MAC address table contains address information that the switch uses to forward traffic between ports. All MAC addresses in the address table are associated with one or more ports. The address table includes these types of addresses:

- Dynamic address—A source MAC address that the switch learns and then ages when it is not in use.
- Static address—A manually entered unicast address that does not age and that is not lost when the switch resets.

The address table lists the destination MAC address, the associated VLAN ID, and port number associated with the address and the type (static or dynamic).

Address Table

With multiple MAC addresses supported on all ports, you can connect any port on the switch to individual workstations, repeaters, switches, routers, or other network devices. The switch provides dynamic addressing by learning the source address of packets it receives on each port and adding the address and its associated port number to the address table. As stations are added or removed from the network, the switch updates the address table, adding new dynamic addresses and aging out those that are not in use.

The aging interval is globally configured. However, the switch maintains an address table for each VLAN, and STP can accelerate the aging interval on a per-VLAN basis.

The switch sends packets between any combination of ports, based on the destination address of the received packet. Using the MAC address table, the switch forwards the packet only to the port associated with the destination address. If the destination address is on the port that sent the packet, the packet is filtered and not forwarded. The switch always uses the store-and-forward method: complete packets are stored and checked for errors before transmission.

MAC Addresses and VLANs

All addresses are associated with a VLAN. An address can exist in more than one VLAN and have different destinations in each. Unicast addresses, for example, could be forwarded to port 1 in VLAN 1 and ports 9, 10, and 1 in VLAN 5.

Each VLAN maintains its own logical address table. A known address in one VLAN is unknown in another until it is learned or statically associated with a port in the other VLAN.

When private VLANs are configured, address learning depends on the type of MAC address:

- Dynamic MAC addresses learned in one VLAN of a private VLAN are replicated in the associated VLANs. For example, a MAC address learned in a private-VLAN secondary VLAN is replicated in the primary VLAN.
- Static MAC addresses configured in a primary or secondary VLAN are not replicated in the associated VLANs. When you configure a static MAC address in a private VLAN primary or secondary VLAN, you should also configure the same static MAC address in all associated VLANs.

Default MAC Address Table Configuration

Feature	Default Setting
Aging time	300 seconds
Dynamic addresses	Automatically learned
Static addresses	None configured

Address Aging Time for VLANs

Dynamic addresses are source MAC addresses that the switch learns and then ages when they are not in use. You can change the aging time setting for all VLANs or for a specified VLAN.

Setting too short an aging time can cause addresses to be prematurely removed from the table. Then when the switch receives a packet for an unknown destination, it floods the packet to all ports in the same VLAN as the receiving port. This unnecessary flooding can impact performance. Setting too long an aging time can cause the address table to be filled with unused addresses, which prevents new addresses from being learned. Flooding results, which can impact switch performance.

MAC Address Change Notification Traps

MAC address change notification tracks users on a network by storing the MAC address change activity. When the switch learns or removes a MAC address, an SNMP notification trap can be sent to the NMS. If you have many users coming and going from the network, you can set a trap-interval time to bundle the notification traps to reduce network traffic. The MAC notification history table stores MAC address activity for each port for which the trap is set. MAC address change notifications are generated for dynamic and secure MAC addresses. Notifications are not generated for self addresses, multicast addresses, or other static addresses.

Static Addresses

A static address has these characteristics:

- Is manually entered in the address table and must be manually removed.
- Can be a unicast or multicast address.
- Does not age and is retained when the switch restarts.

You can add and remove static addresses and define the forwarding behavior for them. The forwarding behavior defines how a port that receives a packet forwards it to another port for transmission. Because all ports are associated with at least one VLAN, the switch acquires the VLAN ID for the address from the ports that you specify. You can specify a different list of destination ports for each source port.

A packet with a static address that arrives on a VLAN where it has not been statically entered is flooded to all ports and not learned.

You add a static address to the address table by specifying the destination MAC unicast address and the VLAN from which it is received. Packets received with this destination address are forwarded to the interface specified with the *interface-id* option.

When you configure a static MAC address in a private-VLAN primary or secondary VLAN, you should also configure the same static MAC address in all associated VLANs. Static MAC addresses configured in a private-VLAN primary or secondary VLAN are not replicated in the associated VLAN.

Unicast MAC Address Filtering

When unicast MAC address filtering is enabled, the switch drops packets with specific source or destination MAC addresses. This feature is disabled by default and only supports unicast static addresses.

Follow these guidelines when using this feature:

- Multicast MAC addresses, broadcast MAC addresses, and router MAC addresses are not supported. If you specify one of these addresses when entering the **mac address-table static** *mac-addr* **vlan** *vlan-id* **drop** global configuration command, one of these messages appears:

  ```
  % Only unicast addresses can be configured to be dropped
  ```

  ```
  % CPU destined address cannot be configured as drop address
  ```

- Packets that are forwarded to the CPU are also not supported.
- If you add a unicast MAC address as a static address and configure unicast MAC address filtering, the switch either adds the MAC address as a static address or drops packets with that MAC address, depending on which command was entered last. The second command that you entered overrides the first command.

  For example, if you enter the **mac address-table static** *mac-addr* **vlan** *vlan-id* **interface** *interface-id* global configuration command followed by the **mac address-table static** *mac-addr* **vlan** *vlan-id* **drop** command, the switch drops packets with the specified MAC address as a source or destination.

  If you enter the **mac address-table static** *mac-addr* **vlan** *vlan-id* **drop** global configuration command followed by the **mac address-table static** *mac-addr* **vlan** *vlan-id* **interface** *interface-id* command, the switch adds the MAC address as a static address.

You enable unicast MAC address filtering and configure the switch to drop packets with a specific address by specifying the source or destination unicast MAC address and the VLAN from which it is received.

MAC Address Learning on a VLAN

By default, MAC address learning is enabled on all VLANs on the switch. You can control MAC address learning on a VLAN to manage the available MAC address table space by controlling which VLANs, and therefore which ports, can learn MAC addresses. Before you disable MAC address learning, be sure that you are familiar with the network topology and the switch system configuration. Disabling MAC address learning on a VLAN could cause flooding in the network.

Follow these guidelines when disabling MAC address learning on a VLAN:

- Use caution before disabling MAC address learning on a VLAN with a configured switch virtual interface (SVI). The switch then floods all IP packets in the Layer 2 domain.
- You can disable MAC address learning on a single VLAN ID (for example, **no mac address-table learning vlan 223**) or on a range of VLAN IDs (for example, **no mac address-table learning vlan 1-20, 15**).
- We recommend that you disable MAC address learning only in VLANs with two ports. If you disable MAC address learning on a VLAN with more than two ports, every packet entering the switch is flooded in that VLAN domain.
- You cannot disable MAC address learning on a VLAN that is used internally by the switch. If the VLAN ID that you enter is an internal VLAN, the switch generates an error message and rejects the command. To view internal VLANs in use, enter the **show vlan internal usage** privileged EXEC command.
- If you disable MAC address learning on a VLAN configured as a private-VLAN primary VLAN, MAC addresses are still learned on the secondary VLAN that belongs to the private VLAN and are then replicated on the primary VLAN. If you disable MAC address learning on the secondary VLAN, but not the primary VLAN of a private VLAN, MAC address learning occurs on the primary VLAN and is replicated on the secondary VLAN.
- You cannot disable MAC address learning on an RSPAN VLAN. The configuration is not allowed.

- If you disable MAC address learning on a VLAN that includes a secure port, MAC address learning is not disabled on that port. If you disable port security, the configured MAC address learning state is enabled.

To reenable MAC address learning on a VLAN, use the **default mac address-table learning vlan** *vlan-id* global configuration command. You can also reenable MAC address learning on a VLAN by entering the **mac address-table learning vlan** *vlan-id* global configuration command. The first (**default**) command returns to a default condition and therefore does not appear in the output from the **show running-config** command. The second command causes the configuration to appear in the **show running-config** privileged EXEC command display.

ARP Table Management

To communicate with a device (over Ethernet, for example), the software first must learn the 48-bit MAC address or the local data link address of that device. The process of learning the local data link address from an IP address is called *address resolution*.

The Address Resolution Protocol (ARP) associates a host IP address with the corresponding media or MAC addresses and the VLAN ID. Using an IP address, ARP finds the associated MAC address. When a MAC address is found, the IP-MAC address association is stored in an ARP cache for rapid retrieval. Then the IP datagram is encapsulated in a link-layer frame and sent over the network. Encapsulation of IP datagrams and ARP requests and replies on IEEE 802 networks other than Ethernet is specified by the Subnetwork Access Protocol (SNAP). By default, standard Ethernet-style ARP encapsulation (represented by the **arpa** keyword) is enabled on the IP interface.

ARP entries added manually to the table do not age and must be manually removed.

How to Perform Switch Administration

Configuring Time and Date Manually

If no other source of time is available, you can manually configure the time and date after the system is restarted. The time remains accurate until the next system restart. We recommend that you use manual configuration only as a last resort. If you have an outside source to which the switch can synchronize, you do not need to manually set the system clock.

Setting the System Clock

If you have an outside source on the network that provides time services, such as an NTP server, you do not need to manually set the system clock.

Beginning in privileged EXEC mode, follow these steps to set the system clock:

	Command	Purpose
1.	**clock set** *hh***:***mm***:***ss day month year* or **clock set** *hh***:***mm***:***ss month day year*	Manually sets the system clock using one of these formats: ■ *hh*:*mm*:*ss*–Specifies the time in hours (24-hour format), minutes, and seconds. The time specified is relative to the configured time zone. ■ *day*–Specifies the day by date in the month. ■ *month*–Specifies the month by name. ■ *year*–Specifies the year (no abbreviation).

Configuring the Time Zone

The *minutes-offset* variable in the **clock timezone** global configuration command is available for those cases where a local time zone is a percentage of an hour different from UTC. For example, the time zone for some sections of Atlantic Canada (AST) is UTC-3.5, where the 3 means 3 hours and .5 means 50 percent. In this case, the necessary command is **clock timezone AST -3 30**.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**clock timezone** *zone hours-offset* [*minutes-offset*]	Sets the time zone. The switch keeps internal time in universal time coordinated (UTC), so this command is used only for display purposes and when the time is manually set. ■ *zone*–Enters the name of the time zone to be displayed when standard time is in effect. The default is UTC. ■ *hours-offset*–Enters the hours offset from UTC. ■ (Optional) *minutes-offset*–Enters the minutes offset from UTC.
3.	**end**	Returns to privileged EXEC mode.

Configuring Summer Time (Daylight Saving Time)

To configure summer time (daylight saving time) in areas where it starts and ends on a particular day of the week each year, perform this task:

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**clock summer-time** *zone* **recurring** [*week day month hh***:***mm week day month hh***:***mm* [*offset*]]	Configures summer time to start and end on the specified days every year. Summer time is disabled by default. If you specify **clock summer-time** *zone* **recurring** without parameters, the summer time rules default to the United States rules. ■ *zone*–Specifies the name of the time zone (for example, PDT) to be displayed when summer time is in effect. ■ (Optional) *week*–Specifies the week of the month (1 to 5 or **last**). ■ (Optional) *day*–Specifies the day of the week (Sunday, Monday...). ■ (Optional) *month*–Specifies the month (January, February...). ■ (Optional) *hh***:***mm*–Specifies the time (24-hour format) in hours and minutes. ■ (Optional) *offset*–Specifies the number of minutes to add during summer time. The default is 60.
3.	**end**	Returns to privileged EXEC mode.

Configuring Summer Time (Exact Date and Time)

To configure summer time when it does not follow a recurring pattern (configure the exact date and time of the next summer time events), perform this task:

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**clock summer-time** *zone* **date** [*month date year hh*:*mm month date year hh*:*mm* [*offset*]] or **clock summer-time** *zone* **date** [*date month year hh*:*mm date month year hh*:*mm* [*offset*]]	Configures summer time to start on the first date and end on the second date. Summer time is disabled by default. ■ *zone*–Specifies the name of the time zone (for example, PDT) to be displayed when summer time is in effect. ■ (Optional) *week*–Specifies the week of the month (1 to 5 or **last**). ■ (Optional) *day*–Specifies the day of the week (Sunday, Monday...). ■ (Optional) *month*–Specifies the month (January, February...). ■ (Optional) *hh*:*mm*–Specifies the time (24-hour format) in hours and minutes. ■ (Optional) *offset*–Specifies the number of minutes to add during summer time. The default is 60.
3.	**end**	Returns to privileged EXEC mode.

Configuring a System Name

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**hostname** *name*	Manually configures a system name. The default setting is *switch*. The name must follow the rules for ARPANET hostnames. They must start with a letter, end with a letter or digit, and have as interior characters only letters, digits, and hyphens. Names can be up to 63 characters.
3.	**end**	Returns to privileged EXEC mode.

Setting Up DNS

If you use the switch IP address as its hostname, the IP address is used and no DNS query occurs. If you configure a hostname that contains no periods (.), a period followed by the default domain name is appended to the hostname before the DNS query is made to map the name to an IP address. The default domain name is the value set by the **ip domain-name** global configuration command. If there is a period (.) in the hostname, the Cisco IOS software looks up the IP address without appending any default domain name to the hostname.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**ip domain-name** *name*	Defines a default domain name that the software uses to complete unqualified hostnames (names without a dotted-decimal domain name). Do not include the initial period that separates an unqualified name from the domain name. At boot-up time, no domain name is configured; however, if the switch configuration comes from a BOOTP or Dynamic Host Configuration Protocol (DHCP) server, then the default domain name might be set by the BOOTP or DHCP server (if the servers were configured with this information).
3.	**ip name-server** *server-address1* [*server-address2 ... server-address6*]	Specifies the address of one or more name servers to use for name and address resolution. You can specify up to six name servers. Separate each server address with a space. The first server specified is the primary server. The switch sends DNS queries to the primary server first. If that query fails, the backup servers are queried.
4.	**ip domain-lookup**	(Optional) Enables DNS-based hostname-to-address translation on your switch. This feature is enabled by default. If your network devices require connectivity with devices in networks for which you do not control name assignment, you can dynamically assign device names that uniquely identify your devices by using the global Internet naming scheme (DNS).
5.	**end**	Returns to privileged EXEC mode.

Configuring Login Banners

Configuring a Message-of-the-Day Login Banner

You can create a single or multiline message banner that appears on the screen when someone logs in to the switch.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**banner motd** *c message c*	Specifies the message of the day. ■ *c*—Enters the delimiting character of your choice, for example, a pound sign (#), and press the **Return** key. The delimiting character signifies the beginning and end of the banner text. Characters after the ending delimiter are discarded. ■ *message*—Enters a banner message up to 255 characters. You cannot use the delimiting character in the message.
3.	**end**	Returns to privileged EXEC mode.

Configuring a Login Banner

You can configure a login banner to be displayed on all connected terminals. This banner appears after the MOTD banner and before the login prompt.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**banner login** *c message c*	Specifies the login message. ■ *c*–Enters the delimiting character of your choice, for example, a pound sign (#), and press the **Return** key. The delimiting character signifies the beginning and end of the banner text. Characters after the ending delimiter are discarded. ■ *message*–Enters a login message up to 255 characters. You cannot use the delimiting character in the message.
3.	**end**	Returns to privileged EXEC mode.

Managing the MAC Address Table

Changing the Address Aging Time

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**mac address-table aging-time** [**0** | *10-1000000*] [**vlan** *vlan-id*]	Sets the length of time that a dynamic entry remains in the MAC address table after the entry is used or updated. The range is 10 to 1000000 seconds. The default is 300. You can also enter 0, which disables aging. Static address entries are never aged or removed from the table. ■ *vlan-id*–Valid IDs are 1 to 4096.
3.	**end**	Returns to privileged EXEC mode.

Configuring MAC Address Change Notification Traps

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**snmp-server host** *host-addr* {**traps** | **informs**} {**version** {**1** | **2c** | **3**}} *community-string notification-type*	Specifies the recipient of the trap message. ■ *host-addr*–Specifies the name or address of the NMS. ■ **traps** (the default)–Sends SNMP traps to the host. ■ **informs**–Sends SNMP informs to the host. ■ Specifies the SNMP version to support. Version 1, the default, is not available with informs. ■ *community-string*–Specifies the string to send with the notification operation. You can set this string by using the **snmp-server host** command, but we recommend that you define this string by using the **snmp-server community** command before using the **snmp-server host** command. ■ *notification-type*–Uses the **mac-notification** keyword.
3.	**snmp-server enable traps mac-notification change**	Enables the switch to send MAC address change notification traps to the NMS.
4.	**mac address-table notification change**	Enables the MAC address change notification feature.
5.	**mac address-table notification change** [**interval** *value*] [**history-size** *value*]	Enters the trap interval time and the history table size. ■ (Optional) **interval** *value*–Specifies the notification trap interval in seconds between each set of traps that are generated to the NMS. The range is 0 to 2147483647 seconds; the default is 1 second. ■ (Optional) **history-size** *value*–Specifies the maximum number of entries in the MAC notification history table. The range is 0 to 500; the default is 1.
6.	**interface** *interface-id*	Enters interface configuration mode, and specifies the Layer 2 interface on which to enable the SNMP MAC address notification trap.
7.	**snmp trap mac-notification change** {**added** | **removed**}	Enables the MAC address change notification trap on the interface. ■ Enables the trap when a MAC address is **added** on this interface. ■ Enables the trap when a MAC address is **removed** from this interface.
8.	**end**	Returns to privileged EXEC mode.

Configuring MAC Address Move Notification Traps

When you configure MAC-move notification, an SNMP notification is generated and sent to the network management system whenever a MAC address moves from one port to another within the same VLAN.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**snmp-server host** *host-addr* {**traps** | **informs**} {**version** {**1** | **2c** | **3**}} *community-string notification-type*	Specifies the recipient of the trap message. ■ *host-addr*–Specifies the name or address of the NMS. ■ **traps** (the default)–Sends SNMP traps to the host. ■ **informs**–Sends SNMP informs to the host. ■ **version**–Specifies the SNMP version to support. Version 1, the default, is not available with informs. ■ *community-string*–Specifies the string to send with the notification operation. You can set this string by using the **snmp-server host** command, but we recommend that you define this string by using the **snmp-server community** command before using the **snmp-server host** command. ■ *notification-type*–Uses the **mac-notification** keyword.
3.	**snmp-server enable traps mac-notification move**	Enables the switch to send MAC address move notification traps to the NMS.
4.	**mac address-table notification mac-move**	Enables the MAC address move notification feature.
5.	**end**	Returns to privileged EXEC mode.

Configuring MAC Threshold Notification Traps

When you configure MAC threshold notification, an SNMP notification is generated and sent to the network management system when a MAC address table threshold limit is reached or exceeded.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**snmp-server host** *host-addr* {**traps** | **informs**} {**version** {**1** | **2c** | **3**}} *community-string notification-type*	Specifies the recipient of the trap message. ■ *host-addr*–Specifies the name or address of the NMS. ■ **traps** (the default)–Sends SNMP traps to the host. ■ **informs**–Sends SNMP informs to the host. ■ **version**–Specifies the SNMP version to support. Version 1, the default, is not available with informs. ■ *community-string*–Specifies the string to send with the notification operation. You can set this string by using the **snmp-server host** command, but we recommend that you define this string by using the **snmp-server community** command before using the **snmp-server host** command. ■ *notification-type*–Uses the **mac-notification** keyword.
3.	**snmp-server enable traps mac-notification threshold**	Enables the switch to send MAC threshold notification traps to the NMS.
4.	**mac address-table notification threshold**	Enables the MAC address threshold notification feature.
5.	**mac address-table notification threshold** [**limit** *percentage*] | [**interval** *time*]	Enters the threshold value for the MAC address threshold usage monitoring. ■ (Optional) **limit** *percentage*–Specifies the percentage of the MAC address table use; valid values are from 1 to 100 percent. The default is 50 percent. ■ (Optional) **interval** *time*–Specifies the time between notifications; valid values are greater than or equal to 120 seconds. The default is 120 seconds.
6.	**end**	Returns to privileged EXEC mode.

Adding and Removing Static Address Entries

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**mac address-table static** *mac-addr* **vlan** *vlan-id* **interface** *interface-id*	Adds a static address to the MAC address table. ■ *mac-addr*–Specifies the destination MAC unicast address to add to the address table. Packets with this destination address received in the specified VLAN are forwarded to the specified interface. ■ *vlan-id*–Specifies the VLAN for which the packet with the specified MAC address is received. Valid VLAN IDs are 1 to 4096. ■ *interface-id*–Specifies the interface to which the received packet is forwarded. Valid interfaces include physical ports or port channels. For static multicast addresses, you can enter multiple interface IDs. For static unicast addresses, you can enter only one interface at a time, but you can enter the command multiple times with the same MAC address and VLAN ID.
3.	**end**	Returns to privileged EXEC mode.

Configuring Unicast MAC Address Filtering

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**mac address-table static** *mac-addr* **vlan** *vlan-id* **drop**	Enables unicast MAC address filtering and configures the switch to drop a packet with the specified source or destination unicast static address. ■ *mac-addr*–Specifies a source or destination unicast MAC address. Packets with this MAC address are dropped. ■ *vlan-id*–Specifies the VLAN for which the packet with the specified MAC address is received. Valid VLAN IDs are 1 to 4096.
3.	**end**	Returns to privileged EXEC mode.

Disabling MAC Address Learning on a VLAN

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**no mac address-table learning vlan** *vlan-id*	Disables MAC address learning on the specified VLAN or VLANs. You can specify a single VLAN ID or a range of VLAN IDs separated by a hyphen or comma. Valid VLAN IDs are 1 to 4096.
3.	**end**	Returns to privileged EXEC mode.

Monitoring and Maintaining Switch Administration

Command	Purpose
clear mac address-table dynamic	Removes all dynamic entries.
clear mac address-table dynamic address *mac-address*	Removes a specific MAC address.
clear mac address-table dynamic interface *interface-id*	Removes all addresses on the specified physical port or port channel.
clear mac address-table dynamic vlan *vlan-id*	Removes all addresses on a specified VLAN.
show clock [**detail**]	Displays the time and date configuration.
show ip igmp snooping groups	Displays the Layer 2 multicast entries for all VLANs or the specified VLAN.
show mac address-table address	Displays MAC address table information for the specified MAC address.
show mac address-table aging-time	Displays the aging time in all VLANs or the specified VLAN.
show mac address-table count	Displays the number of addresses present in all VLANs or the specified VLAN.
show mac address-table dynamic	Displays only dynamic MAC address table entries.
show mac address-table interface	Displays the MAC address table information for the specified interface.
show mac address-table learning	Displays MAC address learning status of all VLANs or the specified VLAN.
show mac address-table notification	Displays the MAC notification parameters and history table.
show mac address-table static	Displays only static MAC address table entries.
show mac address-table vlan	Displays the MAC address table information for the specified VLAN.

Configuration Examples for Performing Switch Admininistration

Setting the System Clock: Example

This example shows how to manually set the system clock to 1:32 p.m. on July 23, 2001:

```
Switch# clock set 13:32:00 23 July 2001
```

Configuring Summer Time: Examples

The first part of the **clock summer-time** global configuration command specifies when summer time begins, and the second part specifies when it ends. All times are relative to the local time zone. The start time is relative to standard time. The end time is relative to summer time. If the starting month is after the ending month, the system assumes that you are in the southern hemisphere.

This example (for daylight savings time) shows how to specify that summer time starts on the first Sunday in April at 02:00 and ends on the last Sunday in October at 02:00:

```
Switch(config)# clock summer-time PDT recurring 1 Sunday April 2:00 last Sunday October 2:00
```

This example shows how to set summer time to start on October 12, 2000, at 02:00, and end on April 26, 2001, at 02:00:

```
Switch(config)# clock summer-time pdt date 12 October 2000 2:00 26 April 2001 2:00
```

Configuring a MOTD Banner: Examples

This example shows how to configure a MOTD banner for the switch by using the pound sign (#) symbol as the beginning and ending delimiter:

```
Switch(config)# banner motd #
This is a secure site. Only authorized users are allowed.
For access, contact technical support.
#
Switch(config)#
```

This example shows the banner that appears from the previous configuration:

```
Unix> telnet 172.2.5.4
Trying 172.2.5.4...
Connected to 172.2.5.4.
Escape character is '^]'.

This is a secure site. Only authorized users are allowed.
For access, contact technical support.

User Access Verification

Password:
```

Configuring a Login Banner: Example

This example shows how to configure a login banner for the switch by using the dollar sign ($) symbol as the beginning and ending delimiter:

```
Switch(config)# banner login $
Access for authorized users only. Please enter your username and password.
$
Switch(config)#
```

Configuring MAC Address Change Notification Traps: Example

This example shows how to specify 172.20.10.10 as the NMS, enable the switch to send MAC address notification traps to the NMS, enable the MAC address-change notification feature, set the interval time to 123 seconds, set the history-size to 100 entries, and enable traps whenever a MAC address is added on the specified port.

```
Switch(config)# snmp-server host 172.20.10.10 traps private mac-notification
Switch(config)# snmp-server enable traps mac-notification change
Switch(config)# mac address-table notification change
Switch(config)# mac address-table notification change interval 123
Switch(config)# mac address-table notification change history-size 100
Switch(config)# interface GigabitEthernet1/18

Switch(config-if)# snmp trap mac-notification change added
```

Sending MAC Address Move Notification Traps: Example

This example shows how to specify 172.20.10.10 as the NMS, enable the switch to send MAC address move notification traps to the NMS, enable the MAC address move notification feature, and enable traps when a MAC address moves from one port to another.

```
Switch(config)# snmp-server host 172.20.10.10 traps private mac-notification
Switch(config)# snmp-server enable traps mac-notification move
Switch(config)# mac address-table notification mac-move
```

Configuring MAC Threshold Notification Traps: Example

This example shows how to specify 172.20.10.10 as the NMS, enable the MAC address threshold notification feature, set the interval time to 123 seconds, and set the limit to 78 per cent.

```
Switch(config)# snmp-server host 172.20.10.10 traps private mac-notification
Switch(config)# snmp-server enable traps mac-notification threshold
Switch(config)# mac address-table notification threshold
Switch(config)# mac address-table notification threshold interval 123
Switch(config)# mac address-table notification threshold limit 78
```

Adding the Static Address to the MAC Address Table: Example

This example shows how to add the static address c2f3.220a.12f4 to the MAC address table. When a packet is received in VLAN 4 with this MAC address as its destination address, the packet is forwarded to the specified port:

```
Switch(config)# mac address-table static c2f3.220a.12f4 vlan 4 interface GigabitEthernet1/17
```

Configuring Unicast MAC Address Filtering: Example

This example shows how to enable unicast MAC address filtering and to configure the switch to drop packets that have a source or destination address of c2f3.220a.12f4. When a packet is received in VLAN 4 with this MAC address as its source or destination, the packet is dropped:

```
Switch(config)# mac address-table static c2f3.220a.12f4 vlan 4 drop
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*
Cisco IOS routing commands.	*Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	–

MIBs

MIBs	MIBs Link
–	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	–

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Configuring PTP

Precision Time Protocol (PTP) is defined in IEEE 1588 as Precision Clock Synchronization for Networked Measurements and Control Systems, and was developed to synchronize the clocks in packet-based networks that include distributed device clocks of varying precision and stability. PTP is designed specifically for industrial, networked measurement and control systems, and is optimal for use in distributed systems because it requires minimal bandwidth and little processing overhead.

For information about configuring PTP on Cisco Industrial Ethernet switches, see Precision Time Protocol Software Configuration Guide for IE 4000, IE 4010 and IE 5000 Switches.

Configuring PROFINET

Restrictions for Configuring PROFINET

Cisco IE series switches support PROFINET I/O, RT but not IRT (isochronous real-time).

Information About Configuring PROFINET

PROFINET is the PROFIBUS International (PI) open Industrial Ethernet Standard that uses TCP/IP and IT standards for automation control. PROFINET is particularly useful for industrial automation systems and process control networks, in which motion control and precision control of instrumentation and test equipment are important. It emphasizes data exchange and defines communication paths to meet speed requirements. PROFINET communication is scalable on three levels:

- Normal non-real-time communication uses TCP/IP and enables bus cycle times of approximately 100 ms.
- Real-time communication enables cycle times of approximately 10 ms.
- Isochronous real-time communication enables cycle times of approximately 1 ms.

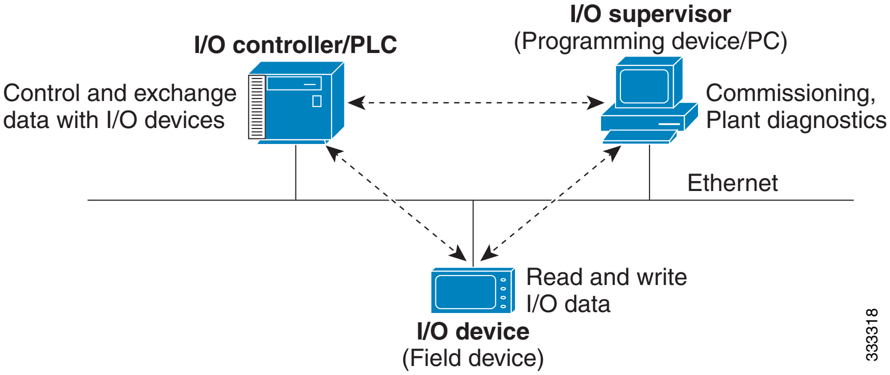
PROFINET I/O is a modular communication framework for distributed automation applications. PROFINET I/O uses cyclic data transfer to exchange data, alarms, and diagnostic information with programmable controllers, input/output (I/O) devices, and other automation controllers (for example, motion controllers).

PROFINET I/O recognizes three classes of devices:

- I/O devices
- I/O controllers
- I/O supervisors

PROFINET Device Roles

Figure 15 PROFINET Device Roles



An I/O controller is a programmable logic controller (PLC) that controls I/O devices and exchanges data such as configuration, alarms, and I/O data through an automation program. The I/O controller and the I/O supervisor exchange diagnostic information. The I/O controller shares configuration and input/output information with the I/O device and receives alarms from the I/O device.

PROFINET is designed to be the sole or primary management system platform. Because the I/O controller detects the switch with the Discovery and Configuration Protocol (DCP), and sets the device name and IP address, you do not need to enter Cisco IOS commands for the basic configuration. For advanced configurations (for example, QoS, DHCP, and similar features) you must use Cisco IOS commands on the switch because these features cannot be configured by using PROFINET.

An I/O supervisor is an engineering station, such as a human machine interface (HMI) or PC, used for commissioning, monitoring, and diagnostic analysis. The I/O supervisor exchanges diagnostic, status, control, and parameter information with the I/O device.

An I/O device is a distributed input/output device such as a sensor, an actuator, or a motion controller.

Note: If Profinet DCP cannot detect the switch/PLC/IO mac addresses, temporarily disable the firewall/virus scan from the Window PC that installed the Siemens STEP7 or TIA Portal.

In a PROFINET I/O system, all the I/O devices communicate over an Ethernet communication network to meet the automation industry requirement for bus cycle times of less than 100 ms. The network uses switches and full-duplex data exchange to avoid data collisions.

PROFINET Device Data Exchange

After PROFINET uses DCP to discover devices, including the switch, they establish application relationships (ARs) and communication relationships (CRs). After a connection is established and information about device parameters is exchanged, input and output data is exchanged. The switch uses non-real-time CRs to exchange the data attributes listed in Table 17 on page 129 and Table 18 on page 129.

Table 17 PROFINET I/O Switch Attributes

PROFINET I/O Switch Configuration Attributes	Value or Action
Device name	Configures a name for the device.
TCP/IP	IP address, subnet mask, default gateway, SVI.
Primary temperature alarm	Enables or disables monitoring for the specified alarm.
Secondary temperature alarm	Enables or disables monitoring for the specified alarm.
RPS failed alarm	Enables or disables monitoring for the specified alarm.
Relay major alarm	Enables or disables monitoring for the specified alarm.
Reset to factory defaults	Uses the PROFINET I/O controller to reset the switch to factory defaults. This action removes the startup configuration and reloads the switch.
Relay major configuration	Specifies the type of port alarm (for example, link fault) that triggers the major relay. Any port configured with the specified alarm type can trigger the major relay.

Table 18 PROFINET I/O Port Attributes

PROFINET I/O Port Configuration Attributes	Value or Action
Speed	10/100/1000/auto,
Duplex	Half/full/auto,
Port mode	Access/trunk,
Link status	Shut down/no shut down,
Configure rate limiting	Broadcast, unicast, multicast threshold exceeds configured levels.
Port link fault alarm	Enables or disables monitoring for specified alarm.
Port not forwarding alarm	Enables or disables monitoring for specified alarm.
Port not operating alarm	Enables or disables monitoring for specified alarm.
Port FCS threshold alarm	Enables or disables monitoring for specified alarm.

PROFINET devices are integrated by using a general station description (GSD) file that contains the data for engineering and data exchange between the I/O controller, the I/O supervisor, and the I/O devices, including the switch. Each PROFINET I/O field device must have an associated GSD file that describes the properties of the device and contains all this information required for configuration:

- Device identification information (device ID, vendor ID and name, product family, number of ports)
- Number and types of pluggable modules
- Error text for diagnostic information
- Communication parameters for I/O devices, including the minimum cycle time, the reduction ratio, and the watch dog time
- Configuration data for the I/O device modules, including speed, duplex, VLAN, port security information, alarms, and broadcast-rate-limiting thresholds
- Parameters configured for I/O device modules for the attributes listed in Table 18 on page 129

The GSD file is on the switch, but the I/O supervisor uses this file.

Note: You must use the GSD file that is associated with the Cisco IOS release on the switch to manage your PROFINET network. Both the I/O supervisor and the Cisco IOS software alert you to a mismatch between the GSD file and the switch Cisco IOS software version.

How to Configure PROFINET

Configuring PROFINET

You can use either the PROFINET software on the I/O supervisor or the Cisco IOS software for basic switch configuration.

After you enable PROFINET, LLDP is automatically enabled on the switch because PROFINET relies on LLDP to fully function. If you disable PROFINET, you can enable or disable LLDP as needed.

Default Configuration

PROFINET is enabled by default on all the base switch module ports. The default config is enabled on VLAN 1 but can be changed to another VLAN ID. If PROFINET has been disabled, follow the instructions in the Enabling PROFINET, page 130.

Enabling PROFINET

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**profinet**	Enables PROFINET on the switch.
3.	**profinet id** *line*	(Optional) Sets the PROFINET device identifier (ID) by using the Cisco IOS software. The maximum length is 240 characters. The only special characters allowed are the period (.) and hyphen (-), and they are allowed only in specific positions within the ID string. It can have multiple labels within the string. Each label can be from 1 to 63 characters, and labels must be separated by a period (.). The final character in the string must not be zero (0). For more details about configuring the PROFINET ID, see the PROFINET specification, document number TC2-06-0007a, filename PN-AL-protocol_2722_V22_Oct07, available from PROFIBUS.
4.	**profinet vlan** *vlan id*	(Optional) Changes the VLAN number. The default VLAN number is 1. The VLAN ID range is 1-4096. Supports one VLAN per switch.
5.	**interface** *<x/y>*	Access interface configuration mode.
6.	**switchport mode access**	Configure the port mode as access.
7.	**switchport access vlan** <>	Configure the access vlan required for the port.
8.	**switchport voice vlan dot1p**	Add the voice vlan dot1p command. ■ Required only for the IE4010 switch platform.
9.	**end**	Returns to privileged EXEC mode.
10.	**show running-config**	Verifies your entries.
11.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Guidelines for the IE4010

The IE4010 does not behave same as other IE platforms when it comes to Vlan 0 tags. You need to add the **voice vlan dot1p** command to the interface to allow the vlan 0 tagged packets.

The following example shows the configuration:

```
Interface gi1/2
description PLC
switchport access vlan 10
switchport mode access
switchport voice vlan dot1p
```

The highlighted command above allows the vlan 0 tagged packets to be accepted on vlan 10 along with the COS values of ingress frames.

Monitoring and Maintaining PROFINET

Table 19 Commands for Displaying the PROFINET Configuration

Command	Purpose
show profinet sessions	Displays the currently connected PROFINET sessions.
show profinet status	Displays the status of the PROFINET subsystem.
show lldp neighbor interface x/x detail	Displays information about the adjacent interface.

```
Example
IE5000#show profinet status
State : Enabled
Vlan : 2
Id : Ie5000
Connected : Yes
ReductRatio : 128
GSD Version: Match
```

Troubleshooting PROFINET

The PLC has LEDs that display red for alarms, and the I/O supervisor software monitors those alarms.

To troubleshoot PROFINET use the **debug profinet** privileged EXEC command with the keywords shown in Table 20 on page 131. Be aware that the output of a **debug** command might cause a serial link to fail. You should use these commands only under the guidance of a Cisco Technical Support engineer. When you use this command, use Telnet to access the Cisco IOS command-line interface (CLI) by using Ethernet rather than a serial port.

Table 20 Commands for Troubleshooting the PROFINET Configuration

Command	Purpose
debug profinet alarm	Displays the alarm status (on or off) and content of PROFINET alarms.
debug profinet cyclic	Displays information about the time-cycle-based PROFINET Ethernet frames.
debug profinet error	Displays the PROFINET session errors.
debug profinet packet ethernet	Displays information about the PROFINET Ethernet packets.

Table 20 Commands for Troubleshooting the PROFINET Configuration (continued)

Command	Purpose
debug profinet packet udp	Displays information about the PROFINET Upper Layer Data Protocol (UDP) packets.
debug profinet platform	Displays information about the interaction between the Cisco IOS software and PROFINET.
debug profinet topology	Displays the PROFINET topology packets received.
debug profinet trace	Displays a group of traced debug output logs.

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Configuring CIP

Restrictions for Configuring CIP

CIP can be enabled on only one VLAN on the switch.

Information About Configuring CIP

The Common Industrial Protocol (CIP) is an industrial protocol for industrial automation applications. It is supported by Open DeviceNet Vendors Association (ODVA), an organization that supports network technologies based upon CIP such as DeviceNet, EtherNet/IP, CIP Safety and CIP Sync.

Previously known as Control and Information Protocol, CIP encompasses a comprehensive suite of messages and services for the collection of manufacturing automation applications - control, safety, synchronization, motion, configuration and information. CIP allows users to integrate these manufacturing applications with enterprise-level Ethernet networks and the Internet.

How to Configure CIP

Default Configuration

By default, CIP is not enabled.

Enabling CIP

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**cip security** {**password** *password* | **window timeout** *value*}	Sets CIP security options on the switch.
3.	**interface vlan 20**	Enters interface configuration mode.
4.	**cip enable**	Enables CIP on a VLAN.
5.	**end**	Returns to privileged EXEC mode.
6.	**show running-config**	Verifies your entries.
7.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Monitoring CIP

Table 21 Commands for Displaying the CIP Configuration

Command	Purpose
show cip {**connection** | **faults** | **file** | **miscellaneous** | **object** | **security**| **session** | **status**}	Displays information about the CIP subsystem.

Troubleshooting CIP

Table 22 Commands for Troubleshooting the CIP Configuration

Command	Purpose
debug cip {**assembly** | **connection manager** | **errors** | **event** | **file** | **io** | **packet** | **request response** | **security** | **session** | **socket**}	Enables debugging of the CIP subsystem.

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	–

MIBs

MIBs	MIBs Link
–	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	–

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Configuring SDM Templates

Prerequisites for Configuring SDM Templates

You must enter the **reload** privileged EXEC command to have your configured SDM template take effect.

Restrictions for Configuring SDM Templates

- When you select and configure SDM templates, you must reload the switch for the configuration to take effect.
- If you try to configure IPv6 features without first selecting a dual IPv4 and IPv6 template, a warning message is generated.
- Using the dual-stack templates results in less TCAM capacity allowed for each resource, so do not use if you plan to forward only IPv4 traffic.

Information About Configuring SDM Templates

SDM Templates

You can use SDM templates to configure system resources in the switch to optimize support for specific features, depending on how the switch is used in the network.

You can select a template to provide maximum system usage for some functions or use the default template to balance resources.

To allocate ternary content addressable memory (TCAM) resources for different usages, the switch SDM templates prioritize system resources to optimize support for certain features. When running the IPservices license, you can select SDM templates to optimize these features:

- Default—The default template gives balance to all Layer 2 functions.
- Dual IPv4 and IPv6—Allows the switch to be used in dual-stack environments (supporting both IPv4 and IPv6).
- Routing—The routing template maximizes system resources for IPv4 unicast routing, typically required for a router or aggregator in the center of a network.

See Dual IPv4 and IPv6 SDM Default Template, page 138.

There are four templates for ip services and one template for lanbase licensing.

Table 23 IP Services license SDM Templates

Resource	Default	IPv4 Routing	Dual-Default	Dual-Routing
Unicast MAC addresses	16 K	16 K	16 K	16 K
IPv4 IGMP or IPv6 groups	1K IPv4	1K IPv4	1K IPv4 1K IPv6	1K IPv4 1K IPv6
Direct routes	16K IPv4	16K IPv4	4K IPv4 4K IPv6	4K IPv4 4K IPv6
Indirect routes	2K IPv4	8K IPv4	1.25K IPv4 1.25K IPv6	2K IPv4 3K IPv6
IPv4 or IPv6 policy-based routing ACEs	0.125K (IPv4 PBR)	0.5K (IPv4 PBR)	0.25K (IPv4 PBR) 0.25K (IPv6 PBR)	0.125K (IPv4 PBR) 0.125K (IPv6 PBR)
IPv4 or IPv6 QoSACEs	1.875K (IPv4 QoS)	0.5K (IPv4 QoS)	0.5K (IPv4 QoS) 0.375K (IPv6 QoS)	0.5K (IPv4 QoS) 0.125K (IPv6 QoS)
IPv4 or IPv6 port or MAC security ACEs	1.875K (IPv4 ACL)	1K (IPv4 ACL)	0.75K (IPv4 ACL) 0.375K (IPv6 ACL)	0.625K (IPv4 ACL) 0.125K (IPv6 ACL)

Table 24 Lanbase license SDM Template

Resource	Default
Unicast MAC addresses	16 K
IPv4 IGMP or IPv6 groups	1K IPv4/1K IPv6
Direct routes	4K IPv4/4K IPv6
Indirect routes	1.25K IPv4/1.25K IPv6
IPv4 or IPv6 policy-based routing ACEs	0.25K (IPv4 PBR)/0.25K (IPv6 PBR)
IPv4 or IPv6 QoSACEs	1K (IPv4 QoS)/0.25K (IPv6 QoS)
IPv4 or IPv6 port or MAC security ACEs	1K (IPv4 ACL)/0.25K (IPv6 ACL)

The first eight rows in the tables (unicast MAC addresses through security ACEs) represent approximate hardware boundaries set when a template is selected. If a section of a hardware resource is full, all processing overflow is sent to the CPU, seriously impacting switch performance.

Dual IPv4 and IPv6 SDM Default Template

You can select an SDM template to support IP Version 6 (IPv6) switching. The dual IPv4 and IPv6 template allows the switch to be used in dual-stack environments (supporting both IPv4 and IPv6). Using the dual-stack templates results in less TCAM capacity allowed for each resource. You should not use this template if you plan to forward only IPv4 traffic.

These SDM templates support IPv4 and IPv6 environments:

- Dual IPv4 and IPv6 default template—Supports Layer 2, QoS, and ACLs for IPv4; and Layer 2, IPv6 host, and ACLs for IPv6.
- Dual IPv4 and IPv6 routing template—Supports Layer 2, multicast, routing (including policy-based routing), QoS, and ACLs for IPv4; and Layer 2, routing, and ACLs for IPv6.

How to Configure the Switch SDM Templates

Setting the SDM Template

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**sdm prefer** {**default** | **dual-ipv4-and-ipv6** {**default**} | **routing**}	Specifies the SDM template to be used on the switch: ■ **default**–Gives balance to all functions. ■ **dual-ipv4-and-ipv6**–Selects a template that supports both IPv4 and IPv6 routing. – **default**–Balances IPv4 and IPv6 Layer 2 functionality. ■ **routing**–Maximizes IPv4 routing on the switch. Use the **no sdm prefer** command to set the switch to the default template. The default template balances the use of system resources.
3.	**end**	Returns to privileged EXEC mode.
4.	**reload**	Reloads the operating system.

Configuration Examples for Configuring SDM Templates

Configuring IP Services Templates: Examples

This is an example of output from the **show sdm prefer default** command:

```
Switch# show sdm prefer default
"IPv4 default" template:
 The selected template optimizes the resources in
 the switch to support this level of features for
 8 routed interfaces and 1024 VLANs.

  number of unicast mac addresses:                  16K
  number of IPv4 IGMP groups + multicast routes:    1K
  number of IPv4 unicast routes:                    18K
    number of directly-connected IPv4 hosts:        16K
    number of indirect IPv4 routes:                 2K
  number of IPv6 multicast groups:                  0
  number of IPv6 unicast routes:                    0
    number of directly-connected IPv6 addresses:    0
    number of indirect IPv6 unicast routes:         0
  number of IPv4 policy based routing aces:         0.125k
  number of IPv4/MAC qos aces:                      1.875k
  number of IPv4/MAC security aces:                 1.875k
  number of IPv6 policy based routing aces:         0
  number of IPv6 qos aces:                          0
  number of IPv6 security aces:                     0
```

This is an example of output from the **show sdm prefer dual-ipv4-and-ipv6 default** command:

```
Switch# show sdm prefer dual-ipv4-and-ipv6 default
"dual IPv4/IPv6 default" template:
 The selected template optimizes the resources in
 the switch to support this level of features for
 8 routed interfaces and 1024 VLANs.

  number of unicast mac addresses:                  16K
  number of IPv4 IGMP groups + multicast routes:    1K
  number of IPv4 unicast routes:                    5.25K
    number of directly-connected IPv4 hosts:        4K
    number of indirect IPv4 routes:                 1.25K
  number of IPv6 multicast groups:                  1K
  number of IPv6 unicast routes:                    5.25K
    number of directly-connected IPv6 addresses:    4K
    number of indirect IPv6 unicast routes:         1.25K
  number of IPv4 policy based routing aces:         0.25K
  number of IPv4/MAC qos aces:                      0.5K
  number of IPv4/MAC security aces:                 0.75K
  number of IPv6 policy based routing aces:         0.25K
  number of IPv6 qos aces:                          0.375k
  number of IPv6 security aces:                     0.375k
```

This is an example of output from the **show sdm prefer dual-ipv4-and-ipv6 routing** command:

```
Switch# show sdm prefer dual-ipv4-and-ipv6 routing
"dual IPv4/IPv6 routing" template:
 The selected template optimizes the resources in
 the switch to support this level of features for
 8 routed interfaces and 1024 VLANs.

  number of unicast mac addresses:                  16K
  number of IPv4 IGMP groups + multicast routes:    1K
  number of IPv4 unicast routes:                    6K
    number of directly-connected IPv4 hosts:        4K
    number of indirect IPv4 routes:                 2K
  number of IPv6 multicast groups:                  1K
  number of IPv6 unicast routes:                    7K
    number of directly-connected IPv6 addresses:    4K
    number of indirect IPv6 unicast routes:         3K
  number of IPv4 policy based routing aces:         0.125k
  number of IPv4/MAC qos aces:                      0.5K
  number of IPv4/MAC security aces:                 0.625k
  number of IPv6 policy based routing aces:         0.125k
  number of IPv6 qos aces:                          0.125k
  number of IPv6 security aces:                     0.125k
```

This is an example of output from the **show sdm prefer routing** command:

```
Switch# show sdm prefer routing
"IPv4 routing" template:
 The selected template optimizes the resources in
 the switch to support this level of features for
 8 routed interfaces and 1024 VLANs.

  number of unicast mac addresses:                  16K
  number of IPv4 IGMP groups + multicast routes:    1K
  number of IPv4 unicast routes:                    24K
    number of directly-connected IPv4 hosts:        16K
    number of indirect IPv4 routes:                 8K
  number of IPv6 multicast groups:                  0
  number of IPv6 unicast routes:                    0
    number of directly-connected IPv6 addresses:    0
    number of indirect IPv6 unicast routes:         0
  number of IPv4 policy based routing aces:         0.375k
  number of IPv4/MAC qos aces:                      0.5K
  number of IPv4/MAC security aces:                 1K
  number of IPv6 policy based routing aces:         0
  number of IPv6 qos aces:                          0
  number of IPv6 security aces:                     0
```

Configuring Lanbase Templates: Example

This is an example of output from the **show sdm prefer** command on a Lanbase image:

```
Switch# show sdm prefer
 The current template is "IPv4 default" template.
 The selected template optimizes the resources in
 the switch to support this level of features for
 8 routed interfaces and 1024 VLANs.

  number of unicast mac addresses:                  16K
  number of IPv4 IGMP groups + multicast routes:    1K
  number of IPv4 unicast routes:                    5.25K
    number of directly-connected IPv4 hosts:        4K
    number of indirect IPv4 routes:                 1.25K
  number of IPv6 multicast groups:                  1K
  number of IPv6 unicast routes:                    5.25K
    number of directly-connected IPv6 addresses:    4K
    number of indirect IPv6 unicast routes:         1.25K
  number of IPv4 policy based routing aces:         0.25K
  number of IPv4/MAC qos aces:                      1K
  number of IPv4/MAC security aces:                 1K
  number of IPv6 policy based routing aces:         0.25K
  number of IPv6 qos aces:                          0.25K
  number of IPv6 security aces:                     0.25K
```

Configuring Switch-Based Authentication

Prerequisites for Configuring Switch-Based Authentication

- If you configure an SDM template and then perform the **show sdm prefer** command, the template currently in use displays.
- You must enter the **reload** privileged EXEC command to have your configured SDM template take effect.
- You should have access to and should configure a RADIUS server before configuring RADIUS features on your switch.
- At a minimum, you must identify the host or hosts that run the RADIUS server software and define the method lists for RADIUS authentication. You can optionally define method lists for RADIUS authorization and accounting.

Restrictions for Configuring Switch-Based Authentication

- To use the Radius CoA interface, a session must already exist on the switch. CoA can be used to identify a session and enforce a disconnect request. The update affects only the specified session.
- To use Secure Shell, you must install the cryptographic (encrypted) software image on your switch. You must obtain authorization to use this feature and to download the cryptographic software files from Cisco.com. For more information, see the release notes for this release.

Information About Configuring Switch-Based Authentication

Prevention for Unauthorized Switch Access

You can prevent unauthorized users from reconfiguring your switch and viewing configuration information. Typically, you want network administrators to have access to your switch while you restrict access to users who dial from outside the network through an asynchronous port, connect from outside the network through a serial port, or connect through a terminal or workstation from within the local network.

To prevent unauthorized access into your switch, you should configure one or more of these security features:

- At a minimum, you should configure passwords and privileges at each switch port. These passwords are locally stored on the switch. When users attempt to access the switch through a port or line, they must enter the password specified for the port or line before they can access the switch.
- For an additional layer of security, you can also configure username and password pairs, which are locally stored on the switch. These pairs are assigned to lines or ports and authenticate each user before that user can access the switch. If you have defined privilege levels, you can also assign a specific privilege level (with associated rights and privileges) to each username and password pair.
- If you want to use username and password pairs, but you want to store them centrally on a server instead of locally, you can store them in a database on a security server. Multiple networking devices can then use the same database to obtain user authentication (and, if necessary, authorization) information.

- You can also enable the login enhancements feature, which logs both failed and unsuccessful login attempts. Login enhancements can also be configured to block future login attempts after a set number of unsuccessful attempts are made.

Password Protection

A simple way of providing terminal access control in your network is to use passwords and assign privilege levels. Password protection restricts access to a network or network device. Privilege levels define what commands users can enter after they have logged into a network device.

Default Password and Privilege Level Configuration

Table 25 Default Password and Privilege Levels

Feature	Default Setting
Enable password and privilege level	No password is defined. The default is level 15 (privileged EXEC level). The password is not encrypted in the configuration file.
Enable secret password and privilege level	No password is defined. The default is level 15 (privileged EXEC level). The password is encrypted before it is written to the configuration file.
Line password	No password is defined.

Enable Secret Passwords with Encryption

To provide an additional layer of security, particularly for passwords that cross the network or that are stored on a Trivial File Transfer Protocol (TFTP) server, you can use either the **enable password** or **enable secret** global configuration commands. Both commands accomplish the same thing; that is, you can establish an encrypted password that users must enter to access privileged EXEC mode (the default) or any privilege level you specify.

We recommend that you use the **enable secret** command because it uses an improved encryption algorithm.

If you configure the **enable secret** command, it takes precedence over the **enable password** command; the two commands cannot be in effect simultaneously.

Use the **level** keyword to define a password for a specific privilege level. After you specify the level and set a password, give the password only to users who need to have access at this level. Use the **privilege level** global configuration command to specify commands accessible at various levels.

If you enable password encryption, it applies to all passwords including username passwords, authentication key passwords, the privileged command password, and console and virtual terminal line passwords.

To remove a password and level, use the **no enable password** [**level** *level*] or **no enable secret** [**level** *level*] global configuration command. To disable password encryption, use the **no service password-encryption** global configuration command.

Password Recovery

Any end user with physical access to the switch can recover from a lost password by interrupting the boot process while the switch is powering on and manually deleting the configuration of the switch.

Press and hold the factory default button when power is applied to the switch. You can release the button once you see the *password-recovery mechanism is enabled* message. You will be at the boot loader prompt at this point and will be able to delete the configuration file (which contains the forgotten password).

Telnet Password for a Terminal Line

When you power-up your switch for the first time, an automatic setup program runs to assign IP information and to create a default configuration for continued use. The setup program also prompts you to configure your switch for Telnet access through a password. If you did not configure this password during the setup program, you can configure it now through the command-line interface (CLI).

Username and Password Pairs

You can configure username and password pairs, which are locally stored on the switch. These pairs are assigned to lines or ports and authenticate each user before that user can access the switch. If you have defined privilege levels, you can also assign a specific privilege level (with associated rights and privileges) to each username and password pair.

Multiple Privilege Levels

By default, the Cisco IOS software has two modes of password security: user EXEC and privileged EXEC. You can configure up to 16 hierarchical levels of commands for each mode. By configuring multiple passwords, you can allow different sets of users to have access to specified commands.

For example, if you want many users to have access to the **clear line** command, you can assign it level 2 security and distribute the level 2 password fairly widely. But if you want more restricted access to the **configure** command, you can assign it level 3 security and distribute that password to a more restricted group of users.

When you set a command to a privilege level, all commands whose syntax is a subset of that command are also set to that level. For example, if you set the **show ip traffic** command to level 15, the **show** commands and **show ip** commands are automatically set to privilege level 15 unless you set them individually to different levels.

To return to the default privilege for a given command, use the **no privilege** *mode* **level** *level command* global configuration command.

Users can override the privilege level you set using the **privilege level** line configuration command by logging in to the line and enabling a different privilege level. They can lower the privilege level by using the **disable** command. If users know the password to a higher privilege level, they can use that password to enable the higher privilege level. You might specify a high level or privilege level for your console line to restrict line usage.

To return to the default line privilege level, use the **no privilege level** line configuration command.

Switch Access with TACACS+

This section describes how to enable and configure Terminal Access Controller Access Control System Plus (TACACS+), which provides detailed accounting information and flexible administrative control over authentication and authorization processes. TACACS+ is facilitated through authentication, authorization, accounting (AAA) and can be enabled only through AAA commands.

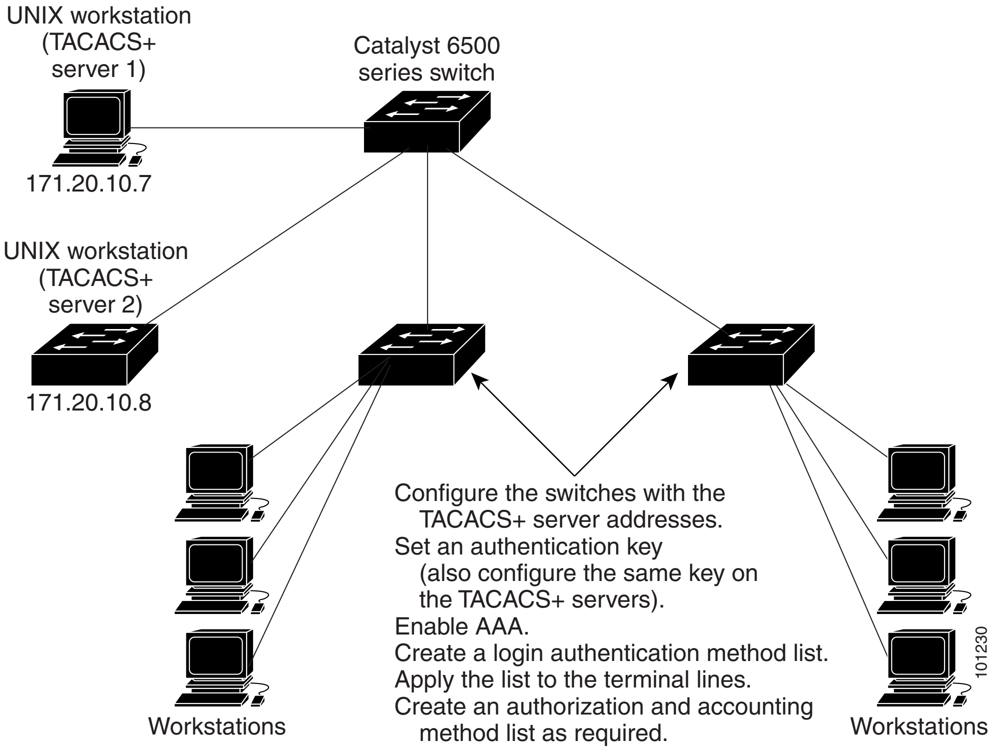
TACACS+

TACACS+ is a security application that provides centralized validation of users attempting to gain access to your switch. TACACS+ services are maintained in a database on a TACACS+ daemon typically running on a UNIX or Windows NT workstation. You should have access to and should configure a TACACS+ server before the configuring TACACS+ features on your switch.

TACACS+ provides for separate and modular authentication, authorization, and accounting facilities. TACACS+ allows for a single access control server (the TACACS+ daemon) to provide each service—authentication, authorization, and accounting—independently. Each service can be tied into its own database to take advantage of other services available on that server or on the network, depending on the capabilities of the daemon.

The goal of TACACS+ is to provide a method for managing multiple network access points from a single management service. Your switch can be a network access server along with other Cisco routers and access servers. A network access server provides connections to a single user, to a network or subnetwork, and to interconnected networks as shown in Figure 16 on page 146.

Figure 16 Typical TACACS+ Network Configuration


UNIX workstation (TACACS+ server 1)
171.20.10.7
Catalyst 6500 series switch
UNIX workstation (TACACS+ server 2)
171.20.10.8
Workstations
Workstations
101230

Configure the switches with the TACACS+ server addresses.
Set an authentication key (also configure the same key on the TACACS+ servers).
Enable AAA.
Create a login authentication method list.
Apply the list to the terminal lines.
Create an authorization and accounting method list as required.

TACACS+, administered through the AAA security services, can provide these services:

- Authentication—Provides complete control of authentication through login and password dialog, challenge and response, and messaging support.

  The authentication facility can conduct a dialog with the user (for example, after a username and password are provided, to challenge a user with several questions, such as home address, mother's maiden name, service type, and social security number). The TACACS+ authentication service can also send messages to user screens. For example, a message could notify users that their passwords must be changed because of the company's password aging policy.

- Authorization—Provides fine-grained control over user capabilities for the duration of the user's session, including but not limited to setting autocommands, access control, session duration, or protocol support. You can also enforce restrictions on what commands a user can execute with the TACACS+ authorization feature.

- Accounting—Collects and sends information used for billing, auditing, and reporting to the TACACS+ daemon. Network managers can use the accounting facility to track user activity for a security audit or to provide information for user billing. Accounting records include user identities, start and stop times, executed commands (such as PPP), number of packets, and number of bytes.

The TACACS+ protocol provides authentication between the switch and the TACACS+ daemon, and it ensures confidentiality because all protocol exchanges between the switch and the TACACS+ daemon are encrypted.

You need a system running the TACACS+ daemon software to use TACACS+ on your switch.

TACACS+ Operation

When a user attempts a simple ASCII login by authenticating to a switch using TACACS+, this process occurs:

1. When the connection is established, the switch contacts the TACACS+ daemon to obtain a username prompt to show to the user. The user enters a username, and the switch then contacts the TACACS+ daemon to obtain a password prompt. The switch displays the password prompt to the user, the user enters a password, and the password is then sent to the TACACS+ daemon.

   TACACS+ allows a dialog between the daemon and the user until the daemon receives enough information to authenticate the user. The daemon prompts for a username and password combination, but can include other items, such as the user's mother's maiden name.

2. The switch eventually receives one of these responses from the TACACS+ daemon:

   - ACCEPT—The user is authenticated and service can begin. If the switch is configured to require authorization, authorization begins at this time.
   - REJECT—The user is not authenticated. The user can be denied access or is prompted to retry the login sequence, depending on the TACACS+ daemon.
   - ERROR—An error occurred at some time during authentication with the daemon or in the network connection between the daemon and the switch. If an ERROR response is received, the switch typically tries to use an alternative method for authenticating the user.
   - CONTINUE—The user is prompted for additional authentication information.

   After authentication, the user undergoes an additional authorization phase if authorization has been enabled on the switch. Users must first successfully complete TACACS+ authentication before proceeding to TACACS+ authorization.

3. If TACACS+ authorization is required, the TACACS+ daemon is again contacted, and it returns an ACCEPT or REJECT authorization response. If an ACCEPT response is returned, the response contains data in the form of attributes that direct the EXEC or NETWORK session for that user and the services that the user can access:

   - Telnet, Secure Shell (SSH), rlogin, or privileged EXEC services
   - Connection parameters, including the host or client IP address, access list, and user timeouts

Default TACACS+ Configuration

TACACS+ and AAA are disabled by default.

To prevent a lapse in security, you cannot configure TACACS+ through a network management application. When enabled, TACACS+ can authenticate users accessing the switch through the CLI.

Note: Although TACACS+ configuration is performed through the CLI, the TACACS+ server authenticates HTTP connections that have been configured with a privilege level of 15.

TACACS+ Server Host and the Authentication Key

You can configure the switch to use a single server or AAA server groups to group existing server hosts for authentication. You can group servers to select a subset of the configured server hosts and use them for a particular service. The server group is used with a global server-host list and contains the list of IP addresses of the selected server hosts.

TACACS+ Login Authentication

To configure AAA authentication, you define a named list of authentication methods and then apply that list to various ports. The method list defines the types of authentication to be performed and the sequence in which they are performed; it must be applied to a specific port before any of the defined authentication methods are performed. The only exception is the default method list (which, by coincidence, is named *default*). The default method list is automatically applied to all ports except those that have a named method list explicitly defined. A defined method list overrides the default method list.

A method list describes the sequence and authentication methods to be queried to authenticate a user. You can designate one or more security protocols to be used for authentication, thus ensuring a backup system for authentication in case the initial method fails. The software uses the first method listed to authenticate users; if that method fails to respond, the software selects the next authentication method in the method list. This process continues until there is successful communication with a listed authentication method or until all defined methods are exhausted. If authentication fails at any point in this cycle—meaning that the security server or local username database responds by denying the user access—the authentication process stops, and no other authentication methods are attempted.

TACACS+ Authorization for Privileged EXEC Access and Network Services

AAA authorization limits the services available to a user. When AAA authorization is enabled, the switch uses information retrieved from the user's profile, which is located either in the local user database or on the security server, to configure the user's session. The user is granted access to a requested service only if the information in the user profile allows it.

You can use the **aaa authorization** global configuration command with the **tacacs+** keyword to set parameters that restrict a user's network access to privileged EXEC mode.

The **aaa authorization exec tacacs+ local** command sets these authorization parameters:

- Use TACACS+ for privileged EXEC access authorization if authentication was performed by using TACACS+.
- Use the local database if authentication was not performed by using TACACS+.

Note: Authorization is bypassed for authenticated users who log in through the CLI even if authorization has been configured.

TACACS+ Accounting

The AAA accounting feature tracks the services that users are accessing and the amount of network resources that they are consuming. When AAA accounting is enabled, the switch reports user activity to the TACACS+ security server in the form of accounting records. Each accounting record contains accounting attribute-value (AV) pairs and is stored on the security server. This data can then be analyzed for network management, client billing, or auditing.

Switch Access with RADIUS

This section describes how to enable and configure the RADIUS, which provides detailed accounting information and flexible administrative control over authentication and authorization processes. RADIUS is facilitated through AAA and can be enabled only through AAA commands.

RADIUS

RADIUS is a distributed client/server system that secures networks against unauthorized access. RADIUS clients run on supported Cisco routers and switches. Clients send authentication requests to a central RADIUS server, which contains all user authentication and network service access information. The RADIUS host is normally a multiuser system running RADIUS server software from Cisco (Cisco Secure Access Control Server Version 3.0), Livingston, Merit, Microsoft, or another software provider. For more information, see the RADIUS server documentation.

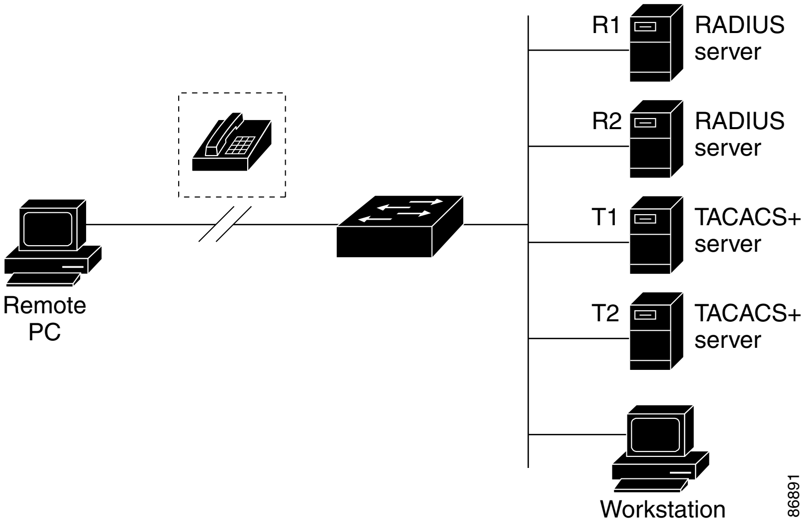
Use RADIUS in these network environments that require access security:

- Networks with multiple-vendor access servers, each supporting RADIUS. For example, access servers from several vendors use a single RADIUS server-based security database. In an IP-based network with multiple vendors' access servers, dial-in users are authenticated through a RADIUS server that has been customized to work with the Kerberos security system.
- Turnkey network security environments in which applications support the RADIUS protocol, such as in an access environment that uses a *smart card* access control system. In one case, RADIUS has been used with Enigma's security cards to validates users and to grant access to network resources.
- Networks already using RADIUS. You can add a Cisco switch containing a RADIUS client to the network. This might be the first step when you make a transition to a TACACS+ server.
- Network in which the user must only access a single service. Using RADIUS, you can control user access to a single host, to a single utility such as Telnet, or to the network through a protocol such as IEEE 802.1x. For more information about this protocol, see Configuring IEEE 802.1x Port-Based Authentication, page 189
- Networks that require resource accounting. You can use RADIUS accounting independently of RADIUS authentication or authorization. The RADIUS accounting functions allow data to be sent at the start and end of services, showing the amount of resources (such as time, packets, bytes, and so forth) used during the session. An Internet service provider might use a freeware-based version of RADIUS access control and accounting software to meet special security and billing needs.

RADIUS is not suitable in these network security situations:

- Multiprotocol access environments. RADIUS does not support AppleTalk Remote Access (ARA), NetBIOS Frame Control Protocol (NBFCP), NetWare Asynchronous Services Interface (NASI), or X.25 PAD connections.
- Switch-to-switch or router-to-router situations. RADIUS does not provide two-way authentication. RADIUS can be used to authenticate from one device to a non-Cisco device if the non-Cisco device requires authentication.
- Networks using a variety of services. RADIUS generally binds a user to one service model.

Figure 17 Transitioning from RADIUS to TACACS+ Services



RADIUS Operation

When a user attempts to log in and authenticate to a switch that is access controlled by a RADIUS server, these events occur:

1. The user is prompted to enter a username and password.
2. The username and encrypted password are sent over the network to the RADIUS server.
3. The user receives one of these responses from the RADIUS server:
   a. ACCEPT—The user is authenticated.
   b. REJECT—The user is either not authenticated and is prompted to re-enter the username and password, or access is denied.
   c. CHALLENGE—A challenge requires additional data from the user.
   d. CHALLENGE PASSWORD—A response requests the user to select a new password.

The ACCEPT or REJECT response is bundled with additional data that is used for privileged EXEC or network authorization. Users must first successfully complete RADIUS authentication before proceeding to RADIUS authorization, if it is enabled. The additional data included with the ACCEPT or REJECT packets includes these items:

- Telnet, SSH, rlogin, or privileged EXEC services
- Connection parameters, including the host or client IP address, access list, and user timeouts

Default RADIUS Configuration

RADIUS and AAA are disabled by default.

To prevent a lapse in security, you cannot configure RADIUS through a network management application. When enabled, RADIUS can authenticate users accessing the switch through the CLI.

RADIUS Change of Authorization

This section provides an overview of the RADIUS interface including available primitives and how they are used during a Change of Authorization (CoA).

Radius COA Overview

A standard RADIUS interface is typically used in a pulled model where the request originates from a network attached device and the response come from the queried servers. Catalyst switches support the RADIUS Change of Authorization (CoA) extensions defined in RFC 5176 that are typically used in a pushed model and allow for the dynamic reconfiguring of sessions from external authentication, authorization, and accounting (AAA) or policy servers.

The switch supports these per-session CoA requests:

- Session reauthentication
- Session termination
- Session termination with port shutdown
- Session termination with port bounce

Change-of-Authorization Requests

Change of Authorization (CoA) requests, as described in RFC 5176, are used in a push model to allow for session identification, host reauthentication, and session termination. The model is comprised of one request (CoA-Request) and two possible response codes:

- CoA acknowledgement (ACK) [CoA-ACK]
- CoA non-acknowledgement (NAK) [CoA-NAK]

The request is initiated from a CoA client (typically a RADIUS or policy server) and directed to the switch that acts as a listener.

RFC 5176 Compliance

The Disconnect Request message, which is also referred to as Packet of Disconnect (POD), is supported by the switch for session termination.

Attribute Number	Attribute Name
24	State
31	Calling-Station-ID
44	Acct-Session-ID
80	Message-Authenticator
101	Error-Cause

Value	Explanation
201	Residual Session Context Removed
202	Invalid EAP Packet (Ignored)
401	Unsupported Attribute
402	Missing Attribute
403	NAS Identification Mismatch
404	Invalid Request
405	Unsupported Service
406	Unsupported Extension
407	Invalid Attribute Value
501	Administratively Prohibited
502	Request Not Routable (Proxy)
503	Session Context Not Found
504	Session Context Not Removable
505	Other Proxy Processing Error
506	Resources Unavailable
507	Request Initiated
508	Multiple Session Selection Unsupported

CoA Request Response Code

The CoA Request response code can be used to convey a command to the switch. The supported commands are listed in Table 26 on page 153.

CoA Session Identification

For disconnect and CoA requests targeted at a particular session, the switch locates the session based on one or more of the following attributes:

- Calling-Station-Id (IETF attribute 31 which contains the host MAC address)
- Audit-Session-Id (Cisco VSA)
- Acct-Session-Id (IETF attribute 44)

Unless all session identification attributes included in the CoA message match the session, the switch returns a Disconnect-NAK or CoA-NAK with the Invalid Attribute Value error-code attribute.

For disconnect and CoA requests targeted to a particular session, any one of these session identifiers can be used:

- Calling-Station-ID (IETF attribute 31, which should contain the MAC address)

- Audit-Session-ID (Cisco vendor-specific attribute)
- Accounting-Session-ID (IETF attribute 44).

If more than one session identification attribute is included in the message, all the attributes must match the session or the switch returns a Disconnect- negative acknowledgement (NAK) or CoA-NAK with the error code *Invalid Attribute Value*.

The packet format for a CoA Request code as defined in RFC 5176 consists of the fields: Code, Identifier, Length, Authenticator, and Attributes in Type:Length:Value (TLV) format.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Code      |  Identifier   |            Length             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                         Authenticator                         |
|                                                               |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Attributes ...
+-+-+-+-+-+-+-+-+-+-+-+-+-
```

The attributes field is used to carry Cisco VSAs.

CoA ACK Response Code

If the authorization state is changed successfully, a positive acknowledgement (ACK) is sent. The attributes returned within CoA ACK will vary based on the CoA Request and are discussed in individual CoA Commands.

CoA NAK Response Code

A negative acknowledgement (NAK) indicates a failure to change the authorization state and can include attributes that indicate the reason for the failure. Use **show** commands to verify a successful CoA.

CoA Request Commands

Table 26 CoA Commands Supported on the Switch

Command[1]	Cisco VSA
Reauthenticate host	Cisco:Avpair="subscriber:command=reauthenticate"
Terminate session	This is a standard disconnect request that does not require a VSA.
Bounce host port	Cisco:Avpair="subscriber:command=bounce-host-port"
Disable host port	Cisco:Avpair="subscriber:command=disable-host-port"

[1] All CoA commands must include the session identifier between the switch and the CoA client.

CoA Session Reauthentication

The AAA server typically generates a session reauthentication request when a host with an unknown identity or posture joins the network and is associated with a restricted access authorization profile (such as a guest VLAN). A reauthentication request allows the host to be placed in the appropriate authorization group when its credentials are known.

To initiate session authentication, the AAA server sends a standard CoA-Request message which contains a Cisco vendor-specific attribute (VSA) in this form: *Cisco:Avpair="subscriber:command=reauthenticate"* and one or more session identification attributes.

The current session state determines the switch response to the message. If the session is currently authenticated by IEEE 802.1x, the switch responds by sending an Extensible Authentication Protocol over LAN (EAPoL) RequestId message to the server.

If the session is currently authenticated by MAC authentication bypass (MAB), the switch sends an access-request to the server, passing the same identity attributes used for the initial successful authentication.

If session authentication is in progress when the switch receives the command, the switch terminates the process, and restarts the authentication sequence, starting with the method configured to be attempted first.

If the session is not yet authorized, or is authorized via guest VLAN, or critical VLAN, or similar policies, the reauthentication message restarts the access control methods, beginning with the method configured to be attempted first. The current authorization of the session is maintained until the reauthentication leads to a different authorization result.

CoA Session Termination

There are three types of CoA requests that can trigger session termination. A CoA Disconnect-Request terminates the session, without disabling the host port. This command causes reinitialization of the authenticator state machine for the specified host, but does not restrict that host's access to the network.

To restrict a host's access to the network, use a CoA Request with the Cisco:Avpair="subscriber:command=disable-host-port" VSA. This command is useful when a host is known to be causing problems on the network, and you need to immediately block network access for the host. When you want to restore network access on the port, reenable it using a non-RADIUS mechanism.

When a device with no supplicant, such as a printer, needs to acquire a new IP address (for example, after a VLAN change), terminate the session on the host port with port-bounce (temporarily disable and then reenable the port).

CoA Disconnect-Request

This command is a standard Disconnect-Request. Because this command is session-oriented, it must be accompanied by one or more of the session identification attributes described in the CoA Session Identification, page 152. If the session cannot be located, the switch returns a Disconnect-NAK message with the "Session Context Not Found" error-code attribute. If the session *is* located, the switch terminates the session. After the session has been completely removed, the switch returns a Disconnect-ACK.

If the switch fails-over to a standby switch before returning a Disconnect-ACK to the client, the process is repeated on the new active switch when the request is resent from the client. If the session is not found following resend, a Disconnect-ACK is sent with the "Session Context Not Found" error-code attribute.

CoA Request: Disable Host Port

This command is carried in a standard CoA-Request message that has this new VSA:

Cisco:Avpair="subscriber:command=disable-host-port"

Because this command is session-oriented, it must be accompanied by one or more of the session identification attributes described in the CoA Session Identification, page 152. If the session cannot be located, the switch returns a CoA-NAK message with the "Session Context Not Found" error-code attribute. If the session is located, the switch disables the hosting port and returns a CoA-ACK message.

If the switch fails before returning a CoA-ACK to the client, the process is repeated on the new active switch when the request is resent from the client. If the switch fails after returning a CoA-ACK message to the client but before the operation has completed, the operation is restarted on the new active switch.

Note: A Disconnect-Request failure following command resend could be the result of either a successful session termination before change-over (if the Disconnect-ACK was not sent) or a session termination by other means (for example, a link failure) that occurred after the original command was issued and before the standby switch became active.

CoA Request: Bounce-Port

This command is carried in a standard CoA-Request message that contains this VSA:
Cisco:Avpair=" subscriber:command=bounce-host-port"

Because this command is session-oriented, it must be accompanied by one or more of the session identification attributes described in the CoA Session Identification, page 152. If the session cannot be located, the switch returns a CoA-NAK message with the “Session Context Not Found” error-code attribute. If the session is located, the switch disables the hosting port for a period of 10 seconds, reenables it (port-bounce), and returns a CoA-ACK.

If the switch fails before returning a CoA-ACK to the client, the process is repeated on the new active switch when the request is resent from the client. If the switch fails after returning a CoA-ACK message to the client but before the operation has completed, the operation is restarted on the new active switch.

RADIUS Server Host

Switch-to-RADIUS-server communication involves several components:

- Hostname or IP address
- Authentication destination port
- Accounting destination port
- Key string
- Timeout period
- Retransmission value

You identify RADIUS security servers by their hostname or IP address, hostname and specific UDP port numbers, or their IP address and specific UDP port numbers. The combination of the IP address and the UDP port number creates a unique identifier, allowing different ports to be individually defined as RADIUS hosts providing a specific AAA service. This unique identifier enables RADIUS requests to be sent to multiple UDP ports on a server at the same IP address.

If two different host entries on the same RADIUS server are configured for the same service—for example, accounting—the second host entry configured acts as a fail-over backup to the first one. Using this example, if the first host entry fails to provide accounting services, the `%RADIUS-4-RADIUS_DEAD` message appears, and then the switch tries the second host entry configured on the same device for accounting services. (The RADIUS host entries are tried in the order that they are configured.)

A RADIUS server and the switch use a shared secret text string to encrypt passwords and exchange responses. To configure RADIUS to use the AAA security commands, you must specify the host running the RADIUS server daemon and a secret text (key) string that it shares with the switch.

The timeout, retransmission, and encryption key values can be configured globally for all RADIUS servers, on a per-server basis, or in some combination of global and per-server settings. To apply these settings globally to all RADIUS servers communicating with the switch, use the three unique global configuration commands: **radius-server timeout**, **radius-server retransmit**, and **radius-server key**. To apply these values on a specific RADIUS server, use the **radius-server host** global configuration command.

Note: If you configure both global and per-server functions (timeout, retransmission, and key commands) on the switch, the per-server timer, retransmission, and key value commands override global timer, retransmission, and key value commands. For information on configuring these settings on all RADIUS servers, see Configuring Settings for All RADIUS Servers, page 176.

You can configure the switch to use AAA server groups to group existing server hosts for authentication. For more information, see Defining AAA Server Groups, page 174.

RADIUS Login Authentication

To configure AAA authentication, you define a named list of authentication methods and then apply that list to various ports. The method list defines the types of authentication to be performed and the sequence in which they are performed; it must be applied to a specific port before any of the defined authentication methods are performed. The only exception is the default method list (which, by coincidence, is named *default*). The default method list is automatically applied to all ports except those that have a named method list explicitly defined.

Radius Method List

A method list defines the sequence and methods to be used to authenticate, to authorize, or to keep accounts on a user. You can use method lists to designate one or more security protocols to be used (such as TACACS+ or local username lookup), which ensures a backup system if the initial method fails. The software uses the first method listed to authenticate, to authorize, or to keep accounts on users. If that method does not respond, the software selects the next method in the list. This process continues until there is successful communication with a listed method or the method list is exhausted.

AAA Server Groups

You can configure the switch to use AAA server groups to group existing server hosts for authentication. You select a subset of the configured server hosts and use them for a particular service. The server group is used with a global server-host list, which lists the IP addresses of the selected server hosts.

Server groups also can include multiple host entries for the same server if each entry has a unique identifier (the combination of the IP address and UDP port number), allowing different ports to be individually defined as RADIUS hosts providing a specific AAA service. If you configure two different host entries on the same RADIUS server for the same service, (for example, accounting), the second configured host entry acts as a failover backup to the first one.

You use the **server** group server configuration command to associate a particular server with a defined group server. You can either identify the server by its IP address or identify multiple host instances or entries by using the optional **auth-port** and **acct-port** keywords.

RADIUS Authorization for User Privileged Access and Network Services

AAA authorization limits the services available to a user. When AAA authorization is enabled, the switch uses information retrieved from the user’s profile, which is in the local user database or on the security server, to configure the user’s session. The user is granted access to a requested service only if the information in the user profile allows it.

You can use the **aaa authorization** global configuration command with the **radius** keyword to set parameters that restrict a user’s network access to privileged EXEC mode.

The **aaa authorization exec radius local** command sets these authorization parameters:

- Use RADIUS for privileged EXEC access authorization if authentication was performed by using RADIUS.
- Use the local database if authentication was not performed by using RADIUS.

Note: Authorization is bypassed for authenticated users who log in through the CLI even if authorization has been configured.

RADIUS Accounting

The AAA accounting feature tracks the services that users are accessing and the amount of network resources that they are consuming. When AAA accounting is enabled, the switch reports user activity to the RADIUS security server in the form of accounting records. Each accounting record contains accounting attribute-value (AV) pairs and is stored on the security server. This data can then be analyzed for network management, client billing, or auditing.

Establishing a Session with a Router if the AAA Server is Unreachable

The **aaa accounting system guarantee-first** command guarantees system accounting as the first record, which is the default condition. In some situations, users might be prevented from starting a session on the console or terminal connection until after the system reloads, which can take more than 3 minutes.

To establish a console or Telnet session with the router if the AAA server is unreachable when the router reloads, use the **no aaa accounting system guarantee-first** command.

Vendor-Specific RADIUS Attributes

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

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

protocol is a value of the Cisco protocol attribute for a particular type of authorization. *Attribute* and *value* are an appropriate attribute-value (AV) pair defined in the Cisco TACACS+ specification, and *sep* is = for mandatory attributes and is * for optional attributes. The full set of features available for TACACS+ authorization can then be used for RADIUS.

For example, this AV pair activates Cisco's *multiple named ip address pools* feature during IP authorization (during PPP IPCP address assignment):

```
cisco-avpair= ”ip:addr-pool=first“
```

Vendor-Proprietary RADIUS Server Communication

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

As mentioned earlier, to configure RADIUS (whether vendor-proprietary or IETF draft-compliant), you must specify the host running the RADIUS server daemon and the secret text string it shares with the switch. You specify the RADIUS host and secret text string by using the **radius-server** global configuration commands.

Switch Access with Kerberos

This section describes how to enable and configure the Kerberos security system, which authenticates requests for network resources by using a trusted third party. To use this feature, the cryptographic (that is, supports encryption) versions of the switch software must be installed on your switch.

You must obtain authorization to use this feature and to download the cryptographic software files from Cisco.com. For more information, see the release notes for this release.

Understanding Kerberos

Kerberos is a secret-key network authentication protocol, which was developed at the Massachusetts Institute of Technology (MIT). It uses the Data Encryption Standard (DES) cryptographic algorithm for encryption and authentication and authenticates requests for network resources. Kerberos uses the concept of a trusted third party to perform secure verification of users and services. This trusted third party is called the *key distribution center* (KDC).

Kerberos verifies that users are who they claim to be and the network services that they use are what the services claim to be. To do this, a KDC or trusted Kerberos server issues tickets to users. These tickets, which have a limited lifespan, are stored in user credential caches. The Kerberos server uses the tickets instead of usernames and passwords to authenticate users and network services.

Note: A Kerberos server can be a switch that is configured as a network security server and that can authenticate users by using the Kerberos protocol.

The Kerberos credential scheme uses a process called *single logon*. This process authenticates a user once and then allows secure authentication (without encrypting another password) wherever that user credential is accepted.

This software release supports Kerberos 5, which allows organizations that are already using Kerberos 5 to use the same Kerberos authentication database on the KDC that they are already using on their other network hosts (such as UNIX servers and PCs).

In this software release, Kerberos supports these network services:

- Telnet
- rlogin
- rsh (Remote Shell Protocol)

Table 5 lists the common Kerberos-related terms and definitions.

Table 27 Kerberos-related terms

Term	Definition
Authentication	A process by which a user or service identifies itself to another service. For example, a client can authenticate to a switch or a switch can authenticate to another switch.
Authorization	A means by which the switch identifies what privileges the user has in a network or on the switch and what actions the user can perform.
Credential	A general term that refers to authentication tickets, such as TGTs[1] and service credentials. Kerberos credentials verify the identity of a user or service. If a network service decides to trust the Kerberos server that issued a ticket, it can be used in place of reentering a username and password. Credentials have a default lifespan of eight hours.
Instance	An authorization level label for Kerberos principals. Most Kerberos principals are of the form *user@REALM* (for example, smith@EXAMPLE.COM). A Kerberos principal with a Kerberos instance has the form *user/instance@REALM* (for example, smith/admin@EXAMPLE.COM). The Kerberos instance can be used to specify the authorization level for the user if authentication is successful. The server of each network service might implement and enforce the authorization mappings of Kerberos instances but is not required to do so. **Note:** The Kerberos principal and instance names *must* be in all lowercase characters. **Note:** The Kerberos realm name *must* be in all uppercase characters.
KDC[2]	Key distribution center that consists of a Kerberos server and database program that is running on a network host.
Kerberized	A term that describes applications and services that have been modified to support the Kerberos credential infrastructure.
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. **Note:** The Kerberos realm name *must* be in all uppercase characters.
Kerberos server	A daemon that is 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.
KEYTAB[3]	A password that a network service shares with the KDC. In Kerberos 5 and later Kerberos versions, the network service authenticates an encrypted service credential by using the KEYTAB to decrypt it. In Kerberos versions earlier than Kerberos 5, KEYTAB is referred to as SRVTAB[4].
Principal	Also known as a Kerberos identity, this is who you are or what a service is according to the Kerberos server. **Note:** The Kerberos principal name *must* be in all lowercase characters.
Service credential	A credential for a network service. When issued from the KDC, this credential is encrypted with the password shared by the network service and the KDC. The password is also shared with the user TGT.
SRVTAB	A password that a network service shares with the KDC. In Kerberos 5 or later Kerberos versions, SRVTAB is referred to as KEYTAB.
TGT	Ticket granting ticket that is a credential that the KDC issues to authenticated users. When users receive a TGT, they can authenticate to network services within the Kerberos realm represented by the KDC.

1. TGT = ticket granting ticket
2. KDC = key distribution center
3. KEYTAB = key table
4. SRVTAB = server table

Kerberos Operation

A Kerberos server can be a switch that is configured as a network security server and that can authenticate remote users by using the Kerberos protocol. Although you can customize Kerberos in a number of ways, remote users attempting to access network services must pass through three layers of security before they can access network services.

To authenticate to network services by using a switch as a Kerberos server, remote users must follow these steps:

1. Authenticating to a Boundary Switch, page 160
2. Obtaining a TGT from a KDC, page 160
3. Authenticating to Network Services, page 160

Authenticating to a Boundary Switch

This section describes the first layer of security through which a remote user must pass. The user must first authenticate to the boundary switch. This process then occurs:

1. The user opens an un-Kerberized Telnet connection to the boundary switch.
2. The switch prompts the user for a username and password.
3. The switch requests a TGT from the KDC for this user.
4. The KDC sends an encrypted TGT that includes the user identity to the switch.
5. The switch attempts to decrypt the TGT by using the password that the user entered.
   - If the decryption is successful, the user is authenticated to the switch.
   - If the decryption is not successful, the user repeats Step 2 either by reentering the username and password (noting if Caps Lock or Num Lock is on or off) or by entering a different username and password.

A remote user who initiates a un-Kerberized Telnet session and authenticates to a boundary switch is inside the firewall, but the user must still authenticate directly to the KDC before getting access to the network services. The user must authenticate to the KDC because the TGT that the KDC issues is stored on the switch and cannot be used for additional authentication until the user logs on to the switch.

Obtaining a TGT from a KDC

This section describes the second layer of security through which a remote user must pass. The user must now authenticate to a KDC and obtain a TGT from the KDC to access network services.

Authenticating to Network Services

This section describes the third layer of security through which a remote user must pass. The user with a TGT must now authenticate to the network services in a Kerberos realm.

Kerberos Configuration

So that remote users can authenticate to network services, you must configure the hosts and the KDC in the Kerberos realm to communicate and mutually authenticate users and network services. To do this, you must identify them to each other. You add entries for the hosts to the Kerberos database on the KDC and add KEYTAB files generated by the KDC to all hosts in the Kerberos realm. You also create entries for the users in the KDC database.

When you add or create entries for the hosts and users, follow these guidelines:

- The Kerberos principal name *must* be in all lowercase characters.
- The Kerberos instance name *must* be in all lowercase characters.
- The Kerberos realm name *must* be in all uppercase characters.

Note: A Kerberos server can be a switch that is configured as a network security server and that can authenticate users by using the Kerberos protocol.

To set up a Kerberos-authenticated server-client system, follow these steps:

- Configure the KDC by using Kerberos commands.
- Configure the switch to use the Kerberos protocol.

Local Authentication and Authorization

You can configure AAA to operate without a server by setting the switch to implement AAA in local mode. The switch then handles authentication and authorization. No accounting is available in this configuration.

Secure Shell

To use this feature, you must install the cryptographic (encrypted) software image on your switch. You must obtain authorization to use this feature and to download the cryptographic software files from Cisco.com. For more information, see the release notes for this release.

For SSH configuration examples, see the “SSH Configuration Examples” section in the “Configuring Secure Shell” chapter of the *Cisco IOS Security Configuration Guide, Cisco IOS Release 12.2*.

SSH in IPv6 functions the same and offers the same benefits as SSH in IPv4. IPv6 enhancements to SSH consist of support for IPv6 addresses that enable a Cisco router to accept and establish secure, encrypted connections with remote IPv6 nodes over an IPv6 transport.

SSH

SSH is a protocol that provides a secure, remote connection to a device. SSH provides more security for remote connections than Telnet does by providing strong encryption when a device is authenticated. This software release supports SSH Version 1 (SSHv1) and SSH Version 2 (SSHv2).

SSH Servers, Integrated Clients, and Supported Versions

The SSH feature has an SSH server and an SSH integrated client, which are applications that run on the switch. You can use an SSH client to connect to a switch running the SSH server. The SSH server works with the SSH client supported in this release and with non-Cisco SSH clients. The SSH client also works with the SSH server supported in this release and with non-Cisco SSH servers.

The switch supports an SSHv1 or an SSHv2 server.

The switch supports an SSHv1 client.

SSH supports the Data Encryption Standard (DES) encryption algorithm, the Triple DES (3DES) encryption algorithm, and password-based user authentication.

SSH also supports these user authentication methods:

- TACACS+ (for more information, see Configuring TACACS+, page 169)
- RADIUS (for more information, see Configuring Radius Server Communication, page 172)
- Local authentication and authorization (for more information, see Configuring the Switch for Local Authentication and Authorization, page 178)

Note: This software release does not support IP Security (IPSec).

Limitations

These limitations apply to SSH:

- The switch supports Rivest, Shamir, and Adelman (RSA) authentication.
- SSH supports only the execution-shell application.
- The SSH server and the SSH client are supported only on DES (56-bit) and 3DES (168-bit) data encryption software.
- The switch supports the Advanced Encryption Standard (AES) encryption algorithm with a 128-bit key, 192-bit key, or 256-bit key. However, symmetric cipher AES to encrypt the keys is not supported.

SSH Configuration Guidelines

Follow these guidelines when configuring the switch as an SSH server or SSH client:

- An RSA key pair generated by a SSHv1 server can be used by an SSHv2 server, and the reverse.
- If you get CLI error messages after entering the **crypto key generate rsa** global configuration command, an RSA key pair has not been generated. Reconfigure the hostname and domain, and then enter the **crypto key generate rsa** command. For more information, see Setting Up the Switch to Run SSH, page 179.
- When generating the RSA key pair, the message `No host name specified` might appear. If it does, you must configure a hostname by using the **hostname** global configuration command.
- When generating the RSA key pair, the message `No domain specified` might appear. If it does, you must configure an IP domain name by using the **ip domain-name** global configuration command.
- When configuring the local authentication and authorization authentication method, make sure that AAA is disabled on the console.

Switch for Secure Socket Layer HTTP

Secure Socket Layer (SSL) version 3.0 supports the HTTP 1.1 server and client. SSL provides server authentication, encryption, and message integrity, as well as HTTP client authentication, to allow secure HTTP communications. To use this feature, the cryptographic (encrypted) software image must be installed on your switch. You must obtain authorization to use this feature and to download the cryptographic software files from Cisco.com. For more information about the crypto image, see the release notes for this release.

Secure HTTP Servers and Clients

On a secure HTTP connection, data to and from an HTTP server is encrypted before being sent over the Internet. HTTP with SSL encryption provides a secure connection to allow such functions as configuring a switch from a Web browser. Cisco's implementation of the secure HTTP server and secure HTTP client uses an implementation of SSL Version 3.0 with application-layer encryption. HTTP over SSL is abbreviated as HTTPS; the URL of a secure connection begins with https:// instead of http://.

The primary role of the HTTP secure server (the switch) is to listen for HTTPS requests on a designated port (the default HTTPS port is 443) and pass the request to the HTTP 1.1 Web server. The HTTP 1.1 server processes requests and passes responses (pages) back to the HTTP secure server, which responds to the original request.

The primary role of the HTTP secure client (the web browser) is to respond to Cisco IOS application requests for HTTPS User Agent services, perform HTTPS User Agent services for the application, and pass the response back to the application.

When SSL is used in a switch cluster, the SSL session terminates at the cluster commander. Cluster member switches must run standard HTTP.

For secure HTTP connections, we recommend that you configure an official CA trustpoint. A CA trustpoint is more secure than a self-signed certificate.

Before you configure a CA trustpoint, you should ensure that the system clock is set. If the clock is not set, the certificate is rejected due to an incorrect date.

Default SSL Settings

Table 28 Default SSL Settings

Default Setting
The standard HTTP server is enabled.
SSL is enabled.
No CA trustpoints are configured.
No self-signed certificates are generated.

Certificate Authority Trustpoints

Certificate authorities (CAs) manage certificate requests and issue certificates to participating network devices. These services provide centralized security key and certificate management for the participating devices. Specific CA servers are referred to as *trustpoints*.

When a connection attempt is made, the HTTPS server provides a secure connection by issuing a certified X.509v3 certificate, obtained from a specified CA trustpoint, to the client. The client (usually a Web browser), in turn, has a public key that allows it to authenticate the certificate.

For secure HTTP connections, we highly recommend that you configure a CA trustpoint. If a CA trustpoint is not configured for the device running the HTTPS server, the server certifies itself and generates the needed RSA key pair. Because a self-certified (self-signed) certificate does not provide adequate security, the connecting client generates a notification that the certificate is self-certified, and the user has the opportunity to accept or reject the connection. This option is useful for internal network topologies (such as testing).

If you do not configure a CA trustpoint, when you enable a secure HTTP connection, either a temporary or a persistent self-signed certificate for the secure HTTP server (or client) is automatically generated.

- If the switch is not configured with a hostname and a domain name, a temporary self-signed certificate is generated. If the switch reboots, any temporary self-signed certificate is lost, and a new temporary new self-signed certificate is assigned.
- If the switch has been configured with a host and domain name, a persistent self-signed certificate is generated. This certificate remains active if you reboot the switch or if you disable the secure HTTP server so that it will be there the next time you reenable a secure HTTP connection.

Note: The certificate authorities and trustpoints must be configured on each device individually. Copying them from other devices makes them invalid on the switch.

Note: The values that follow *TP self-signed* depend on the serial number of the device.

You can use an optional command (**ip http secure-client-auth**) to allow the HTTPS server to request an X.509v3 certificate from the client. Authenticating the client provides more security than server authentication by itself.

CipherSuites

A CipherSuite specifies the encryption algorithm and the digest algorithm to use on a SSL connection. When connecting to the HTTPS server, the client Web browser offers a list of supported CipherSuites, and the client and server negotiate the best encryption algorithm to use from those on the list that are supported by both. For example, Netscape Communicator 4.76 supports U.S. security with RSA Public Key Cryptography, MD2, MD5, RC2-CBC, RC4, DES-CBC, and DES-EDE3-CBC.

For the best possible encryption, you should use a client browser that supports 128-bit encryption, such as Microsoft Internet Explorer Version 5.5 (or later) or Netscape Communicator Version 4.76 (or later). The SSL_RSA_WITH_DES_CBC_SHA CipherSuite provides less security than the other CipherSuites, as it does not offer 128-bit encryption.

The more secure and more complex CipherSuites require slightly more processing time. This list defines the CipherSuites supported by the switch and ranks them from fastest to slowest in terms of router processing load (speed):

1. SSL_RSA_WITH_DES_CBC_SHA—RSA key exchange (RSA Public Key Cryptography) with DES-CBC for message encryption and SHA for message digest
2. SSL_RSA_WITH_RC4_128_MD5—RSA key exchange with RC4 128-bit encryption and MD5 for message digest
3. SSL_RSA_WITH_RC4_128_SHA—RSA key exchange with RC4 128-bit encryption and SHA for message digest
4. SSL_RSA_WITH_3DES_EDE_CBC_SHA—RSA key exchange with 3DES and DES-EDE3-CBC for message encryption and SHA for message digest

RSA (in conjunction with the specified encryption and digest algorithm combinations) is used for both key generation and authentication on SSL connections. This usage is independent of whether or not a CA trustpoint is configured.

Secure Copy Protocol

The Secure Copy Protocol (SCP) feature provides a secure and authenticated method for copying switch configurations or switch image files. SCP relies on Secure Shell (SSH), an application and a protocol that provides a secure replacement for the Berkeley r-tools.

For SSH to work, the switch needs an RSA public/private key pair. This is the same with SCP, which relies on SSH for its secure transport.

Because SSH also relies on AAA authentication, and SCP relies further on AAA authorization, correct configuration is necessary.

- Before enabling SCP, you must correctly configure SSH, authentication, and authorization on the switch.

- Because SCP relies on SSH for its secure transport, the switch must have an Rivest, Shamir, and Adelman (RSA) key pair.

Note: When using SCP, you cannot enter the password into the copy command. You must enter the password when prompted.

The behavior of SCP is similar to that of remote copy (rcp), which comes from the Berkeley r-tools suite, except that SCP relies on SSH for security. SCP also requires that authentication, authorization, and accounting (AAA) authorization be configured so the router can determine whether the user has the correct privilege level.

A user who has appropriate authorization can use SCP to copy any file in the Cisco IOS File System (IFS) to and from a switch by using the **copy** command. An authorized administrator can also do this from a workstation.

Note: For information about how to configure and verify SCP, see the “Secure Copy Protocol” section in the *Cisco IOS Security Configuration Guide: Securing User Services, Release 12.4*: http://www.cisco.com/en/US/docs/ios/sec_user_services/configuration/guide/sec_secure_copy_ps6350_TSD_Products_Configuration_Guide_Chapter.html

How to Configure Switch-Based Authentication

Configuring Password Protection

Setting or Changing a Static Enable Password

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**enable password** *password*	Defines a new password or changes an existing password for access to privileged EXEC mode. By default, no password is defined. *password*–Specifies a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. It can contain the question mark (?) character if you precede the question mark with the key combination Crtl-v when you create the password; for example, to create the password abc?123, do this: Enter **abc**. Press **Crtl-v**. Enter **?123**. When the system prompts you to enter the enable password, you need not precede the question mark by pressing Ctrl V; you can enter **abc?123** at the password prompt.
3.	**end**	Returns to privileged EXEC mode.

Protecting Enable and Enable Secret Passwords with Encryption

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.

2.	**enable password** [**level** *level*] {*password* | *encryption-type encrypted-password*}
	or
	enable secret [**level** *level*] {*password* | *encryption-type encrypted-password*}
		Defines a new password or changes an existing password for access to privileged EXEC mode.
		or
		Defines a secret password, which is saved using a nonreversible encryption method.
		■ (Optional) *level*—Specifies the range is from 0 to 15. Level 1 is normal user EXEC mode privileges. The default level is 15 (privileged EXEC mode privileges).
		■ *password*—Specifies a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. By default, no password is defined.
		■ (Optional) *encryption-type*—Only type 5, a Cisco proprietary encryption algorithm, is available. If you specify an encryption type, you must provide an encrypted password—an encrypted password that you copy from another switch configuration.
		Note: If you specify an encryption type and then enter a clear text password, you cannot reenter privileged EXEC mode. You cannot recover a lost encrypted password by any method.

3.	**service password-encryption**	(Optional) Encrypts the password when the password is defined or when the configuration is written. Encryption prevents the password from being readable in the configuration file.
4.	**end**	Returns to privileged EXEC mode.

Disabling Password Recovery

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**no service password-recovery**	Disables password recovery. This setting is saved in an area of the flash memory that is accessible by the boot loader and the Cisco IOS image, but it is not part of the file system and is not accessible by any user.
3.	**end**	Returns to privileged EXEC mode.
4.	**show version**	Verifies the configuration by checking the last few lines of the command output.

Setting a Telnet Password for a Terminal Line

	Command	Purpose
1.		Attaches a PC or workstation with emulation software to the switch console port. The default data characteristics of the console port are 9600, 8, 1, no parity. You might need to press the Return key several times to see the command-line prompt.
2.	**enable password** *password*	Enters privileged EXEC mode.
3.	**configure terminal**	Enters global configuration mode.
4.	**line vty 0 15**	Configures the number of Telnet sessions (lines), and enters line configuration mode. There are 16 possible sessions on a command-capable switch. The 0 and 15 mean that you are configuring all 16 possible Telnet sessions.
5.	**password** *password*	Enters a Telnet password for the line or lines. *password*–Specifies a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. By default, no password is defined.
6.	**end**	Returns to privileged EXEC mode.

Configuring Username and Password Pairs

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**username** *name* [**privilege** *level*] {**password** *encryption-type password*}	Enters the username, privilege level, and password for each user. ■ *name*–Specifies the user ID as one word. Spaces and quotation marks are not allowed. ■ (Optional) *level*–Specifies the privilege level the user has after gaining access. The range is 0 to 15. Level 15 gives privileged EXEC mode access. Level 1 gives user EXEC mode access. ■ *encryption-type*–Enters 0 to specify that an unencrypted password will follow. Enter 7 to specify that a hidden password will follow. ■ *password*–Specifies the password the user must enter to gain access to the switch. The password must be from 1 to 25 characters, can contain embedded spaces, and must be the last option specified in the **username** command. ■ To disable username authentication for a specific user, use the **no username** *name* global configuration command.

	Command	Purpose
3.	**line console 0** or **line vty 0 15**	Enters line configuration mode, and configure the console port (line 0) or the VTY lines (line 0 to 15).
4.	**login local**	Enables local password checking at login time. Authentication is based on the username specified in Step 2. To disable password checking and allow connections without a password, use the **no login** line configuration command.
5.	**end**	Returns to privileged EXEC mode.

Setting the Privilege Level for a Command

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**privilege** *mode* **level** *level command*	Sets the privilege level for a command. ■ *mode*–Enters **configure** for global configuration mode, **exec** for EXEC mode, **interface** for interface configuration mode, or **line** for line configuration mode. ■ *level*–The range is from 0 to 15. Level 1 is for normal user EXEC mode privileges. Level 15 is the level of access permitted by the **enable** password. ■ *command*–Specifies the command to which you want to restrict access.
3.	**enable password level** *level password*	Specifies the enable password for the privilege level. ■ *level*–The range is from 0 to 15. Level 1 is for normal user EXEC mode privileges. ■ *password*–Specifies a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. By default, no password is defined.
4.	**end**	Returns to privileged EXEC mode.
5.	**show privilege**	Verifies the password and accesses level configuration.

Changing the Default Privilege Level for Lines

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**line vty** *line*	Selects the virtual terminal line on which to restrict access.

	Command	Purpose
3.	**privilege level** *level*	Changes the default privilege level for the line. *level*–The range is from 0 to 15. Level 1 is for normal user EXEC mode privileges. Level 15 is the level of access permitted by the **enable** password.
4.	**end**	Returns to privileged EXEC mode.
5.	**show privilege**	Verifies the password and accesses level configuration.

Logging Into and Exiting a Privilege Level

Command	Purpose
enable *level*	Logs in to a specified privilege level. *level*–The range is 0 to 15.
disable *level*	Exits to a specified privilege level. *level*–The range is 0 to 15.

Configuring TACACS+

This section describes how to configure your switch to support TACACS+. At a minimum, you must identify the host or hosts maintaining the TACACS+ daemon and define the method lists for TACACS+ authentication. You can optionally define method lists for TACACS+ authorization and accounting. A method list defines the sequence and methods to be used to authenticate, to authorize, or to keep accounts on a user. You can use method lists to designate one or more security protocols to be used, thus ensuring a backup system if the initial method fails. The software uses the first method listed to authenticate, to authorize, or to keep accounts on users; if that method does not respond, the software selects the next method in the list. This process continues until there is successful communication with a listed method or the method list is exhausted.

Identifying the TACACS+ Server Host and Setting the Authentication Key

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**tacacs-server host** *hostname* [**port** *integer*] [**timeout** *integer*] [**key** *string*]	Identifies the IP host or hosts maintaining a TACACS+ server. Enters this command multiple times to create a list of preferred hosts. The software searches for hosts in the order in which you specify them. ■ *hostname*–Specifies the name or IP address of the host. ■ (Optional) **port** *integer*–Specifies a server port number. The default is port 49. The range is 1 to 65535. ■ (Optional) **timeout** *integer*–Specifies a time in seconds the switch waits for a response from the daemon before it times out and declares an error. The default is 5 seconds. The range is 1 to 1000 seconds. ■ (Optional) **key** *string*–Specifies the encryption key for encrypting and decrypting all traffic between the switch and the TACACS+ daemon. You must configure the same key on the TACACS+ daemon for encryption to be successful.
3.	**aaa new-model**	Enables AAA.
4.	**aaa group server tacacs+** *group-name*	(Optional) Defines the AAA server-group with a group name. This command puts the switch in a server group subconfiguration mode.
5.	**server** *ip-address*	(Optional) Associates a particular TACACS+ server with the defined server group. Repeat this step for each TACACS+ server in the AAA server group. Each server in the group must be previously defined in Step 2.
6.	**end**	Returns to privileged EXEC mode.
7.	**show tacacs**	Verifies your entries.

Configuring TACACS+ Login Authentication

Before You Begin

To secure the switch for HTTP access by using AAA methods, you must configure the switch with the **ip http authentication aaa** global configuration command. Configuring AAA authentication does not secure the switch for HTTP access by using AAA methods.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**aaa new-model**	Enables AAA.
3.	**aaa authentication login** {**default** | *list-name*} *method1* [*method2...*]	Creates a login authentication method list. ■ To create a default list that is used when a named list is *not* specified in the **login authentication** command, use the **default** keyword followed by the methods that are to be used in default situations. The default method list is automatically applied to all ports. ■ *list-name*—Specifies a character string to name the list you are creating. ■ *method1...*—Specifies the actual method the authentication algorithm tries. The additional methods of authentication are used only if the previous method returns an error, not if it fails. Select one of these methods: ■ **enable**—Uses the enable password for authentication. Before you can use this authentication method, you must define an enable password by using the **enable** *password* global configuration command. ■ **group tacacs+**—Uses TACACS+ authentication. Before you can use this authentication method, you must configure the TACACS+ server. For more information, see Identifying the TACACS+ Server Host and Setting the Authentication Key, page 170. ■ **line**—Uses the line password for authentication. Before you can use this authentication method, you must define a line password. Use the **password** *password* line configuration command. ■ **local**—Uses the local username database for authentication. You must enter username information in the database. Use the **username** *password* global configuration command. ■ **local-case**—Uses a case-sensitive local username database for authentication. You must enter username information in the database by using the **username** *name* **password** global configuration command. ■ **none**—Does not use any authentication for login.
4.	**line** [**console** | **tty** | **vty**] *line-number* [*ending-line-number*]	Enters line configuration mode, and configures the lines to which you want to apply the authentication list.
5.	**login authentication** {**default** | *list-name*}	Applies the authentication list to a line or set of lines. ■ If you specify **default**, use the default list created with the **aaa authentication login** command. ■ *list-name*—Specifies the list created with the **aaa authentication login** command.
6.	**end**	Returns to privileged EXEC mode.

Configuring TACACS+ Authorization for Privileged EXEC Access and Network Services

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**aaa authorization network tacacs+**	Configures the switch for user TACACS+ authorization for all network-related service requests.
3.	**aaa authorization exec tacacs+**	Configures the switch for user TACACS+ authorization if the user has privileged EXEC access. The **exec** keyword might return user profile information (such as **autocommand** information).
4.	**end**	Returns to privileged EXEC mode.

Starting TACACS+ Accounting

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**aaa accounting network start-stop tacacs+**	Enables TACACS+ accounting for all network-related service requests.
3.	**aaa accounting exec start-stop tacacs+**	Enables TACACS+ accounting to send a start-record accounting notice at the beginning of a privileged EXEC process and a stop-record at the end.
4.	**end**	Returns to privileged EXEC mode.

Configuring Radius Server Communication

Before You Begin

You should have access to and should configure a RADIUS server before configuring RADIUS features on your switch.

At a minimum, you must identify the host or hosts that run the RADIUS server software and define the method lists for RADIUS authentication. You can optionally define method lists for RADIUS authorization and accounting.

Some configuration settings need to be configured on the RADIUS server that include the IP address of the switch and the key string to be shared by both the server and the switch.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**radius-server host** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]	Specifies the IP address or hostname of the remote RADIUS server host. ■ (Optional) **auth-port** *port-number*–Specifies the UDP destination port for authentication requests. ■ (Optional) **acct-port** *port-number*–Specifies the UDP destination port for accounting requests. ■ (Optional) **timeout** *seconds*–Specifies the time interval that the switch waits for the RADIUS server to reply before resending. The range is 1 to 1000. This setting overrides the **radius-server timeout** global configuration command setting. If no timeout is set with the **radius-server host** command, the setting of the **radius-server timeout** command is used. ■ (Optional) **retransmit** *retries*–Specifies the number of times a RADIUS request is resent to a server if that server is not responding or responding slowly. The range is 1 to 1000. If no retransmit value is set with the **radius-server host** command, the setting of the **radius-server retransmit** global configuration command is used. ■ (Optional) **key** *string*–Specifies the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. **Note:** The key is a text string that must match the encryption key used on the RADIUS server. Always configure the key as the last item in the **radius-server host** command. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key. To configure the switch to recognize more than one host entry associated with a single IP address, enter this command as many times as necessary, making sure that each UDP port number is different. The switch software searches for hosts in the order in which you specify them. Set the timeout, retransmit, and encryption key values to use with the specific RADIUS host.
3.	**end**	Returns to privileged EXEC mode.

Defining AAA Server Groups

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**radius-server host** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]	Specifies the IP address or hostname of the remote RADIUS server host. ■ (Optional) **auth-port** *port-number*–Specifies the UDP destination port for authentication requests. ■ (Optional) **acct-port** *port-number*–Specifies the UDP destination port for accounting requests. ■ (Optional) **timeout** *seconds*–Specifies the time interval that the switch waits for the RADIUS server to reply before resending. The range is 1 to 1000. This setting overrides the **radius-server timeout** global configuration command setting. If no timeout is set with the **radius-server host** command, the setting of the **radius-server timeout** command is used. ■ (Optional) **retransmit** *retries*–Specifies the number of times a RADIUS request is resent to a server if that server is not responding or responding slowly. The range is 1 to 1000. If no retransmit value is set with the **radius-server host** command, the setting of the **radius-server retransmit** global configuration command is used. ■ (Optional) **key** *string*, specifies the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. **Note:** The key is a text string that must match the encryption key used on the RADIUS server. Always configure the key as the last item in the **radius-server host** command. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key. To configure the switch to recognize more than one host entry associated with a single IP address, enter this command as many times as necessary, making sure that each UDP port number is different. The switch software searches for hosts in the order in which you specify them. Set the timeout, retransmit, and encryption key values to use with the specific RADIUS host.
3.	**aaa new-model**	Enables AAA.
4.	**aaa group server radius** *group-name*	Defines the AAA server group with a group name. This command puts the switch in a server group configuration mode.
5.	**server** *ip-address*	Associates a particular RADIUS server with the defined server group. Repeat this step for each RADIUS server in the AAA server group. Each server in the group must be previously defined in Step 2.
6.	**end**	Returns to privileged EXEC mode.
7.		Enable RADIUS login authentication. See Defining AAA Server Groups, page 174.

Configuring RADIUS Login Authentication

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**aaa new-model**	Enables AAA.
3.	**aaa authentication login** {**default** | *list-name*} *method1* [*method2...*]	Creates a login authentication method list. ■ To create a default list that is used when a named list is *not* specified in the **login authentication** command, use the **default** keyword followed by the methods that are to be used in default situations. The default method list is automatically applied to all ports. ■ *list-name*–Specifies a character string to name the list you are creating. ■ *method1...*–Specifies the actual method the authentication algorithm tries. The additional methods of authentication are used only if the previous method returns an error, not if it fails. Select one of these methods: – **enable**–Uses the enable password for authentication. Before you can use this authentication method, you must define an enable password by using the **enable** *password* global configuration command. – **group radius**–Uses RADIUS authentication. Before you can use this authentication method, you must configure the RADIUS server. For more information, see RADIUS Server Host, page 155. – **line**–Uses the line password for authentication. Before you can use this authentication method, you must define a line password. Use the **password** *password* line configuration command. – **local**–Uses the local username database for authentication. You must enter username information in the database. Use the **username** *name* **password** global configuration command. – **local-case**–Uses a case-sensitive local username database for authentication. You must enter username information in the database by using the **username** *password* global configuration command. – **none**–Does not use any authentication for login.
4.	**line** [**console** | **tty** | **vty**] *line-number* [*ending-line-number*]	Enters line configuration mode, and configures the lines to which you want to apply the authentication list.
5.	**login authentication** {**default** | *list-name*}	Applies the authentication list to a line or set of lines. ■ If you specify **default**, use the default list created with the **aaa authentication login** command. ■ *list-name*–Specifies the list created with the **aaa authentication login** command.
6.	**end**	Returns to privileged EXEC mode.

Configuring RADIUS Authorization for User Privileged Access and Network Services

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**aaa authorization network radius**	Configures the switch for user RADIUS authorization for all network-related service requests.
3.	**aaa authorization exec radius**	Configures the switch for user RADIUS authorization if the user has privileged EXEC access. The **exec** keyword might return user profile information (such as **autocommand** information).
4.	**end**	Returns to privileged EXEC mode.

Starting RADIUS Accounting

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**aaa accounting network start-stop radius**	Enables RADIUS accounting for all network-related service requests.
3.	**aaa accounting exec start-stop radius**	Enables RADIUS accounting to send a start-record accounting notice at the beginning of a privileged EXEC process and a stop-record at the end.
4.	**end**	Returns to privileged EXEC mode.

Configuring Settings for All RADIUS Servers

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**radius-server key** *string*	Specifies the shared secret text string used between the switch and all RADIUS servers. **Note:** The key is a text string that must match the encryption key used on the RADIUS server. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key.
3.	**radius-server retransmit** *retries*	Specifies the number of times the switch sends each RADIUS request to the server before giving up. The default is 3; the range 1 to 1000.
4.	**radius-server timeout** *seconds*	Specifies the number of seconds a switch waits for a reply to a RADIUS request before resending the request. The default is 5 seconds; the range is 1 to 1000.

	Command	Purpose
5.	**radius-server deadtime** *minutes*	Specifies the number of minutes a RADIUS server, which is not responding to authentication requests, to be skipped, thus avoiding the wait for the request to timeout before trying the next configured server. The default is 0; the range is 1 to 1440 minutes.
6.	**radius-server vsa send** [**accounting** | **authentication**]	Enables the switch to recognize and use VSAs as defined by RADIUS IETF attribute 26. ■ (Optional) **accounting**–Limits the set of recognized vendor-specific attributes to only accounting attributes. ■ (Optional) **authentication**–Limits the set of recognized vendor-specific attributes to only authentication attributes. If you enter this command without keywords, both accounting and authentication vendor-specific attributes are used.
7.	**end**	Returns to privileged EXEC mode.

Configuring the Switch for Vendor-Proprietary RADIUS Server Communication

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**radius-server host** {*hostname* | *ip-address*} **non-standard**	Specifies the IP address or hostname of the remote RADIUS server host and identifies that it is using a vendor-proprietary implementation of RADIUS.
3.	**radius-server key** *string*	Specifies the shared secret text string used between the switch and the vendor-proprietary RADIUS server. The switch and the RADIUS server use this text string to encrypt passwords and exchange responses. **Note:** The key is a text string that must match the encryption key used on the RADIUS server. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key.
4.	**end**	Returns to privileged EXEC mode.
5.	**show running-config**	Verifies your settings.
6.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring CoA on the Switch

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**aaa new-model**	Enables AAA.
3.	**aaa server radius dynamic-author**	Configures the switch as an authentication, authorization, and accounting (AAA) server to facilitate interaction with an external policy server.

	Command	Purpose
4.	**client** {*ip-address* | *name*} [**vrf** *vrfname*] [**server-key** *string*]	Enters dynamic authorization local server configuration mode and specifies a RADIUS client from which a device will accept CoA and disconnect requests.
5.	**server-key** [**0** | **7**] *string*	Configures the RADIUS key to be shared between a device and RADIUS clients.
6.	**port** *port-number*	Specifies the port on which a device listens for RADIUS requests from configured RADIUS clients.
7.	**auth-type** {**any** | **all** | **session-key**}	Specifies the type of authorization the switch uses for RADIUS clients. The client must match all the configured attributes for authorization.
8.	**ignore session-key**	(Optional) Configures the switch to ignore the session-key.
9.	**ignore server-key**	(Optional) Configures the switch to ignore the server-key.
10.	**authentication command bounce-port ignore**	(Optional) Configures the switch to ignore a CoA request to temporarily disable the port hosting a session. The purpose of temporarily disabling the port is to trigger a DHCP renegotiation from the host when a VLAN change occurs and there is no supplicant on the endpoint to detect the change.
11.	**authentication command disable-port ignore**	(Optional) Configures the switch to ignore a nonstandard command requesting that the port hosting a session be administratively shut down. Shutting down the port results in termination of the session. Uses standard CLI or SNMP commands to reenable the port.
12.	**end**	Returns to privileged EXEC mode.

Configuring the Switch for Local Authentication and Authorization

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**aaa new-model**	Enables AAA.
3.	**aaa authentication login default local**	Sets the login authentication to use the local username database. The **default** keyword applies the local user database authentication to all ports.
4.	**aaa authorization exec local**	Configures user AAA authorization, checks the local database, and allows the user to run an EXEC shell.
5.	**aaa authorization network local**	Configures user AAA authorization for all network-related service requests.

	Command	Purpose
6.	**username** *name* [**privilege** *level*] {**password** *encryption-type* *password*}	Enters the local database, and establishes a username-based authentication system. Repeat this command for each user. ■ *name*–Specifies the user ID as one word. Spaces and quotation marks are not allowed. ■ (Optional) *level*–Specifies the privilege level the user has after gaining access. The range is 0 to 15. Level 15 gives privileged EXEC mode access. Level 0 gives user EXEC mode access. ■ *encryption-type*–Enters 0 to specify that an unencrypted password follows. Enter 7 to specify that a hidden password follows. ■ *password*–Specifies the password the user must enter to gain access to the switch. The password must be from 1 to 25 characters, can contain embedded spaces, and must be the last option specified in the **username** command.
7.	**end**	Returns to privileged EXEC mode.
8.	**show running-config**	Verifies your entries.
9.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Secure Shell

Setting Up the Switch to Run SSH

	Task	Purpose
1.	Download the cryptographic software image from Cisco.com.	(Required) For more information, see the notes for this release.
2.	Configure a hostname and IP domain name for the switch.	Follow this procedure only if you are configuring the switch as an SSH server.
3.	Generate an RSA key pair for the switch, which automatically enables SSH.	Follow this procedure only if you are configuring the switch as an SSH server.
4.	Configure user authentication for local or remote access.	(Required) For more information, see Configuring the Switch for Local Authentication and Authorization, page 178.

Configuring the SSH Server

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**hostname** *hostname*	Configures a hostname for your switch.
3.	**ip domain-name** *domain_name*	Configures a host domain for your switch.

	Command	Purpose
4.	**crypto key generate rsa**	Enables the SSH server for local and remote authentication on the switch and generates an RSA key pair. We recommend that a minimum modulus size of 1024 bits. When you generate RSA keys, you are prompted to enter a modulus length. A longer modulus length might be more secure, but it takes longer to generate and to use.
5.	**ip ssh version** [**1** | **2**]	(Optional) Configures the switch to run SSH Version 1 or SSH Version 2. ■ **1**—Configures the switch to run SSH Version 1. ■ **2**—Configures the switch to run SSH Version 2. If you do not enter this command or do not specify a keyword, the SSH server selects the latest SSH version supported by the SSH client. For example, if the SSH client supports SSHv1 and SSHv2, the SSH server selects SSHv2.
6.	**ip ssh** {**timeout** *seconds* | **authentication-retries** *number*}	Configures the SSH control parameters. ■ Specifies the time-out value in seconds; the default is 120 seconds. The range is 0 to 120 seconds. This parameter applies to the SSH negotiation phase. After the connection is established, the switch uses the default time-out values of the CLI-based sessions. By default, up to five simultaneous, encrypted SSH connections for multiple CLI-based sessions over the network are available (session 0 to session 4). After the execution shell starts, the CLI-based session time-out value returns to the default of 10 minutes. ■ Specifies the number of times that a client can reauthenticate to the server. The default is 3; the range is 0 to 5. Repeat this step when configuring both parameters.
7.	**line vty** *line_number* [*ending_line_number*] **transport input ssh**	(Optional) Configures the virtual terminal line settings. ■ Enters line configuration mode to configure the virtual terminal line settings. *line_number* and *ending_line_number* specifiy a pair of lines. The range is 0 to 15. ■ Specifies that the switch prevent non-SSH Telnet connections. This limits the router to only SSH connections.
8.	**end**	Returns to privileged EXEC mode.
9.	**show ip ssh** or **show ssh**	Shows the version and configuration information for your SSH server. Shows the status of the SSH server on the switch.

Configuring Secure HTTP Servers and Clients

Configuring a CA Trustpoint

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**hostname** *hostname*	Specifies the hostname of the switch (required only if you have not previously configured a hostname).
3.	**ip domain-name** *domain-name*	Specifies the IP domain name of the switch (required only if you have not previously configured an IP domain name).
4.	**crypto key generate rsa**	(Optional) Generates an RSA key pair. RSA key pairs are required before you can obtain a certificate for the switch. RSA key pairs are generated automatically. You can use this command to regenerate the keys, if needed.
5.	**crypto ca trustpoint** *name*	Specifies a local configuration name for the CA trustpoint and enter CA trustpoint configuration mode.
6.	**enrollment url** *url*	Specifies the URL to which the switch should send certificate requests.
7.	**enrollment http-proxy** *host-name port-number*	(Optional) Configures the switch to obtain certificates from the CA through an HTTP proxy server.
8.	**crl query** *url*	Configures the switch to request a certificate revocation list (CRL) to ensure that the certificate of the peer has not been revoked.
9.	**primary**	(Optional) Specifies that the trustpoint should be used as the primary (default) trustpoint for CA requests.
10.	**exit**	Exits CA trustpoint configuration mode and returns to global configuration mode.
11.	**crypto ca authentication** *name*	Authenticates the CA by getting the public key of the CA. Uses the same name used in Step 5.
12.	**crypto ca enroll** *name*	Obtains the certificate from the specified CA trustpoint. This command requests a signed certificate for each RSA key pair.
13.	**end**	Returns to privileged EXEC mode.
14.	**show crypto ca trustpoints**	Verifies the configuration.

Configuring the Secure HTTP Server

Before You Begin

If you are using a certificate authority for certification, you should use the previous procedure to configure the CA trustpoint on the switch before enabling the HTTP server. If you have not configured a CA trustpoint, a self-signed certificate is generated the first time that you enable the secure HTTP server. After you have configured the server, you can configure options (path, access list to apply, maximum number of connections, or timeout policy) that apply to both standard and secure HTTP servers.

	Command	Purpose

1.	**show ip http server status**	(Optional) Displays the status of the HTTP server to determine if the secure HTTP server feature is supported in the software. You should see one of these lines in the output:
		`HTTP secure server capability: Present`
		or
		`HTTP secure server capability: Not present`

2.	**configure terminal**	Enters global configuration mode.
3.	**ip http secure-server**	Enables the HTTPS server if it has been disabled. The HTTPS server is enabled by default.
4.	**ip http secure-port** *port-number*	(Optional) Specifies the port number to be used for the HTTPS server. The default port number is 443. Valid options are 443 or any number in the range 1025 to 65535.
5.	**ip http secure-ciphersuite** {[**3des-ede-cbc-sha**] [**rc4-128-md5**] [**rc4-128-sha**] [**des-cbc-sha**]}	(Optional) Specifies the CipherSuites (encryption algorithms) to be used for encryption over the HTTPS connection. If you do not have a reason to specify a particularly CipherSuite, you should allow the server and client to negotiate a CipherSuite that they both support. This is the default.
6.	**ip http secure-client-auth**	(Optional) Configures the HTTP server to request an X.509v3 certificate from the client for authentication during the connection process. The default is for the client to request a certificate from the server, but the server does not attempt to authenticate the client.
7.	**ip http secure-trustpoint** *name*	Specifies the CA trustpoint to use to get an X.509v3 security certificate and to authenticate the client certificate connection. **Note:** Use of this command assumes you have already configured a CA trustpoint according to the previous procedure.
8.	**ip http path** *path-name*	(Optional) Sets a base HTTP path for HTML files. The path specifies the location of the HTTP server files on the local system (usually located in system flash memory).
9.	**ip http access-class** *access-list-number*	(Optional) Specifies an access list to use to allow access to the HTTP server.
10.	**ip http max-connections** *value*	(Optional) Sets the maximum number of concurrent connections that are allowed to the HTTP server. The range is 1 to 16; the default value is 5.
11.	**ip http timeout-policy idle** *seconds* **life** *seconds* **requests** *value*	(Optional) Specifies how long a connection to the HTTP server can remain open under the defined circumstances: ■ **idle**–Specifies the maximum time period when no data is received or response data cannot be sent. The range is 1 to 600 seconds. The default is 180 seconds (3 minutes). ■ **life**–Specifies the maximum time period from the time that the connection is established. The range is 1 to 86400 seconds (24 hours). The default is 180 seconds. ■ **requests**–Specifies the maximum number of requests processed on a persistent connection. The maximum value is 86400. The default is 1.
12.	**end**	Returns to privileged EXEC mode.
13.	**show ip http server secure status**	Displays the status of the HTTP secure server to verify the configuration.

Configuring the Secure HTTP Client

Before You Begin

The standard HTTP client and secure HTTP client are always enabled. A certificate authority is required for secure HTTP client certification. This procedure assumes that you have previously configured a CA trustpoint on the switch. If a CA trustpoint is not configured and the remote HTTPS server requires client authentication, connections to the secure HTTP client fail.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**ip http client secure-trustpoint** *name*	(Optional) Specifies the CA trustpoint to be used if the remote HTTP server requests client authentication. Using this command assumes that you have already configured a CA trustpoint by using the previous procedure. The command is optional if client authentication is not needed or if a primary trustpoint has been configured.
3.	**ip http client secure-ciphersuite** {[**3des-ede-cbc-sha**] [**rc4-128-md5**] [**rc4-128-sha**] [**des-cbc-sha**]}	(Optional) Specifies the CipherSuites (encryption algorithms) to be used for encryption over the HTTPS connection. If you do not have a reason to specify a particular CipherSuite, you should allow the server and client to negotiate a CipherSuite that they both support. This is the default.
4.	**end**	Returns to privileged EXEC mode.
5.	**show ip http client secure status**	Displays the status of the HTTP secure server to verify the configuration.
6.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Monitoring and Maintaining Switch-Based Authentication

Command	Purpose
show running-config	Verifies your configured entries.
copy running-config startup-config	Saves your entries in the configuration file.
show tacacs	Displays the TACACS+ server statistics.
debug radius	Displays the information associated with RADIUS.
debug aaa coa	Displays the debug information for CoA processing.
debug cmdhd	Displays the debug information for the command handler.
show aaa attributes protocol radius	Displays the RADIUS attributes.
show ip ssh	Displays the version and configuration information for the SSH server.
show ssh	Displays the status of the SSH server.
show ip http client secure status	Displays the HTTP secure client configuration.
show ip http server secure status	Displays the HTTP secure server configuration.

Configuration Examples for Configuring Switch-Based Authentication

Changing the Enable Password: Example

This example shows how to change the enable password to *l1u2c3k4y5*. The password is not encrypted and provides access to level 15 (traditional privileged EXEC mode access):

```
Switch(config)# enable password l1u2c3k4y5
```

Configuring the Encrypted Password: Example

This example shows how to configure the encrypted password *$1$FaD0$Xyti5Rkls3LoyxzS8* for privilege level 2:

```
Switch(config)# enable secret level 2 5 $1$FaD0$Xyti5Rkls3LoyxzS8
```

Setting the Telnet Password for a Terminal Line: Example

This example shows how to set the Telnet password to *let45me67in89*:

```
Switch(config)# line vty 10
Switch(config-line)# password let45me67in89
```

Setting the Privilege Level for a Command: Example

This example shows how to set the **configure** command to privilege level 14 and define *SecretPswd14* as the password users must enter to use level 14 commands:

```
Switch(config)# privilege exec level 14 configure
Switch(config)# enable password level 14 SecretPswd14
```

Configuring the RADIUS Server: Examples

This example shows how to configure one RADIUS server to be used for authentication and another to be used for accounting:

```
Switch(config)# radius-server host 172.29.36.49 auth-port 1612 key rad1
Switch(config)# radius-server host 172.20.36.50 acct-port 1618 key rad2
```

This example shows how to configure *host1* as the RADIUS server and to use the default ports for both authentication and accounting:

```
Switch(config)# radius-server host host1
```

Defining AAA Server Groups: Example

In this example, the switch is configured to recognize two different RADIUS group servers (*group1* and *group2*). Group1 has two different host entries on the same RADIUS server configured for the same services. The second host entry acts as a fail-over backup to the first entry.

```
Switch(config)# radius-server host 172.20.0.1 auth-port 1000 acct-port 1001
Switch(config)# radius-server host 172.10.0.1 auth-port 1645 acct-port 1646
Switch(config)# aaa new-model
Switch(config)# aaa group server radius group1
Switch(config-sg-radius)# server 172.20.0.1 auth-port 1000 acct-port 1001
Switch(config-sg-radius)# exit
Switch(config)# aaa group server radius group2
Switch(config-sg-radius)# server 172.20.0.1 auth-port 2000 acct-port 2001
Switch(config-sg-radius)# exit
```

Configuring Vendor-Specific RADIUS Attributes: Examples

This example shows how to provide a user logging in from a switch with immediate access to privileged EXEC commands:

```
cisco-avpair= ”shell:priv-lvl=15“
```

This example shows how to specify an authorized VLAN in the RADIUS server database:

```
cisco-avpair= ”tunnel-type(#64)=VLAN(13)”
cisco-avpair= ”tunnel-medium-type(#65)=802 media(6)”
cisco-avpair= ”tunnel-private-group-id(#81)=vlanid”
```

This example shows how to apply an input ACL in ASCII format to an interface for the duration of this connection:

```
cisco-avpair= “ip:inacl#1=deny ip 10.10.10.10 0.0.255.255 20.20.20.20 255.255.0.0”
cisco-avpair= “ip:inacl#2=deny ip 10.10.10.10 0.0.255.255 any”
cisco-avpair= “mac:inacl#3=deny any any decnet-iv”
```

This example shows how to apply an output ACL in ASCII format to an interface for the duration of this connection:

```
cisco-avpair= “ip:outacl#2=deny ip 10.10.10.10 0.0.255.255 any”
```

Configuring a Vendor-Proprietary RADIUS Host: Example

This example shows how to specify a vendor-proprietary RADIUS host and to use a secret key of *rad124* between the switch and the server:

```
Switch(config)# radius-server host 172.20.30.15 nonstandard
Switch(config)# radius-server key rad124
```

Sample Output for a Self-Signed Certificate: Example

If a self-signed certificate has been generated, this information is included in the output of the **show running-config** privileged EXEC command. This is a partial sample output from that command displaying a self-signed certificate.

```
Switch# show running-config
Building configuration...

<output truncated>

crypto pki trustpoint TP-self-signed-3080755072
 enrollment selfsigned
 subject-name cn=IOS-Self-Signed-Certificate-3080755072
 revocation-check none
 rsakeypair TP-self-signed-3080755072
!
!
crypto ca certificate chain TP-self-signed-3080755072
 certificate self-signed 01
  3082029F 30820208 A0030201 02020101 300D0609 2A864886 F70D0101 04050030
  59312F30 2D060355 04031326 494F532D 53656C66 2D536967 6E65642D 43657274
  69666963 6174652D 33303830 37353530 37323126 30240609 2A864886 F70D0109
```

```
<output truncated>
```

You can remove this self-signed certificate by disabling the secure HTTP server and entering the **no crypto pki trustpoint TP-self-signed-30890755072** global configuration command. If you later reenable a secure HTTP server, a new self-signed certificate is generated.

Verifying Secure HTTP Connection: Example

To verify the secure HTTP connection by using a Web browser, enter https://*URL*, where the *URL* is the IP address or hostname of the server switch. If you configure a port other than the default port, you must also specify the port number after the URL. For example:

```
https://209.165.129:1026
```

or

```
https://host.domain.com:1026
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*
Secure Copy Protocol configuration	*Cisco IOS Security Configuration Guide: Securing User Services*
RADIUS Server Load Balancing configuration	*Cisco IOS Security Configuration Guide*
Kerberos configuration examples	*Cisco IOS Security Configuration Guide: Security Server Protocols*
Authenticating a network service	*Cisco IOS Security Configuration Guide: Security Server Protocols*
Authenticating for KDC	*Cisco IOS Security Configuration Guide: Security Server Protocols*
Kerberos configuration task list	*Cisco IOS Security Configuration Guide: Security Server Protocols*
Login enhancement configuration	Cisco IOS User Security Configuration Guide
Password protection commands	*Cisco IOS Security Command Reference*
Kerberos commands	*Cisco IOS Security Command Reference*
Secure Shell commands	*Cisco IOS Security Command Reference*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Configuring IEEE 802.1x Port-Based Authentication

Restrictions for Configuring IEEE 802.1x Port-Based Authentication

- To use this feature, the switch must be running the LAN Base image.

Information About Configuring IEEE 802.1x Port-Based Authentication

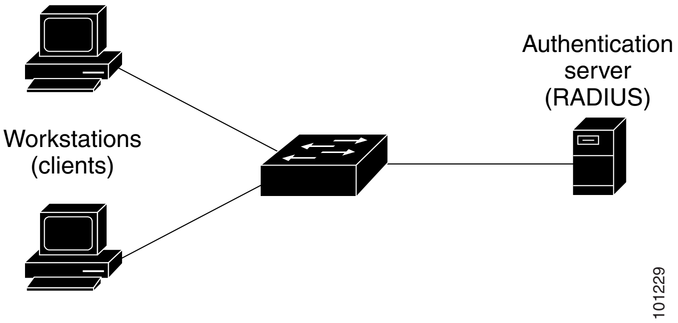
IEEE 802.1x Port-Based Authentication

The standard defines a client-server-based access control and authentication protocol to prevent unauthorized clients from connecting to a LAN through publicly accessible ports. The authentication server authenticates each client connected to a switch port before making available any switch or LAN services.

Until the client is authenticated, IEEE 802.1x access control allows only Extensible Authentication Protocol over LAN (EAPOL), Cisco Discovery Protocol (CDP), and Spanning Tree Protocol (STP) traffic through the port to which the client is connected. After authentication, normal traffic passes through the port.

Device Roles

Figure 18 802.1x Device Roles



- *Client*—The device (workstation) that requests access to the LAN and switch services and responds to requests from the switch. The workstation must be running 802.1x-compliant client software such as that offered in the Microsoft Windows XP operating system. (The client is the *supplicant* in the 802.1x standard.)

To resolve Windows XP network connectivity and 802.1x authentication issues, read the Microsoft Knowledge Base article:
http://support.microsoft.com/support/kb/articles/Q303/5/97.ASP

- *Authentication server*—Performs the actual authentication of the client. The authentication server validates the identity of the client and notifies the switch whether or not the client is authorized to access the LAN and switch services. Because the switch acts as the proxy, the authentication service is transparent to the client. In this release, the 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 or later. 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* (edge switch or wireless access point)—Controls the physical access to the network based on the authentication status of the client. The switch acts as an intermediary (proxy) between the client and the authentication server, requesting identity information from the client, verifying that information with the authentication server, and relaying a response to the client. The switch includes the RADIUS client, which is responsible for encapsulating and decapsulating the EAP frames and interacting with the authentication server. (The switch is the *authenticator* in the 802.1x standard.)

  When the switch receives EAPOL frames and relays them to the authentication server, the Ethernet header is stripped, and the remaining EAP frame is re-encapsulated in the RADIUS format. The EAP frames are not modified during encapsulation, and the authentication server must support EAP within the native frame format. When the switch receives 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 client.

  The devices that can act as intermediaries (switches, or a wireless access point) must be running software that supports the RADIUS client and 802.1x authentication.

Authentication Process

When 802.1x port-based authentication is enabled and the client supports 802.1x-compliant client software, these events occur:

- If the client identity is valid and the 802.1x authentication succeeds, the switch grants the client access to the network.

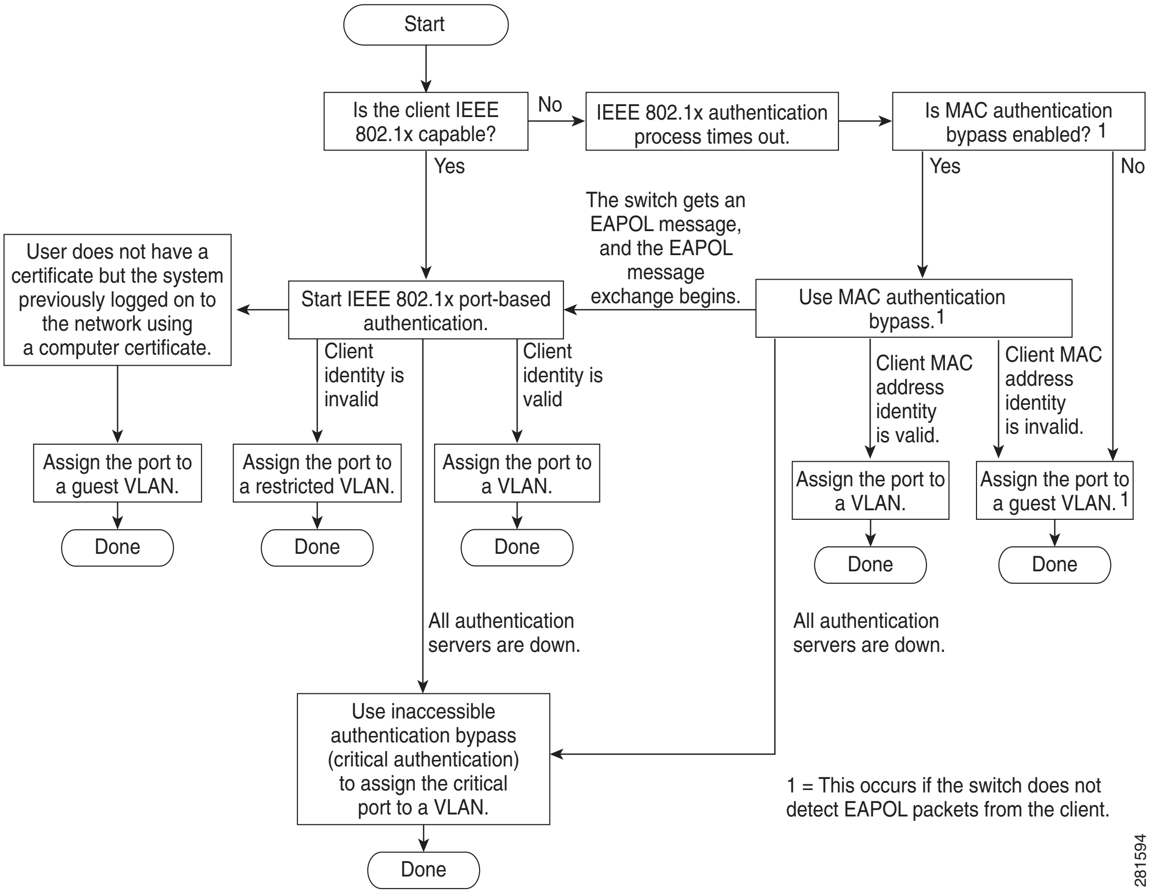
- If 802.1x authentication times out while waiting for an EAPOL message exchange and MAC authentication bypass is enabled, the switch can use the client MAC address for authorization. If the client MAC address is valid and the authorization succeeds, the switch grants the client access to the network. If the client MAC address is invalid and the authorization fails, the switch assigns the client to a guest VLAN that provides limited services if a guest VLAN is configured.

- If the switch gets an invalid identity from an 802.1x-capable client and a restricted VLAN is specified, the switch can assign the client to a restricted VLAN that provides limited services.

- If the RADIUS authentication server is unavailable (down) and inaccessible authentication bypass is enabled, the switch grants the client access to the network by putting the port in the critical-authentication state in the RADIUS-configured or the user-specified access VLAN.

  Note: Inaccessible authentication bypass is also referred to as critical authentication or the AAA fail policy.

Figure 19 Authentication Flowchart


Start
Is the client IEEE 802.1x capable?
No
IEEE 802.1x authentication process times out.
Is MAC authentication bypass enabled? 1
Yes
Yes
No
The switch gets an EAPOL message, and the EAPOL message exchange begins.
User does not have a certificate but the system previously logged on to the network using a computer certificate.
Start IEEE 802.1x port-based authentication.
Use MAC authentication bypass. 1
Client identity is invalid
Client identity is valid
Client MAC address identity is valid.
Client MAC address identity is invalid.
Assign the port to a guest VLAN.
Assign the port to a restricted VLAN.
Assign the port to a VLAN.
Assign the port to a VLAN.
Assign the port to a guest VLAN. 1
Done
Done
Done
Done
Done
All authentication servers are down.
All authentication servers are down.
Use inaccessible authentication bypass (critical authentication) to assign the critical port to a VLAN.
Done
1 = This occurs if the switch does not detect EAPOL packets from the client.
281594

The switch reauthenticates a client when one of these situations occurs:

- Periodic reauthentication is enabled, and the reauthentication timer expires.

  You can configure the reauthentication timer to use a switch-specific value or to be based on values from the RADIUS server.

  After 802.1x authentication using a RADIUS server is configured, the switch uses timers based on the Session-Timeout RADIUS attribute (Attribute[27]) and the Termination-Action RADIUS attribute (Attribute [29]).

  The Session-Timeout RADIUS attribute (Attribute[27]) specifies the time after which reauthentication occurs.

  The Termination-Action RADIUS attribute (Attribute [29]) specifies the action to take during reauthentication. The actions are *Initialize* and *ReAuthenticate*. When the *Initialize* action is set (the attribute value is *DEFAULT*), the 802.1x session ends, and connectivity is lost during reauthentication. When the *ReAuthenticate* action is set (the attribute value is RADIUS-Request), the session is not affected during reauthentication.

- You manually reauthenticate the client by entering the **dot1x re-authenticate interface** *interface-id* privileged EXEC command.

If multidomain authentication (MDA) is enabled on a port, this flow can be used with some exceptions that are applicable to voice authorization. For more information on MDA, see Multidomain Authentication, page 197.

Switch-to-RADIUS-Server Communication

RADIUS security servers are identified by their hostname or IP address, hostname and specific UDP port numbers, or IP address and specific UDP port numbers. The combination of the IP address and the UDP port number creates a unique identifier, which enables RADIUS requests to be sent to multiple UDP ports on a server at the same IP address. If two different host entries on the same RADIUS server are configured for the same service—for example, authentication—the second host entry configured acts as the failover backup to the first one. The RADIUS host entries are tried in the order in which they were configured.

Authentication Initiation and Message Exchange

During 802.1x authentication, the switch or the client can initiate authentication. If you enable authentication on a port by using the **authentication port-control auto** interface configuration command, the switch initiates authentication when the link state changes from down to up or periodically as long as the port remains up and unauthenticated. The switch sends an EAP-request/identity frame to the client to request its identity. Upon receipt of the frame, the client responds with an EAP-response/identity frame.

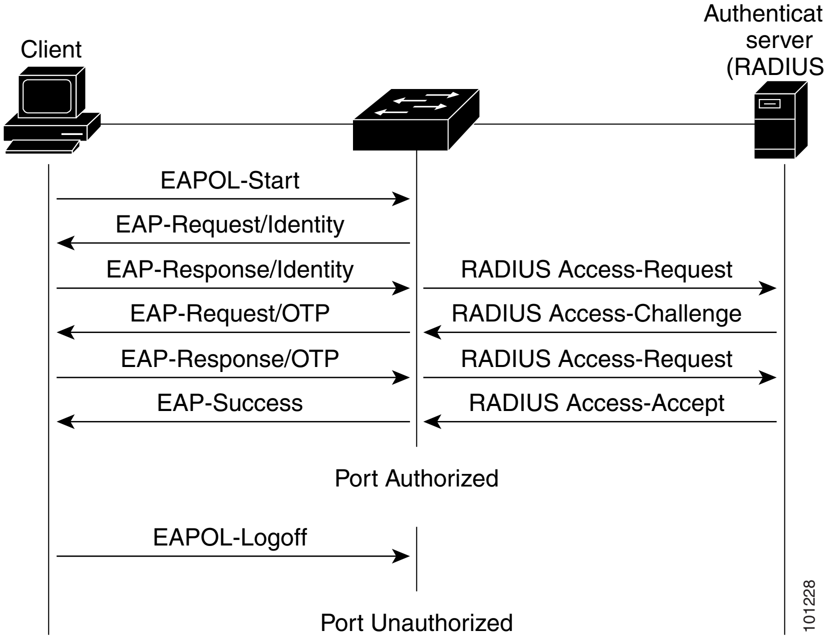
However, if during boot up, the client does not receive an EAP-request/identity frame from the switch, the client can initiate authentication by sending an EAPOL-start frame, which prompts the switch to request the client's identity.

Note: If 802.1x authentication is not enabled or supported on the network access device, any EAPOL frames from the client are dropped. If the client does not receive an EAP-request/identity frame after three attempts to start authentication, the client sends frames as if the port is in the authorized state. A port in the authorized state effectively means that the client has been successfully authenticated. For more information, see Ports in Authorized and Unauthorized States, page 195.

When the client supplies its identity, the switch begins its role as the intermediary, passing EAP frames between the client and the authentication server until authentication succeeds or fails. If the authentication succeeds, the switch port becomes authorized. If the authentication fails, authentication can be retried, the port might be assigned to a VLAN that provides limited services, or network access is not granted. For more information, see Ports in Authorized and Unauthorized States, page 195.

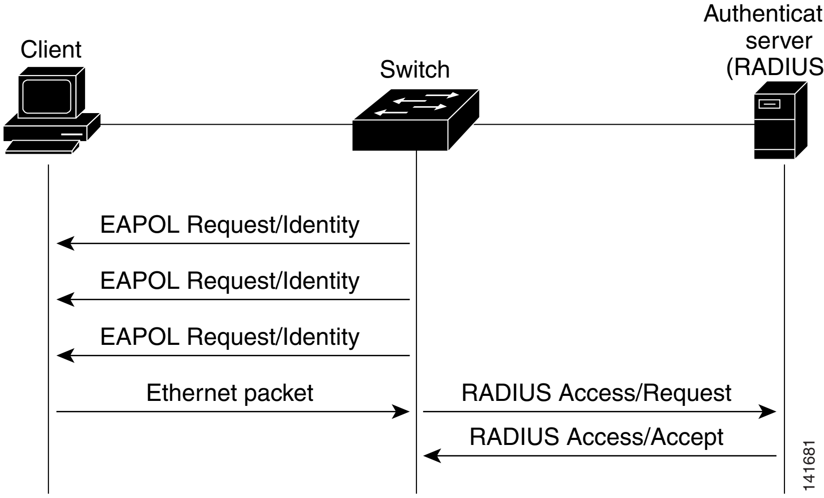
The specific exchange of EAP frames depends on the authentication method being used. Figure 20 on page 193 shows a message exchange initiated by the client when the client uses the One-Time-Password (OTP) authentication method with a RADIUS server.

Figure 20 Message Exchange



If 802.1x authentication times out while waiting for an EAPOL message exchange and MAC authentication bypass is enabled, the switch can authorize the client when the switch detects an Ethernet packet from the client. The switch uses the MAC address of the client as its identity and includes this information in the RADIUS-access/request frame that is sent to the RADIUS server. After the server sends the switch the RADIUS-access/accept frame (authorization is successful), the port becomes authorized. If authorization fails and a guest VLAN is specified, the switch assigns the port to the guest VLAN. If the switch detects an EAPOL packet while waiting for an Ethernet packet, the switch stops the MAC authentication bypass process and stops 802.1x authentication.

Figure 21 Message Exchange During MAC Authentication Bypass



Authentication Manager

Port-Based Authentication Methods

Table 29 on page 194 lists the authentication methods supported in these host modes:

- Single host–Only one data or voice host (client) can be authenticated on a port.
- Multiple host–Multiple data hosts can be authenticated on the same port. (If a port becomes unauthorized in multiple-host mode, the switch denies network access to all of the attached clients.)
- Multidomain authentication (MDA)–Both a data device and voice device can be authenticated on the same switch port. The port is divided into a data domain and a voice domain.
- Multiple authentication–Multiple hosts can authenticate on the data VLAN. This mode also allows one client on the VLAN if a voice VLAN is configured.

Table 29 802.1x Features

Authentication Method	Mode			
	Single Host	Multiple Host	MDA[1]	Multiple Authentication[2]
802.1x	VLAN assignment Per-user ACL Filter-ID attribute Downloadable ACL Redirect URL	VLAN assignment Per-user ACL Filter-ID attribute Downloadable ACL Redirect URL	VLAN assignment Per-user ACL Filter-Id attribute Downloadable ACL Redirect URL	Per-user ACL Filter-Id attribute Downloadable ACL Redirect URL
MAC authentication bypass	VLAN assignment Per-user ACL Filter-ID attribute Downloadable ACL Redirect URL	VLAN assignment Per-user ACL Filter-ID attribute Downloadable ACL Redirect URL	VLAN assignment Per-user ACL Filter-Id attribute Downloadable ACL Redirect URL	Per-user ACL Filter-Id attribute Downloadable ACL Redirect URL
Standalone web authentication	Proxy ACL, Filter-Id attribute, downloadable ACL[2]			
NAC Layer 2 IP validation	Filter-Id attribute Downloadable ACL Redirect URL	Filter-Id attribute Downloadable ACL Redirect URL	Filter-Id attribute Downloadable ACL Redirect URL	Filter-Id attribute[3] Downloadable ACL Redirect URL
Web authentication as fallback method[3]	Proxy ACL Filter-Id attribute Downloadable ACL	Proxy ACL Filter-Id attribute Downloadable ACL	Proxy ACL Filter-Id attribute Downloadable ACL	Proxy ACL[3] Filter-Id attribute Downloadable ACL

[1] MDA = Multidomain authentication.

[2] Also referred to as *multiauth*.

[3] For clients that do not support 802.1x authentication.

Per-User ACLs and Filter-Ids

Support was added for MDA- and multiauth-enabled ports. In 12.2(52)SE and later, support was added for ports in multihost mode.

An ACL configured on the switch is not compatible with an ACL configured on another device running Cisco IOS software, such as a Catalyst 6500 switch.

The ACLs configured on the switch are compatible with other devices running the Cisco IOS release.

Note: You can only set **any** as the source in the ACL.

Note: For any ACL configured for multiple-host mode, the source portion of statement must be *any*. (For example, permit icmp *any* host 10.10.1.1.)

You must specify *any* in the source ports of any defined ACL. Otherwise, the ACL cannot be applied and authorization fails. Single host is the only exception to support backward compatibility.

More than one host can be authenticated on MDA- enabled and multiauth ports. The ACL policy applied for one host does not effect the traffic of another host.

If only one host is authenticated on a multihost port, and the other hosts gain network access without authentication, the ACL policy for the first host can be applied to the other connected hosts by specifying *any* in the source address.

Authentication Manager CLI Commands

The authentication-manager interface-configuration commands control all the authentication methods, such as 802.1x, MAC authentication bypass, and web authentication. The authentication manager commands determine the priority and order of authentication methods applied to a connected host.

The authentication manager commands control generic authentication features, such as host-mode, violation mode, and the authentication timer. Generic authentication commands include the **authentication host-mode**, **authentication violation**, and **authentication timer** interface configuration commands.

802.1x-specific commands begin with the **dot1x** or **authentication** keyword. For example, the **authentication port-control auto** interface configuration command enables authentication on an interface. However, the **dot1x system-authentication control g**lobal configuration command only globally enables or disables 802.1x authentication.

Note: If 802.1x authentication is globally disabled, other authentication methods are still enabled on that port, such as web authentication.

You can filter out verbose system messages generated by the authentication manager. The filtered content typically relates to authentication success. You can also filter verbose messages for 802.1x authentication and MAB authentication. There is a separate command for each authentication method:

- The **no authentication logging verbose** global configuration command filters verbose messages from the authentication manager.
- The **no dot1x logging verbose** global configuration command filters 802.1x authentication verbose messages.
- The **no mab logging verbose** global configuration command filters MAC authentication bypass (MAB) verbose messages

Ports in Authorized and Unauthorized States

During 802.1x authentication, depending on the switch port state, the switch can grant a client access to the network. The port starts in the *unauthorized* state. While in this state, the port that is not configured as a voice VLAN port disallows all ingress and egress traffic except for 802.1x authentication, CDP, and STP packets. When a client is successfully authenticated, the port changes to the *authorized* state, allowing all traffic for the client to flow normally. If the port is configured as a voice VLAN port, the port allows VoIP traffic and 802.1x protocol packets before the client is successfully authenticated.

If a client that does not support 802.1x authentication connects to an unauthorized 802.1x port, the switch requests the client's identity. In this situation, the client does not respond to the request, the port remains in the unauthorized state, and the client is not granted access to the network.

In contrast, when an 802.1x-enabled client connects to a port that is not running the 802.1x standard, the client initiates the authentication process by sending the EAPOL-start frame. When no response is received, the client sends the request for a fixed number of times. Because no response is received, the client begins sending frames as if the port is in the authorized state.

You control the port authorization state by using the **authentication port-control** interface configuration command and these keywords:

- **force-authorized**—Disables 802.1x authentication and causes the port to change to the authorized state without any authentication exchange required. The port sends and receives normal traffic without 802.1x-based authentication of the client. This is the default setting.
- **force-unauthorized**—Causes the port to remain in the unauthorized state, ignoring all attempts by the client to authenticate. The switch cannot provide authentication services to the client through the port.
- **auto**—Enables 802.1x authentication and causes the port to begin in the unauthorized state, allowing only EAPOL frames to be sent and received through the port. The authentication process begins when the link state of the port changes from down to up or when an EAPOL-start frame is received. The switch requests the identity of the client and begins relaying authentication messages between the client and the authentication server. Each client attempting to access the network is uniquely identified by the switch by using the client MAC address.

If the client is successfully authenticated (receives an Accept frame from the authentication server), the port state changes to authorized, and all frames from the authenticated client are allowed through the port. If the authentication fails, the port remains in the unauthorized state, but authentication can be retried. If the authentication server cannot be reached, the switch can resend 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 client logs off, it sends an EAPOL-logoff message, causing the switch port to change to the unauthorized state.

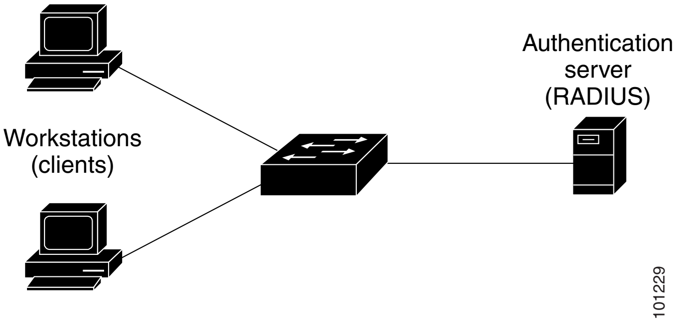
If the link state of a port changes from up to down, or if an EAPOL-logoff frame is received, the port returns to the unauthorized state.

802.1x Host Mode

You can configure an 802.1x port for single-host or for multiple-hosts mode. In single-host mode (see Figure 18 on page 189), only one client can be connected to the 802.1x-enabled switch port. The switch detects the client by sending an EAPOL frame when the port link state changes to the up state. If a client leaves or is replaced with another client, the switch changes the port link state to down, and the port returns to the unauthorized state.

In multiple-hosts mode, you can attach multiple hosts to a single 802.1x-enabled port. Figure 22 on page 197 shows 802.1x port-based authentication in a wireless LAN. In this mode, only one of the attached clients must be authorized for all clients to be granted network access. If the port becomes unauthorized (reauthentication fails or an EAPOL-logoff message is received), the switch denies network access to all of the attached clients. In this topology, the wireless access point is responsible for authenticating the clients attached to it, and it also acts as a client to the switch.

Figure 22 Multiple Host Mode Example



The switch supports multidomain authentication (MDA), which allows both a data device and a voice device, such as an IP Phone (Cisco or non-Cisco), to connect to the same switch port. For more information, see Multidomain Authentication, page 197.

Multidomain Authentication

The switch supports multidomain authentication (MDA), which allows both a data device and voice device, such as an IP phone (Cisco or non-Cisco), to authenticate on the same switch port. The port is divided into a data domain and a voice domain.

MDA does not enforce the order of device authentication. However, for best results, we recommend that a voice device is authenticated before a data device on an MDA-enabled port.

Follow these guidelines for configuring MDA:

- To configure a switch port for MDA, see Configuring the Host Mode, page 222.
- You must configure the voice VLAN for the IP phone when the host mode is set to multidomain. For more information, see Configuring VLANs, page 289
- To authorize a voice device, the AAA server must be configured to send a Cisco Attribute-Value (AV) pair attribute with a value of `device-traffic-class=voice`. Without this value, the switch treats the voice device as a data device.
- The guest VLAN and restricted VLAN features only apply to the data devices on an MDA-enabled port. The switch treats a voice device that fails authorization as a data device.
- If more than one device attempts authorization on either the voice or the data domain of a port, it is error disabled.
- Until a device is authorized, the port drops its traffic. Non-Cisco IP phones or voice devices are allowed into both the data and voice VLANs. The data VLAN allows the voice device to contact a DHCP server to obtain an IP address and acquire the voice VLAN information. After the voice device starts sending on the voice VLAN, its access to the data VLAN is blocked.
- A voice device MAC address that is binding on the data VLAN is not counted towards the port security MAC address limit.
- MDA can use MAC authentication bypass as a fallback mechanism to allow the switch port to connect to devices that do not support 802.1x authentication. For more information, see MAC Authentication Bypass Guidelines, page 217.
- When a *data* or a *voice* device is detected on a port, its MAC address is blocked until authorization succeeds. If the authorization fails, the MAC address remains blocked for 5 minutes.

- If more than five devices are detected on the *data* VLAN or more than one voice device is detected on the *voice* VLAN while a port is unauthorized, the port is error disabled.
- When a port host mode changes from single- or multihost to multidomain mode, an authorized data device remains authorized on the port. However, a Cisco IP phone on the port voice VLAN is automatically removed and must be reauthenticated on that port.
- Active fallback mechanisms such as guest VLAN and restricted VLAN remain configured after a port changes from single-host or multiple-host mode to multidomain mode.
- Switching a port host mode from multidomain to single-host or multiple-hosts mode removes all authorized devices from the port.
- If a data domain is authorized first and placed in the guest VLAN, non-802.1x-capable voice devices need their packets tagged on the voice VLAN to trigger authentication. The phone need not need to send tagged traffic. (The same is true for an 802.1x-capable phone.)
- We do not recommend per-user ACLs with an MDA-enabled port. An authorized device with a per-user ACL policy might impact traffic on both the port voice and data VLANs. You can use only one device on the port to enforce per-user ACLs.

For more information, see Configuring the Host Mode, page 222.

802.1x Multiple Authentication Mode

Multiple-authentication (multiauth) mode allows multiple authenticated clients on the data VLAN. Each host is individually authenticated. If a voice VLAN is configured, this mode also allows one client on the VLAN. (If the port detects any additional voice clients, they are discarded from the port, but no violation errors occur.)

If a hub or access point is connected to an 802.1x-enabled port, each connected client must be authenticated.

For non-802.1x devices, you can use MAC authentication bypass or web authentication as the per-host authentication fallback method to authenticate different hosts with different methods on a single port.

There is no limit to the number of data hosts can authenticate on a multiauthport. However, only one voice device is allowed if the voice VLAN is configured. Since there is no host limit defined violation will not be trigger, if a second voice is seen we silently discard it but do not trigger violation.

For MDA functionality on the voice VLAN, multiple-authentication mode assigns authenticated devices to either a data or a voice VLAN, depending on the VSAs received from the authentication server.

Note: When a port is in multiple-authentication mode, the guest VLAN and the authentication-failed VLAN features do not activate.

For more information about critical authentication mode and the critical VLAN, see 802.1x Authentication with Inaccessible Authentication Bypass, page 207.

For more information about configuring multiauth mode on a port, see Configuring the Host Mode, page 222.

MAC Move

When a MAC address is authenticated on one switch port, that address is not allowed on another authentication manager-enabled port of the switch. If the switch detects that same MAC address on another authentication manager-enabled port, the address is not allowed.

There are situations where a MAC address might need to move from one port to another on the same switch. For example, when there is another device (for example a hub or an IP phone) between an authenticated host and a switch port, you might want to disconnect the host from the device and connect it directly to another port on the same switch.

You can globally enable MAC move so the device is reauthenticated on the new port. When a host moves to a second port, the session on the first port is deleted, and the host is reauthenticated on the new port.

MAC move is supported on all host modes. (The authenticated host can move to any port on the switch, no matter which host mode is enabled on the that port.)

When a MAC address moves from one port to another, the switch terminates the authenticated session on the original port and initiates a new authentication sequence on the new port.

The MAC move feature applies to both voice and data hosts.

Note: In open authentication mode, a MAC address is immediately moved from the original port to the new port, with no requirement for authorization on the new port.

For more information see Configuring Optional 802.1x Authentication Features, page 224.

MAC Replace

The MAC replace feature can be configured to address the violation that occurs when a host attempts to connect to a port where another host was previously authenticated.

Note: This feature does not apply to ports in multiauth mode, because violations are not triggered in that mode. It does not apply to ports in multiple host mode, because in that mode, only the first host requires authentication.

If you configure the **authentication violation** interface configuration command with the **replace** keyword, the authentication process on a port in multidomain mode is:

- A new MAC address is received on a port with an existing authenticated MAC address.
- The authentication manager replaces the MAC address of the current data host on the port with the new MAC address.
- The authentication manager initiates the authentication process for the new MAC address.
- If the authentication manager determines that the new host is a voice host, the original voice host is removed.

If a port is in open authentication mode, any new MAC address is immediately added to the MAC address table.

For more information see Configuring Optional 802.1x Authentication Features, page 224.

802.1x Accounting

The 802.1x standard defines how users are authorized and authenticated for network access but does not keep track of network usage. 802.1x accounting is disabled by default. You can enable 802.1x accounting to monitor this activity on 802.1x-enabled ports:

- User successfully authenticates.
- User logs off.
- Link-down occurs.
- Reauthentication successfully occurs.
- Reauthentication fails.

The switch does not log 802.1x accounting information. Instead, it sends this information to the RADIUS server, which must be configured to log accounting messages.

802.1x Accounting Attribute-Value Pairs

The information sent to the RADIUS server is represented in the form of Attribute-Value (AV) pairs. These AV pairs provide data for different applications. (For example, a billing application might require information that is in the Acct-Input-Octets or the Acct-Output-Octets attributes of a RADIUS packet.)

AV pairs are automatically sent by a switch that is configured for 802.1x accounting. Three types of RADIUS accounting packets are sent by a switch:

- START—Sent when a new user session starts
- INTERIM—Sent during an existing session for updates
- STOP—Sent when a session terminates

Attribute Number	AV Pair Name	START	INTERIM	STOP
Attribute[1]	User-Name	Always	Always	Always
Attribute[4]	NAS-IP-Address	Always	Always	Always
Attribute[5]	NAS-Port	Always	Always	Always
Attribute[8]	Framed-IP-Address	Never	Sometimes[1]	Sometimes[1]
Attribute[25]	Class	Always	Always	Always
Attribute[30]	Called-Station-ID	Always	Always	Always
Attribute[31]	Calling-Station-ID	Always	Always	Always
Attribute[40]	Acct-Status-Type	Always	Always	Always
Attribute[41]	Acct-Delay-Time	Always	Always	Always
Attribute[42]	Acct-Input-Octets	Never	Always	Always
Attribute[43]	Acct-Output-Octets	Never	Always	Always
Attribute[44]	Acct-Session-ID	Always	Always	Always
Attribute[45]	Acct-Authentic	Always	Always	Always
Attribute[46]	Acct-Session-Time	Never	Always	Always
Attribute[49]	Acct-Terminate-Cause	Never	Never	Always
Attribute[61]	NAS-Port-Type	Always	Always	Always

[1] The Framed-IP-Address AV pair is sent only if a valid Dynamic Host Control Protocol (DHCP) binding exists for the host in the DHCP snooping bindings table.

You can view the AV pairs that are being sent by the switch by entering the **debug radius accounting** privileged EXEC command.

For more information about AV pairs, see RFC 3580, “802.1x Remote Authentication Dial In User Service (RADIUS) Usage Guidelines.”

802.1x Readiness Check

The 802.1x readiness check monitors 802.1x activity on all the switch ports and displays information about the devices connected to the ports that support 802.1x. You can use this feature to determine if the devices connected to the switch ports are 802.1x-capable. You use an alternate authentication such as MAC authentication bypass or web authentication for the devices that do not support 802.1x functionality.

This feature only works if the supplicant on the client supports a query with the NOTIFY EAP notification packet. The client must respond within the 802.1x timeout value.

Follow these guidelines to enable the readiness check on the switch:

- The readiness check is typically used before 802.1x is enabled on the switch.
- The 802.1x readiness check is allowed on all ports that can be configured for 802.1x. The readiness check is not available on a port that is configured as **dot1x force-unauthorized**.
- If you use the **dot1x test eapol-capable** privileged EXEC command without specifying an interface, all the ports on the switch stack are tested.
- When you configure the **dot1x test eapol-capable** command on an 802.1x-enabled port, and the link comes up, the port queries the connected client about its 802.1x capability. When the client responds with a notification packet, it is 802.1x-capable. A syslog message is generated if the client responds within the timeout period. If the client does not respond to the query, the client is not 802.1x-capable. No syslog message is generated.
- The readiness check can be sent on a port that handles multiple hosts (for example, a PC that is connected to an IP phone). A syslog message is generated for each of the clients that respond to the readiness check within the timer period.

For information on configuring the switch for the 802.1x readiness check, see Configuring 802.1x Readiness Check, page 220.

802.1x Authentication with VLAN Assignment

The RADIUS server sends the VLAN assignment to configure the switch port. The RADIUS server database maintains the username-to-VLAN mappings, assigning the VLAN based on the username of the client connected to the switch port. You can use this feature to limit network access for certain users.

When a voice device is authorized and the RADIUS server returns an authorized VLAN, the voice VLAN on the port is configured to send and receive packets on the assigned voice VLAN. Voice VLAN assignment behaves the same as data VLAN assignment on multidomain authentication (MDA)-enabled ports. For more information, see Multidomain Authentication, page 197.

When configured on the switch and the RADIUS server, 802.1x authentication with VLAN assignment has these characteristics:

- If no VLAN is supplied by the RADIUS server or if 802.1x authentication is disabled, the port is configured in its access VLAN after successful authentication. Recall that an access VLAN is a VLAN assigned to an access port. All packets sent from or received on this port belong to this VLAN.
- If 802.1x authentication is enabled but the VLAN information from the RADIUS server is not valid, authorization fails and configured VLAN remains in use. This prevents ports from appearing unexpectedly in an inappropriate VLAN because of a configuration error.

  Configuration errors could include specifying a VLAN for a routed port, a malformed VLAN ID, a nonexistent or internal (routed port) VLAN ID, an RSPAN VLAN, a shut down or suspended VLAN. In the case of a mutlidomain host port, configuration errors can also be due to an attempted assignment of a data VLAN that matches the configured or assigned voice VLAN ID (or the reverse).
- If 802.1x authentication is enabled and all information from the RADIUS server is valid, the authorized device is placed in the specified VLAN after authentication.
- If the multiple-hosts mode is enabled on an 802.1x port, all hosts are placed in the same VLAN (specified by the RADIUS server) as the first authenticated host.
- Enabling port security does not impact the RADIUS server-assigned VLAN behavior.
- If 802.1x authentication is disabled on the port, it is returned to the configured access VLAN and configured voice VLAN.

- If an 802.1x port is authenticated and put in the RADIUS server-assigned VLAN, any change to the port access VLAN configuration does not take effect. In the case of a multidomain host, the same applies to voice devices when the port is fully authorized with these exceptions:
  - If the VLAN configuration change of one device results in matching the other device configured or assigned VLAN, then authorization of all devices on the port is terminated and multidomain host mode is disabled until a valid configuration is restored where data and voice device configured VLANs no longer match.
  - If a voice device is authorized and is using a downloaded voice VLAN, the removal of the voice VLAN configuration, or modifying the configuration value to *dot1p* or *untagged* results in voice device un-authorization and the disablement of multi-domain host mode.

When the port is in the force authorized, force unauthorized, unauthorized, or shutdown state, it is put into the configured access VLAN.

The 802.1x authentication with VLAN assignment feature is not supported on trunk ports, dynamic ports, or with dynamic-access port assignment through a VLAN Membership Policy Server (VMPS).

To configure VLAN assignment you need to perform these tasks:

- Enable AAA authorization by using the **network** keyword to allow interface configuration from the RADIUS server.
- Enable 802.1x authentication. (The VLAN assignment feature is automatically enabled when you configure 802.1x authentication on an access port.)
- Assign vendor-specific tunnel attributes in the RADIUS server. The RADIUS server must return these attributes to the switch:
  - [64] Tunnel-Type = VLAN
  - [65] Tunnel-Medium-Type = 802
  - [81] Tunnel-Private-Group-ID = VLAN name, VLAN ID, or VLAN-Group
  - [83] Tunnel-Preference

  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* assigned to the 802.1x-authenticated user.

For examples of tunnel attributes, see Configuring Vendor-Specific RADIUS Attributes: Examples, page 185.

Voice Aware 802.1x Security

You use the voice aware 802.1x security feature on the switch to disable only the VLAN on which a security violation occurs, whether it is a data or voice VLAN. You can use this feature in IP phone deployments where a PC is connected to the IP phone. A security violation found on the data VLAN results in the shutdown of only the data VLAN. The traffic on the voice VLAN flows through the switch without interruption.

Follow these guidelines to configure voice aware 802.1x voice security on the switch:

- You enable voice aware 802.1x security by entering the **errdisable detect cause security-violation shutdown vlan** global configuration command. You disable voice aware 802.1x security by entering the **no** version of this command. This command applies to all 802.1x-configured ports in the switch.

Note: If you do not include the **shutdown vlan** keywords, the entire port is shut down when it enters the error-disabled state.

- If you use the **errdisable recovery cause security-violation** global configuration command to configure error-disabled recovery, the port is automatically reenabled. If error-disabled recovery is not configured for the port, you reenable it by using the **shutdown** and **no-shutdown** interface configuration commands.

- You can reenable individual VLANs by using the **clear errdisable interface** *interface-id* **vlan** [*vlan-list*] privileged EXEC command. If you do not specify a range, all VLANs on the port are enabled.

802.1x Authentication with Per-User ACLs

You can enable per-user access control lists (ACLs) to provide different levels of network access and service to an 802.1x-authenticated user. When the RADIUS server authenticates a user connected to an 802.1x port, it retrieves the ACL attributes based on the user identity and sends them to the switch. The switch applies the attributes to the 802.1x port for the duration of the user session. The switch removes the per-user ACL configuration when the session is over, if authentication fails, or if a link-down condition occurs. The switch does not save RADIUS-specified ACLs in the running configuration. When the port is unauthorized, the switch removes the ACL from the port.

You can configure router ACLs and input port ACLs on the same switch. However, a port ACL takes precedence over a router ACL. If you apply input port ACL to an interface that belongs to a VLAN, the port ACL takes precedence over an input router ACL applied to the VLAN interface. Incoming packets received on the port to which a port ACL is applied are filtered by the port ACL. Incoming routed packets received on other ports are filtered by the router ACL. Outgoing routed packets are filtered by the router ACL. To avoid configuration conflicts, you should carefully plan the user profiles stored on the RADIUS server.

RADIUS supports per-user attributes, including vendor-specific attributes. These vendor-specific attributes (VSAs) are in octet-string format and are passed to the switch during the authentication process. The VSAs used for per-user ACLs are `inacl#<`*n*`>` for the ingress direction and `outacl#<`*n*`>` for the egress direction. MAC ACLs are supported only in the ingress direction. The switch supports VSAs only in the ingress direction. It does not support port ACLs in the egress direction on Layer 2 ports. For more information, see Configuring Network Security with ACLs, page 575

Use only the extended ACL syntax style to define the per-user configuration stored on the RADIUS server. When the definitions are passed from the RADIUS server, they are created by using the extended naming convention. However, if you use the Filter-Id attribute, it can point to a standard ACL.

You can use the Filter-Id attribute to specify an inbound or outbound ACL that is already configured on the switch. The attribute contains the ACL number followed by *.in* for ingress filtering or *.out* for egress filtering. If the RADIUS server does not allow the *.in* or *.out* syntax, the access list is applied to the outbound ACL by default. Because of limited support of Cisco IOS access lists on the switch, the Filter-Id attribute is supported only for IP ACLs numbered 1 to 199 and 1300 to 2699 (IP standard and IP extended ACLs).

The maximum size of the per-user ACL is

4000 ASCII characters but is limited by the maximum size of RADIUS-server per-user ACLs.

For examples of vendor-specific attributes, see Configuring Vendor-Specific RADIUS Attributes: Examples, page 185. For more information about configuring ACLs, see Configuring Network Security with ACLs, page 575

Note: Per-user ACLs are supported only in single-host mode.

To configure per-user ACLs, you need to perform these tasks:

- Enable AAA authentication.
- Enable AAA authorization by using the **network** keyword to allow interface configuration from the RADIUS server.
- Enable 802.1x authentication.
- Configure the user profile and VSAs on the RADIUS server.
- Configure the 802.1x port for single-host mode.

For more configuration information, see Authentication Manager, page 194.

802.1x Authentication with Downloadable ACLs and Redirect URLs

You can download ACLs and redirect URLs from a RADIUS server to the switch during 802.1x authentication or MAC authentication bypass of the host. You can also download ACLs during web authentication.

Note: A downloadable ACL is also referred to as a *dACL*.

If more than one host is authenticated and the host is in single-host, MDA, or multiple-authentication mode, the switch changes the source address of the ACL to the host IP address.

You can apply the ACLs and redirect URLs to all the devices connected to the 802.1x-enabled port.

If no ACLs are downloaded during 802.1x authentication, the switch applies the static default ACL on the port to the host. On a voice VLAN port configured in multi-auth or MDA mode, the switch applies the ACL only to the phone as part of the authorization policies.

Note: The auth-default ACL does not appear in the running configuration.

The auth-default ACL is created when at least one host with an authorization policy is detected on the port. The auth-default ACL is removed from the port when the last authenticated session ends. You can configure the auth-default ACL by using the **ip access-list extended auth-default-acl** global configuration command.

Note: The auth-default ACL does not support Cisco Discovery Protocol (CDP) bypass in the single host mode. You must configure a static ACL on the interface to support CDP bypass.

The 802.1x and MAB authentication methods support two authentication modes, *open* and *closed*. If there is no static ACL on a port in *closed* authentication mode:

- An auth-default-ACL is created.
- The auth-default-ACL allows only DHCP traffic until policies are enforced.
- When the first host authenticates, the authorization policy is applied without IP address insertion.
- When a second host is detected, the policies for the first host are refreshed, and policies for the first and subsequent sessions are enforced with IP address insertion.

If there is no static ACL on a port in *open* authentication mode:

- An auth-default-ACL-OPEN is created and allows all traffic.
- Policies are enforced with IP address insertion to prevent security breaches.
- Web authentication is subject to the auth-default-ACL-OPEN.

To control access for hosts with no authorization policy, you can configure a directive. The supported values for the directive are *open* and *default*. When you configure the *open* directive, all traffic is allowed. The *default* directive subjects traffic to the access provided by the port. You can configure the directive either in the user profile on the AAA server or on the switch. To configure the directive on the AAA server, use the **authz-directive =** *open*/*default* global command. To configure the directive on the switch, use the **epm access-control open** global configuration command.

Note: The default value of the directive is *default*.

If a host falls back to web authentication on a port without a configured ACL:

- If the port is in open authentication mode, the auth-default-ACL-OPEN is created.
- If the port is in closed authentication mode, the auth-default-ACL is created.

The access control entries (ACEs) in the fallback ACL are converted to per-user entries. If the configured fallback profile does not include a fallback ACL, the host is subject to the auth-default-ACL associated with the port.

Note: If you use a custom logo with web authentication and it is stored on an external server, the port ACL must allow access to the external server before authentication. You must either configure a static port ACL or change the auth-default-ACL to provide appropriate access to the external server.

Cisco Secure ACS and Attribute-Value Pairs for the Redirect URL

The switch uses these *cisco-av-pair* VSAs:

- url-redirect is the HTTP to HTTPS URL.
- url-redirect-acl is the switch ACL name or number.

The switch uses the CiscoSecure-Defined-ACL attribute value pair to intercept an HTTP or HTTPS request from the end point device. The switch then forwards the client web browser to the specified redirect address. The url-redirect attribute value pair on the Cisco Secure ACS contains the URL to which the web browser is redirected. The url-redirect-acl attribute value pair contains the name or number of an ACL that specifies the HTTP or HTTPS traffic to redirect. Traffic that matches a permit ACE in the ACL is redirected.

Note: Define the URL redirect ACL and the default port ACL on the switch.

If a redirect URL is configured for a client on the authentication server, a default port ACL on the connected client switch port must also be configured.

Cisco Secure ACS and Attribute-Value Pairs for Downloadable ACLs

You can set the CiscoSecure-Defined-ACL Attribute-Value pair on the Cisco Secure ACS with the RADIUS cisco-av-pair vendor-specific attributes (VSAs). This pair specifies the names of the downloadable ACLs on the Cisco Secure ACS with the #ACL#-IP-*name*-*number* attribute.

- The *name* is the ACL name.
- The *number* is the version number (for example, 3f783768).

If a downloadable ACL is configured for a client on the authentication server, a default port ACL on the connected client switch port must also be configured.

If the default ACL is configured on the switch and the Cisco Secure ACS sends a host-access-policy to the switch, it applies the policy to traffic from the host connected to a switch port. If the policy does not apply, the switch applies the default ACL. If the Cisco Secure ACS sends the switch a downloadable ACL, this ACL takes precedence over the default ACL that is configured on the switch port. However, if the switch receives an host access policy from the Cisco Secure ACS but the default ACL is not configured, the authorization failure is declared.

For configuration details, see Authentication Manager, page 194 and Configuring 802.1x Authentication with Downloadable ACLs and Redirect URLs, page 231.

VLAN ID-Based MAC Authentication

You can use VLAN ID-based MAC authentication if you want to authenticate hosts based on a static VLAN ID instead of a downloadable VLAN. When you have a static VLAN policy configured on your switch, VLAN information is sent to an IAS (Microsoft) RADIUS server along with the MAC address of each host for authentication. The VLAN ID configured on the connected port is used for MAC authentication. By using VLAN ID-based MAC authentication with an IAS server, you can have a fixed number of VLANs in the network.

The feature also limits the number of VLANs monitored and handled by STP. The network can be managed as a fixed VLAN.

Note: This feature is not supported on Cisco ACS Server. (The ACS server ignores the sent VLAN-IDs for new hosts and only authenticates based on the MAC address.)

For configuration information, see Configuring Optional 802.1x Authentication Features, page 224. Additional configuration is similar MAC authentication bypass, as described in Configuring 802.1x User Distribution, page 229.

802.1x Authentication with Guest VLAN

You can configure a guest VLAN for each 802.1x port on the switch to provide limited services to clients, such as downloading the 802.1x client. These clients might be upgrading their system for 802.1x authentication, and some hosts, such as Windows 98 systems, might not be 802.1x-capable.

When you enable a guest VLAN on an 802.1x port, the switch assigns clients to a guest VLAN when the switch does not receive a response to its EAP request/identity frame or when EAPOL packets are not sent by the client. The port is automatically set to multi-host mode.

The switch maintains the EAPOL packet history. If an EAPOL packet is detected on the interface during the lifetime of the link, the switch determines that the device connected to that interface is an 802.1x-capable supplicant, and the interface does not change to the guest VLAN state. EAPOL history is cleared if the interface link status goes down. If no EAPOL packet is detected on the interface, the interface changes to the guest VLAN state.

If devices send EAPOL packets to the switch during the lifetime of the link, the switch no longer allows clients that fail authentication access to the guest VLAN.

If the switch is trying to authorize an 802.1x-capable voice device and the AAA server is unavailable, the authorization attempt fails, but the detection of the EAPOL packet is saved in the EAPOL history. When the AAA server becomes available, the switch authorizes the voice device. However, the switch no longer allows other devices access to the guest VLAN. To prevent this situation, use one of these command sequences:

- Enter the **authentication event no-response action authorize vlan** *vlan-id* interface configuration command to allow access to the guest VLAN.
- Enter the **shutdown** interface configuration command followed by the **no shutdown** interface configuration command to restart the port.

Note: If an EAPOL packet is detected after the interface has changed to the guest VLAN, the interface reverts to an unauthorized state, and 802.1x authentication restarts.

Any number of 802.1x-incapable clients are allowed access when the switch port is moved to the guest VLAN. If an 802.1x-capable client joins the same port on which the guest VLAN is configured, the port is put into the unauthorized state in the user-configured access VLAN, and authentication is restarted.

Guest VLANs are supported on 802.1x ports in single host, multiple host, or multi-domain modes.

You can configure any active VLAN except an RSPAN VLAN, a private VLAN, or a voice VLAN as an 802.1x guest VLAN. The guest VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.

The switch supports *MAC authentication bypass*. When MAC authentication bypass is enabled on an 802.1x port, the switch can authorize clients based on the client MAC address when 802.1x authentication times out while waiting for an EAPOL message exchange. After detecting a client on an 802.1x port, the switch waits for an Ethernet packet from the client. The switch sends the authentication server a RADIUS-access/request frame with a username and password based on the MAC address. If authorization succeeds, the switch grants the client access to the network. If authorization fails, the switch assigns the port to the guest VLAN if one is specified. For more information, see 802.1x Authentication with MAC Authentication Bypass, page 210.

For more information, see Configuring a Guest VLAN, page 226.

802.1x Authentication with Restricted VLAN

You can configure a restricted VLAN (also referred to as an *authentication failed VLAN*) for each 802.1x port on a switch to provide limited services to clients that cannot access the guest VLAN. These clients are 802.1x-compliant and cannot access another VLAN because they fail the authentication process. A restricted VLAN allows users without valid credentials in an authentication server (typically, visitors to an enterprise) to access a limited set of services. The administrator can control the services available to the restricted VLAN.

Note: You can configure a VLAN to be both the guest VLAN and the restricted VLAN if you want to provide the same services to both types of users.

Without this feature, the client attempts and fails authentication indefinitely, and the switch port remains in the spanning-tree blocking state. With this feature, you can configure the switch port to be in the restricted VLAN after a specified number of authentication attempts (the default value is 3 attempts).

The authenticator counts the failed authentication attempts for the client. When this count exceeds the configured maximum number of authentication attempts, the port moves to the restricted VLAN. The failed attempt count increments when the RADIUS server replies with either an *EAP failure* or an empty response without an EAP packet. When the port moves into the restricted VLAN, the failed attempt counter resets.

Users who fail authentication remain in the restricted VLAN until the next reauthentication attempt. A port in the restricted VLAN tries to reauthenticate at configured intervals (the default is 60 seconds). If reauthentication fails, the port remains in the restricted VLAN. If reauthentication is successful, the port moves either to the configured VLAN or to a VLAN sent by the RADIUS server. You can disable reauthentication. If you do this, the only way to restart the authentication process is for the port to receive a *link dow*n or *EAP logoff* event. We recommend that you keep reauthentication enabled if a client might connect through a hub. When a client disconnects from the hub, the port might not receive the *link down* or *EAP logoff* event.

After a port moves to the restricted VLAN, a simulated EAP success message is sent to the client. This prevents clients from indefinitely attempting authentication. Some clients (for example, devices running Windows XP) cannot implement DHCP without EAP success.

Restricted VLANs are supported only on 802.1x ports in single-host mode and on Layer 2 ports.

You can configure any active VLAN except an RSPAN VLAN, a primary private VLAN, or a voice VLAN as an 802.1x restricted VLAN. The restricted VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.

Other security features such as dynamic ARP inspection, DHCP snooping, and IP source guard can be configured independently on a restricted VLAN.

For more information, see Configuring a Restricted VLAN, page 226.

802.1x Authentication with Inaccessible Authentication Bypass

Use the inaccessible authentication bypass feature, also referred to as *critical authentication* or the *AAA fail policy,* when the switch cannot reach the configured RADIUS servers and new hosts cannot be authenticated. You can configure the switch to connect those hosts to *critical ports*.

When a new host tries to connect to the critical port, that host is moved to a user-specified access VLAN, the *critical VLAN*. The administrator gives limited authentication to the hosts.

When the switch tries to authenticate a host connected to a critical port, the switch checks the status of the configured RADIUS server. 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, which is a special case of the authentication state.

Support on Multiple-Authentication Ports

When a port is configured on any host mode and the AAA server is unavailable, the port is then configured to multi-host mode and moved to the critical VLAN. To support this inaccessible bypass on multiple-authentication (multiauth) ports, use the **authentication event server dead action reinitialize vlan** *vlan-id* command. When a new host tries to connect to the critical port, that port is reinitialized and all the connected hosts are moved to the user-specified access VLAN.

This command is supported on all host modess.

Authentication Results

The behavior of the inaccessible authentication bypass 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 puts the port in the critical-authentication state in the RADIUS-configured or user-specified 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 exchange times out, and the switch puts the critical port in the critical-authentication state during the next authentication attempt.

You can configure the critical port to reinitialize hosts and move them out of the critical VLAN when the RADIUS server is again available. When this is configured, all critical ports in the critical-authentication state are automatically reauthenticated. For more information, see Configuring Inaccessible Authentication Bypass, page 227.

Feature Interactions

Inaccessible authentication bypass interacts with these features:

- Guest VLAN—Inaccessible authentication bypass is compatible with guest VLAN. When a guest VLAN is enabled on 8021.x port, the features interact as follows:
  - If at least one RADIUS server is available, the switch assigns a client to a guest VLAN when the switch does not receive a response to its EAP request/identity frame or when EAPOL packets are not sent by the client.
  - If all the RADIUS servers are not available and the client is connected to a critical port, the switch authenticates the client and puts the critical port in the critical-authentication state in the RADIUS-configured or user-specified access VLAN.
  - If all the RADIUS servers are not available and the client is not connected to a critical port, the switch might not assign clients to the guest VLAN if one is configured.
  - If all the RADIUS servers are not available and if a client is connected to a critical port and was previously assigned to a guest VLAN, the switch keeps the port in the guest VLAN.
- Restricted VLAN—If the port is already authorized in a restricted VLAN and the RADIUS servers are unavailable, the switch puts the critical port in the critical-authentication state in the restricted VLAN.
- 802.1x accounting—Accounting is not affected if the RADIUS servers are unavailable.
- Private VLAN—You can configure inaccessible authentication bypass on a private VLAN host port. The access VLAN must be a secondary private VLAN.
- Voice VLAN—Inaccessible authentication bypass is compatible with voice VLAN, but the RADIUS-configured or user-specified access VLAN and the voice VLAN must be different.
- Remote Switched Port Analyzer (RSPAN)—Do not configure an RSPAN VLAN as the RADIUS-configured or user-specified access VLAN for inaccessible authentication bypass.

802.1x Authentication with Voice VLAN Ports

A voice VLAN port is a special access port associated with two VLAN identifiers:

- VVID to carry voice traffic to and from the IP phone. The VVID is used to configure the IP phone connected to the port.
- PVID to carry the data traffic to and from the workstation connected to the switch through the IP phone. The PVID is the native VLAN of the port.

The IP phone uses the VVID for its voice traffic, regardless of the authorization state of the port. This allows the phone to work independently of 802.1x authentication.

In single-host mode, only the IP phone is allowed on the voice VLAN. In multiple-hosts mode, additional clients can send traffic on the voice VLAN after a supplicant is authenticated on the PVID. When multiple-hosts mode is enabled, the supplicant authentication affects both the PVID and the VVID.

A voice VLAN port becomes active when there is a link, and the device MAC address appears after the first CDP message from the IP phone. Cisco IP phones do not relay CDP messages from other devices. As a result, if several IP phones are connected in series, the switch recognizes only the one directly connected to it. When 802.1x authentication is enabled on a voice VLAN port, the switch drops packets from unrecognized IP phones more than one hop away.

When 802.1x authentication is enabled on a port, you cannot configure a port VLAN that is equal to a voice VLAN.

Note: If you enable 802.1x authentication on an access port on which a voice VLAN is configured and to which a Cisco IP Phone is connected, the Cisco IP phone loses connectivity to the switch for up to 30 seconds.

For more information about voice VLANs, see Configuring Voice VLAN, page 327

802.1x Authentication with Port Security

In general, Cisco does not recommend enabling port security when IEEE 802.1x is enabled. Since IEEE 802.1x enforces a single MAC address per port (or per VLAN when MDA is configured for IP telephony), port security is redundant and in some cases may interfere with expected IEEE 802.1x operations.

802.1x Authentication with Wake-on-LAN

The 802.1x authentication with the wake-on-LAN (WoL) feature allows dormant PCs to be powered when the switch receives a specific Ethernet frame, known as the *magic packet*. You can use this feature in environments where administrators need to connect to systems that have been powered down.

When a host that uses WoL is attached through an 802.1x port and the host powers off, the 802.1x port becomes unauthorized. The port can only receive and send EAPOL packets, and WoL magic packets cannot reach the host. When the PC is powered off, it is not authorized, and the switch port is not opened.

When the switch uses 802.1x authentication with WoL, the switch forwards traffic to unauthorized 802.1x ports, including magic packets. While the port is unauthorized, the switch continues to block ingress traffic other than EAPOL packets. The host can receive packets but cannot send packets to other devices in the network.

Note: If PortFast is not enabled on the port, the port is forced to the bidirectional state.

When you configure a port as unidirectional by using the **authentication control-direction in** interface configuration command, the port changes to the spanning-tree forwarding state. The port can send packets to the host but cannot receive packets from the host.

When you configure a port as bidirectional by using the **authentication control-direction both** interface configuration command, the port is access-controlled in both directions. The port does not receive packets from or send packets to the host.

802.1x Authentication with MAC Authentication Bypass

You can configure the switch to authorize clients based on the client MAC address (see Figure 19 on page 191) by using the MAC authentication bypass feature. For example, you can enable this feature on 802.1x ports connected to devices such as printers.

If 802.1x authentication times out while waiting for an EAPOL response from the client, the switch tries to authorize the client by using MAC authentication bypass.

When the MAC authentication bypass feature is enabled on an 802.1x port, the switch uses the MAC address as the client identity. The authentication server has a database of client MAC addresses that are allowed network access. After detecting a client on an 802.1x port, the switch waits for an Ethernet packet from the client. The switch sends the authentication server a RADIUS-access/request frame with a username and password based on the MAC address. If authorization succeeds, the switch grants the client access to the network. If authorization fails, the switch assigns the port to the guest VLAN if one is configured.

If an EAPOL packet is detected on the interface during the lifetime of the link, the switch determines that the device connected to that interface is an 802.1x-capable supplicant and uses 802.1x authentication (not MAC authentication bypass) to authorize the interface. EAPOL history is cleared if the interface link status goes down.

If the switch already authorized a port by using MAC authentication bypass and detects an 802.1x supplicant, the switch does not unauthorize the client connected to the port. When reauthentication occurs, the switch uses 802.1x authentication as the preferred reauthentication process if the previous session ended because the Termination-Action RADIUS attribute value is DEFAULT.

Clients that were authorized with MAC authentication bypass can be reauthenticated. The reauthentication process is the same as that for clients that were authenticated with 802.1x. During reauthentication, the port remains in the previously assigned VLAN. If reauthentication is successful, the switch keeps the port in the same VLAN. If reauthentication fails, the switch assigns the port to the guest VLAN, if one is configured.

If reauthentication is based on the Session-Timeout RADIUS attribute (Attribute[27]) and the Termination-Action RADIUS attribute (Attribute [29]) and if the Termination-Action RADIUS attribute (Attribute [29]) action is *Initialize,* (the attribute value is *DEFAULT*), the MAC authentication bypass session ends, and connectivity is lost during reauthentication. If MAC authentication bypass is enabled and the 802.1x authentication times out, the switch uses the MAC authentication bypass feature to initiate reauthorization. For more information about these AV pairs, see RFC 3580, “802.1X Remote Authentication Dial In User Service (RADIUS) Usage Guidelines.”

MAC authentication bypass interacts with the features:

- 802.1x authentication—You can enable MAC authentication bypass only if 802.1x authentication is enabled on the port.
- Guest VLAN—If a client has an invalid MAC address identity, the switch assigns the client to a guest VLAN if one is configured.
- Restricted VLAN—This feature is not supported when the client connected to an 802.lx port is authenticated with MAC authentication bypass.
- Port security—See 802.1x Authentication with Port Security, page 209.
- Voice VLAN—See 802.1x Authentication with Voice VLAN Ports, page 209.
- VLAN Membership Policy Server (VMPS)—802.1x and VMPS are mutually exclusive.
- Private VLAN—You can assign a client to a private VLAN.
- Network admission control (NAC) Layer 2 IP validation—This feature takes effect after an 802.1x port is authenticated with MAC authentication bypass, including hosts in the exception list.
- Network Edge Access Topology (NEAT)—MAB and NEAT are mutually exclusive. You cannot enable MAB when NEAT is enabled on an interface, and you cannot enable NEAT when MAB is enabled on an interface.

For more configuration information, see Authentication Manager, page 194.

Cisco IOS Release 12.2(55)SE and later supports filtering of verbose MAB system messages. See Authentication Manager CLI Commands, page 195.

802.1x User Distribution

You can configure 802.1x user distribution to load-balance users with the same group name across multiple different VLANs.

The VLANs are either supplied by the RADIUS server or configured through the switch CLI under a VLAN group name.

- 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. You can search for the selected VLAN group name among the VLAN group names that you configured by using the switch CLI. If the VLAN group name is found, the corresponding VLANs under this VLAN group name are searched to find the least populated VLAN. Load balancing is achieved by moving the corresponding authorized user to that VLAN.

  Note: The RADIUS server can send the VLAN information in any combination of VLAN-IDs, VLAN names, or VLAN groups.

802.1x User Distribution Configuration Guidelines

- Confirm 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 authenticated ports 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 cleared.
- 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.

For more information, see Configuring 802.1x User Distribution, page 229.

Network Admission Control Layer 2 802.1x Validation

The switch supports the Network Admission Control (NAC) Layer 2 802.1x validation, which checks the antivirus condition or *posture* of endpoint systems or clients before granting the devices network access. With NAC Layer 2 802.1x validation, you can do these tasks:

- Download the Session-Timeout RADIUS attribute (Attribute[27]) and the Termination-Action RADIUS attribute (Attribute[29]) from the authentication server.
- Set the number of seconds between reauthentication attempts as the value of the Session-Timeout RADIUS attribute (Attribute[27]) and get an access policy against the client from the RADIUS server.

- Set the action to be taken when the switch tries to reauthenticate the client by using the Termination-Action RADIUS attribute (Attribute[29]). If the value is the *DEFAULT* or is not set, the session ends. If the value is RADIUS-Request, the reauthentication process starts.
- Set the list of VLAN number or name or VLAN group name as the value of the Tunnel Group Private ID (Attribute[81]) and the preference for the VLAN number or name or VLAN group name as the value of the Tunnel Preference (Attribute[83]). If you do not configure the Tunnel Preference, the first Tunnel Group Private ID (Attribute[81]) attribute is picked up from the list.
- View the NAC posture token, which shows the posture of the client, by using the **show authentication** privileged EXEC command.
- Configure secondary private VLANs as guest VLANs.

Configuring NAC Layer 2 802.1x validation is similar to configuring 802.1x port-based authentication except that you must configure a posture token on the RADIUS server. For information about configuring NAC Layer 2 802.1x validation, see Configuring NAC Layer 2 802.1x Validation, page 229 and the Configuring Periodic Reauthentication, page 223.

For more information about NAC, see the *Network Admission Control Software Configuration Guide*.

For more configuration information, see Authentication Manager, page 194.

Flexible Authentication Ordering

You can use flexible authentication ordering to configure the order of methods that a port uses to authenticate a new host. MAC authentication bypass and 802.1x can be the primary or secondary authentication methods, and web authentication can be the fallback method if either or both of those authentication attempts fail. For the configuration commands, see Configuring Optional 802.1x Authentication Features, page 224

Open1x Authentication

Open1x authentication allows a device access to a port before that device is authenticated. When open authentication is configured, a new host can pass traffic according to the access control list (ACL) defined on the port. After the host is authenticated, the policies configured on the RADIUS server are applied to that host.

You can configure open authentication with these scenarios:

- Single-host mode with open authentication—Only one user is allowed network access before and after authentication.
- MDA mode with open authentication—Only one user in the voice domain and one user in the data domain are allowed.
- Multiple-hosts mode with open authentication—Any host can access the network.
- Multiple-authentication mode with open authentication—Similar to MDA, except multiple hosts can be authenticated.

For more information see Configuring the Host Mode, page 222.

Note: If open authentication is configured, it takes precedence over other authentication controls. This means that if you use the **authentication open** interface configuration command, the port will grant access to the host irrespective of the **authentication port-control** interface configuration command.

802.1x Supplicant and Authenticator Switches with Network Edge Access Topology (NEAT)

The Network Edge Access Topology (NEAT) feature extends identity to areas outside the wiring closet (such as conference rooms). This allows any type of device to authenticate on the port.

- You can configure a switch to act as a supplicant to another switch by using the 802.1x supplicant feature. This configuration is helpful in a scenario, where, for example, a switch is outside a wiring closet and is connected to an upstream switch through a trunk port. A switch configured with the 802.1x switch supplicant feature authenticates with the upstream switch for secure connectivity.

  Once the supplicant switch authenticates successfully the port mode changes from access to trunk.

- If the access VLAN is configured on the authenticator switch, it becomes the native VLAN for the trunk port after successful authentication.

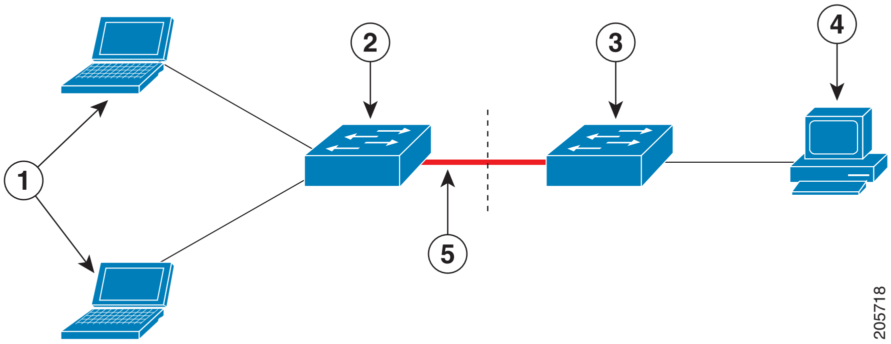
You can enable MDA or multiauth mode on the authenticator switch interface that connects to one more supplicant switches. Multihost mode is not supported on the authenticator switch interface.

Use the **dot1x supplicant force-multicast** global configuration command on the supplicant switch for Network Edge Access Topology (NEAT) to work in all host modes.

- Host authorization ensures that only traffic from authorized hosts (connecting to the switch with supplicant) is allowed on the network. The switches use Client Information Signalling Protocol (CISP) to send the MAC addresses connecting to the supplicant switch to the authenticator switch, as shown in Figure 23 on page 213.

- Auto enablement automatically enables trunk configuration on the authenticator switch, allowing user traffic from multiple VLANs coming from supplicant switches. Configure the cisco-av-pair as *device-traffic-class=switch* at the ACS. (You can configure this under the *group* or the *user* settings.)

Figure 23 Authenticator and Supplicant Switch using CISP



1	Workstations (clients)	2	Supplicant switch (outside wiring closet)
3	Authenticator switch	4	Access control server (ACS)
5	Trunk port		

802.1x Supplicant and Authenticator Switch Guidelines

- You can configure NEAT ports with the same configurations as the other authentication ports. When the supplicant switch authenticates, the port mode is changed from *access* to *trunk* based on the switch vendor-specific attributes (VSAs). (device-traffic-class=switch)

- The VSA changes the authenticator switch port mode from access to trunk and enables 802.1x trunk encapsulation and the access VLAN if any would be converted to a native trunk VLAN. VSA does not change any of the port configurations on the supplicant

- To change the host mode *and* to apply a standard port configuration on the authenticator switch port, you can also use Auto Smartports user-defined macros, instead of the switch VSA. This allows you to remove unsupported configurations on the authenticator switch port and to change the port mode from *access* to *trunk*. For information, see the *AutoSmartports Configuration Guide*.

For more information, see Configuring an Authenticator, page 230.

Using IEEE 802.1x Authentication with ACLs and the RADIUS Filter-Id Attribute

The switch supports both IP standard and IP extended port access control lists (ACLs) applied to ingress ports.

- ACLs that you configure
- ACLs from the Access Control Server (ACS)

An IEEE 802.1x port in single-host mode uses ACLs from the ACS to provide different levels of service to an IEEE 802.1x-authenticated user. When the RADIUS server authenticates this type of user and port, it sends ACL attributes based on the user identity to the switch. The switch applies the attributes to the port for the duration of the user session. If the session is over, authentication fails, or a link fails, the port becomes unauthorized, and the switch removes the ACL from the port.

Only IP standard and IP extended port ACLs from the ACS support the Filter-Id attribute. It specifies the name or number of an ACL. The Filter-id attribute can also specify the direction (inbound or outbound) and a user or a group to which the user belongs.

- The Filter-Id attribute for the user takes precedence over that for the group.
- If a Filter-Id attribute from the ACS specifies an ACL that is already configured, it takes precedence over a user-configured ACL.
- If the RADIUS server sends more than one Filter-Id attribute, only the last attribute is applied.

If the Filter-Id attribute is not defined on the switch, authentication fails, and the port returns to the unauthorized state.

Authentication Manager Common Session ID

Authentication manager uses a single session ID (referred to as a common session ID) for a client no matter which authentication method is used. This ID is used for all reporting purposes, such as the **show** commands and MIBs. The session ID appears with all per-session syslog messages.

The session ID includes:

- The IP address of the Network Access Device (NAD)
- A monotonically increasing unique 32-bit integer
- The session start time stamp (a 32-bit integer)

Default 802.1x Authentication Settings

Table 30 on page 214 shows the default 802.1x authentication settings.

Table 30 Default 802.1x Authentication Settings

Feature	Default Setting
Switch 802.1x enable state	Disabled.
Per-port 802.1x enable state	Disabled (force-authorized). The port sends and receives normal traffic without 802.1x-based authentication of the client.
AAA	Disabled.

Table 30 Default 802.1x Authentication Settings (continued)

Feature	Default Setting
RADIUS server ■ IP address ■ UDP authentication port ■ Key	 ■ None specified. ■ 1812. ■ None specified.
Host mode	Single-host mode.
Control direction	Bidirectional control.
Periodic reauthentication	Disabled.
Number of seconds between reauthentication attempts	3600 seconds.
Reauthentication number	2 times (number of times that the switch restarts the authentication process before the port changes to the unauthorized state).
Quiet period	60 seconds (number of seconds that the switch remains in the quiet state following a failed authentication exchange with the client).
Retransmission time	30 seconds (number of seconds that the switch should wait for a response to an EAP request/identity frame from the client before resending the request).
Maximum retransmission number	2 times (number of times that the switch will send an EAP-request/identity frame before restarting the authentication process).
Client timeout period	30 seconds (when relaying a request from the authentication server to the client, the amount of time the switch waits for a response before resending the request to the client.)
Authentication server timeout period	30 seconds (when relaying a response from the client to the authentication server, the amount of time the switch waits for a reply before resending the response to the server.) You can change this timeout period by using the **authentication timer server** interface configuration command.
Inactivity timeout	Disabled.
Guest VLAN	None specified.
Inaccessible authentication bypass	Disabled.
Restricted VLAN	None specified.
Authenticator (switch) mode	None specified.
MAC authentication bypass	Disabled.
Voice-aware security	Disabled

802.1x Accounting

Enabling AAA system accounting with 802.1x accounting allows system reload events to be sent to the accounting RADIUS server for logging. The server can then infer that all active 802.1x sessions are closed.

Because RADIUS uses the unreliable UDP transport protocol, accounting messages might be lost due to poor network conditions. If the switch does not receive the accounting response message from the RADIUS server after a configurable number of retransmissions of an accounting request, this system message appears:

```
Accounting message %s for session %s failed to receive Accounting Response.
```

When the stop message is not sent successfully, this message appears:

```
00:09:55: %RADIUS-4-RADIUS_DEAD: RADIUS server 172.20.246.201:1645,1646 is not responding.
```

Note: You must configure the RADIUS server to perform accounting tasks, such as logging start, stop, and interim-update messages and time stamps. To turn on these functions, enable logging of “Update/Watchdog packets from this AAA client” in your RADIUS server Network Configuration tab. Next, enable “CVS RADIUS Accounting” in your RADIUS server System Configuration tab.

802.1x Authentication Guidelines

- When 802.1x authentication is enabled, ports are authenticated before any other Layer 2 features are enabled.
- If the VLAN to which an 802.1x-enabled port is assigned changes, this change is transparent and does not affect the switch. For example, this change occurs if a port is assigned to a RADIUS server-assigned VLAN and is then assigned to a different VLAN after reauthentication.

  If the VLAN to which an 802.1x port is assigned to shut down, disabled, or removed, the port becomes unauthorized. For example, the port is unauthorized after the access VLAN to which a port is assigned shuts down or is removed.
- The 802.1x protocol is supported on Layer 2 static-access ports, and voice VLAN ports, but it is not supported on these port types:
  - Trunk port—If you try to enable 802.1x authentication on a trunk port, an error message appears, and 802.1x authentication is not enabled. If you try to change the mode of an 802.1x-enabled port to trunk, an error message appears, and the port mode is not changed.
  - Dynamic ports—A port in dynamic mode can negotiate with its neighbor to become a trunk port. If you try to enable 802.1x authentication on a dynamic port, an error message appears, and 802.1x authentication is not enabled. If you try to change the mode of an 802.1x-enabled port to dynamic, an error message appears, and the port mode is not changed.
  - Dynamic-access ports—If you try to enable 802.1x authentication on a dynamic-access (VLAN Query Protocol [VQP]) port, an error message appears, and 802.1x authentication is not enabled. If you try to change an 802.1x-enabled port to dynamic VLAN assignment, an error message appears, and the VLAN configuration is not changed.
  - EtherChannel port—Do not configure a port that is an active or a not-yet-active member of an EtherChannel as an 802.1x port. If you try to enable 802.1x authentication on an EtherChannel port, an error message appears, and 802.1x authentication is not enabled.
  - Switched Port Analyzer (SPAN) and Remote SPAN (RSPAN) destination ports—You can enable 802.1x authentication on a port that is a SPAN or RSPAN destination port. However, 802.1x authentication is disabled until the port is removed as a SPAN or RSPAN destination port. You can enable 802.1x authentication on a SPAN or RSPAN source port.
- Before globally enabling 802.1x authentication on a switch by entering the **dot1x system-auth-control** global configuration command, remove the EtherChannel configuration from the interfaces on which 802.1x authentication and EtherChannel are configured.
- System messages related to 802.1x authentication can be filtered. See Authentication Manager CLI Commands, page 195.

VLAN Assignment, Guest VLAN, Restricted VLAN, and Inaccessible Authentication Bypass Guidelines

- When 802.1x authentication is enabled on a port, you cannot configure a port VLAN that is equal to a voice VLAN.

- The 802.1x authentication with VLAN assignment feature is not supported on trunk ports, dynamic ports, or with dynamic-access port assignment through a VMPS.
- You can configure 802.1x authentication on a private-VLAN port, but do not configure 802.1x authentication with port security, a voice VLAN, a guest VLAN, a restricted VLAN, or a per-user ACL on private-VLAN ports.
- You can configure any VLAN except an RSPAN VLAN, private VLAN, or a voice VLAN as an 802.1x guest VLAN. The guest VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.
- After you configure a guest VLAN for an 802.1x port to which a DHCP client is connected, you might need to get a host IP address from a DHCP server. You can change the settings for restarting the 802.1x authentication process on the switch before the DHCP process on the client times out and tries to get a host IP address from the DHCP server. Decrease the settings for the 802.1x authentication process (**authentication timer inactivity** and **authentication timer reauthentication** interface configuration commands). The amount to decrease the settings depends on the connected 802.1x client type.
- When configuring the inaccessible authentication bypass feature, follow these guidelines:
  - The feature is supported on 802.1x port in single-host mode and multihosts mode.
  - If the client is running Windows XP and the port to which the client is connected is in the critical-authentication state, Windows XP might report that the interface is not authenticated.
  - If the Windows XP client is configured for DHCP and has an IP address from the DHCP server, receiving an EAP-Success message on a critical port might not reinitiate the DHCP configuration process.
  - You can configure the inaccessible authentication bypass feature and the restricted VLAN on an 802.1x port. If the switch tries to reauthenticate a critical port in a restricted VLAN and all the RADIUS servers are unavailable, switch changes the port state to the critical authentication state and remains in the restricted VLAN.
- You can configure any VLAN except an RSPAN VLAN or a voice VLAN as an 802.1x restricted VLAN. The restricted VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.

MAC Authentication Bypass Guidelines

- Unless otherwise stated, the MAC authentication bypass guidelines are the same as the 802.1x authentication guidelines. For more information, see 802.1x Authentication Guidelines, page 216.
- If you disable MAC authentication bypass from a port after the port has been authorized with its MAC address, the port state is not affected.
- If the port is in the unauthorized state and the client MAC address is not the authentication-server database, the port remains in the unauthorized state. However, if the client MAC address is added to the database, the switch can use MAC authentication bypass to reauthorize the port.
- If the port is in the authorized state, the port remains in this state until reauthorization occurs.
- You can configure a timeout period for hosts that are connected by MAC authentication bypass but are inactive. The range is 1to 65535 seconds.

Maximum Number of Allowed Devices Per Port Guidelines

This is the maximum number of devices allowed on an 802.1x-enabled port:

- In single-host mode, only one device is allowed on the access VLAN. If the port is also configured with a voice VLAN, an unlimited number of Cisco IP phones can send and receive traffic through the voice VLAN.

- In multidomain authentication (MDA) mode, one device is allowed for the access VLAN, and one IP phone is allowed for the voice VLAN.
- In multiple-host mode, only one 802.1x supplicant is allowed on the port, but an unlimited number of non-802.1x hosts are allowed on the access VLAN. An unlimited number of devices are allowed on the voice VLAN.

How to Configure IEEE 802.1x Port-Based Authentication

802.1x Authentication Configuration Process

To configure 802.1x port-based authentication, you must enable authentication, authorization, and accounting (AAA) and specify the authentication method list. A method list describes the sequence and authentication method to be queried to authenticate a user.

To allow per-user ACLs or VLAN assignment, you must enable AAA authorization to configure the switch for all network-related service requests.

This is the 802.1x AAA configuration process:

1. A user connects to a port on the switch.
2. Authentication is performed.
3. The VLAN assignment is enabled, as appropriate, based on the RADIUS server configuration.
4. The switch sends a start message to an accounting server.
5. Reauthentication is performed, as necessary.
6. The switch sends an interim accounting update to the accounting server, that is based on the result of reauthentication.
7. The user disconnects from the port.
8. The switch sends a stop message to the accounting server.

Beginning in privileged EXEC mode, follow these steps to configure 802.1x port-based authentication:

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**aaa new-model**	Enables AAA.
3.	**aaa authentication dot1x** {**default**} *method1*	Creates an 802.1x authentication method list. To create a default list to use when a named list is *not* specified in the **authentication** command, use the **default** keyword followed by the method to use in default situations. The default method list is automatically applied to all ports. For *method1*, enter the **group radius** keywords to use the list of all RADIUS servers for authentication. **Note:** Though other keywords are visible in the command-line help string, only the **group radius** keywords are supported.
4.	**dot1x system-auth-control**	Enables 802.1x authentication globally on the switch.

	Command	Purpose
5.	**aaa authorization network** {**default**} **group radius**	(Optional) Configures the switch to use user-RADIUS authorization for all network-related service requests, such as per-user ACLs or VLAN assignment. For per-user ACLs, single-host mode must be configured. This setting is the default.
6.	**radius-server host** *ip-address*	(Optional) Specifies the IP address of the RADIUS server.
7.	**radius-server key** *string*	(Optional) Specifies the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server.
8.	**interface** *interface-id*	Specifies the port connected to the client to enable for 802.1x authentication, and enter interface configuration mode.
9.	**switchport mode access**	(Optional) Sets the port to access mode only if you configured the RADIUS server in Step 6 and Step 7.
10.	**authentication port-control auto**	Enables 802.1x authentication on the port.
11.	dot1x pae authenticator	Sets the interface Port Access Entity to act only as an authenticator and ignore messages meant for a supplicant.
12.	**end**	Returns to privileged EXEC mode.
13.	**show authentication**	Verifies your entries.
14.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring the Switch-to-RADIUS-Server Communication

You can globally configure the timeout, retransmission, and encryption key values for all RADIUS servers by using the **radius-server host** global configuration command. If you want to configure these options on a per-server basis, use the **radius-server timeout**, **radius-server retransmit**, and the **radius-server key** global configuration commands. For more information, see Configuring Settings for All RADIUS Servers, page 176.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**radius-server host** {*hostname* | *ip-address*} **auth-port** *port-number* **key** *string*	Configures the RADIUS server parameters. *hostname* | *ip-address*–Specifies the hostname or IP address of the remote RADIUS server. **auth-port** *port-number*–Specifies the UDP destination port for authentication requests. The default is 1812. The range is 0 to 65536. **key** *string*–Specifies the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. The key is a text string that must match the encryption key used on the RADIUS server. **Note:** Always configure the key as the last item in the **radius-server host** command syntax because leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in the key, do not enclose the key in quotation marks unless the quotation marks are part of the key. This key must match the encryption used on the RADIUS daemon. If you want to use multiple RADIUS servers, reenter this command.
3.	**end**	Returns to privileged EXEC mode.
4.	**show running-config**	Verifies your entries.
5.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring 802.1x Readiness Check

	Command	Purpose
1.	**dot1x test eapol-capable** [**interface** *interface-id*]	Enables the 802.1x readiness check on the switch. *interface-id*–Specifies the port on which to check for 802.1x readiness. **Note:** If you omit the optional **interface** keyword, all interfaces on the switch are tested.
1.	**configure terminal**	(Optional) Enters global configuration mode.
2.	**dot1x test timeout** *timeout*	(Optional) Configures the timeout used to wait for EAPOL response. The range is from 1 to 65535 seconds. The default is 10 seconds.
3.	**end**	(Optional) Returns to privileged EXEC mode.
4.	**show running-config**	(Optional) Verifies your modified timeout values.

Enabling Voice Aware 802.1x Security

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**errdisable detect cause security-violation shutdown vlan**	Shuts down any VLAN on which a security violation error occurs. **Note:** If the **shutdown vlan** keywords are not included, the entire port enters the error-disabled state and shuts down.
3.	**errdisable recovery cause security-violation**	(Optional) Enables automatic per-VLAN error recovery.
4.	**clear errdisable interface** *interface-id* **vlan** [*vlan-list*]	(Optional) Reenables individual VLANs that have been error-disabled. ■ *interface-id*–Specifies the port on which to reenable individual VLANs. ■ (Optional) *vlan-list*–Specifies a list of VLANs to be reenabled. If *vlan-list* is not specified, all VLANs are reenabled.
5.	**shutdown** **no-shutdown**	(Optional) Reenables an error-disabled VLAN, and clear all error-disable indications.
6.	**end**	Returns to privileged EXEC mode.
7.	**show errdisable detect**	Verifies your entries.
8.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring 802.1x Violation Modes

You can configure an 802.1x port so that it shuts down, generates a syslog error, or discards packets from a new device when:

- A device connects to an 802.1x-enabled port
- The maximum number of allowed about devices have been authenticated on the port

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**aaa new-model**	Enables AAA.
3.	**aaa authentication dot1x** {**default**} *method1*	Creates an 802.1x authentication method list. To create a default list to use when a named list is *not* specified in the **authentication** command, use the **default** keyword followed by the method that is to be used in default situations. The default method list is automatically applied to all ports. *method1*–Specifies the **group radius** keywords to use the list of all RADIUS servers for authentication. **Note:** Though other keywords are visible in the command-line help string, only the **group radius** keywords are supported.
4.	**interface** *interface-id*	Specifies the port connected to the client that is to be enabled for 802.1x authentication, and enter interface configuration mode.

	Command	Purpose
5.	**switchport mode access**	Sets the port to access mode.
6.	**authentication violation** {**shutdown** | **restrict** | **protect** | **replace**}	Configures the violation mode. ■ **shutdown**–Error-disables the port. ■ **restrict**–Generates a syslog error. ■ **protect**–Drops packets from any new device that sends traffic to the port. ■ **replace**–Removes the current session and authenticates with the new host.
7.	**end**	Returns to privileged EXEC mode.
8.	**show authentication**	Verifies your entries.
9.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring the Host Mode

This task describes how to configure a single host (client) or multiple hosts on an 802.1x-authorized port.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**radius-server vsa send authentication**	Configures the network access server to recognize and use vendor-specific attributes (VSAs).
3.	**interface** *interface-id*	Specifies the port to which multiple hosts are indirectly attached, and enter interface configuration mode.
4.	**authentication host-mode** [**multi-auth** | **multi-domain** | **multi-host** | **single-host**]	The keywords have these meanings: ■ **multi-auth**–Allows one client on the voice VLAN and multiple authenticated clients on the data VLAN. Each host is individually authenticated. **Note:** The **multi-auth** keyword is only available with the **authentication host-mode** command. ■ **multi-host**–Allows multiple hosts on an 802.1x-authorized port after a single host has been authenticated. ■ **multi-domain**–Allows both a host and a voice device, such as an IP phone (Cisco or non-Cisco), to be authenticated on an 802.1x-authorized port. **Note:** You must configure the voice VLAN for the IP phone when the host mode is set to **multi-domain**. For more information, see Configuring Voice VLAN, page 327 ■ **single-host**–Allows a single host (client) on an 802.1x-authorized port. Make sure that the **authentication port-control** interface configuration command set is set to **auto** for the specified interface.
5.	**switchport voice vlan** *vlan-id*	(Optional) Configures the voice VLAN.

	Command	Purpose
6.	**end**	Returns to privileged EXEC mode.
7.	**show authentication interface** *interface-id*	Verifies your entries.
8.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Periodic Reauthentication

You can enable periodic 802.1x client reauthentication and specify how often it occurs. If you do not specify a time period before enabling reauthentication, the number of seconds between attempts is 3600. Beginning in privileged EXEC mode, follow these steps to enable periodic reauthentication of the client and to configure the number of seconds between reauthentication attempts. This procedure is optional.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the port to be configured, and enter interface configuration mode.
3.	authentication periodic	Enables periodic reauthentication of the client, which is disabled by default. **Note:** The default value is 3600 seconds. To change the value of the reauthentication timer or to have the switch use a RADIUS-provided session timeout, enter the **authentication timer reauthenticate** command.
4.	**authentication timer** {{[**inactivity** | **reauthenticate**]} {**restart** *value*}}	Sets the number of seconds between reauthentication attempts. ■ **inactivity**–Interval in seconds after which if there is no activity from the client then it is unauthorized ■ **reauthenticate**–Time in seconds after which an automatic reauthentication attempt is be initiated. ■ **restart** *value*–Interval in seconds after which an attempt is made to authenticate an unauthorized port. This command affects the behavior of the switch only if periodic reauthentication is enabled.
5.	**authentication timer reauthenticate** *seconds*	Sets the number of seconds that the switch waits for a response to an EAP-request/identity frame from the client before resending the request. The range is 1 to 65535 seconds; the default is 5. **Note:** You should change the default value of this command only to adjust for unusual circumstances such as unreliable links or specific behavioral problems with certain clients and authentication servers.
6.	**end**	Returns to privileged EXEC mode.
7.	**show authentication interface** *interface-id*	Verifies your entries.
8.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Configuring Optional 802.1x Authentication Features

	Command	Purpose
1.	**dot1x reauthenticate interface** *interface-id*	(Optional) Manually initiates a reauthentication of the specified IEEE 802.1x-enabled port.
2.	authentication mac-move permit	(Optional) Enables MAC move on the switch.
3.	**authentication violation** {**protect** | **replace** | **restrict** | **shutdown**}	(Optional) **replace**–Enables MAC replace on the interface. The port removes the current session and initiates authentication with the new host. The other keywords have these effects: ■ **protect**–Drops port packets with unexpected MAC addresses without generating a system message. ■ **restrict**–Drops violating packets by the CPU and a system message is generated. ■ **shutdown**–Error-disables the port when it receives an unexpected MAC address.
1.	**configure terminal**	Enters global configuration mode.
2.	mab request format attribute 32 vlan access-vlan	(Optional) Enables VLAN ID-based MAC authentication.
3.	**interface** *interface-id*	(Optional) Specifies the port to be configured, and enters interface configuration mode.
4.	**authentication timer inactivity** *seconds*	(Optional) Sets the number of seconds that the switch remains in the quiet state after a failed authentication exchange with the client. The range is 1 to 65535 seconds; the default is 60.
5.	**authentication timer reauthenticate** *seconds*	(Optional) Sets the number of seconds that the switch waits for a response to an EAP-request/identity frame from the client before resending the request. The range is 1 to 65535 seconds; the default is 5. **Note:** You should change the default value of this command only to adjust for unusual circumstances such as unreliable links or specific behavioral problems with certain clients and authentication servers.
6.	**dot1x max-reauth-req** *count*	(Optional) Sets the number of times that the switch sends an EAP-request/identity frame to the client before restarting the authentication process. The range is 1 to 10; the default is 2. **Note:** You should change the default value of this command only to adjust for unusual circumstances such as unreliable links or specific behavioral problems with certain clients and authentication servers.
7.	**dot1x max-req** *count*	(Optional) Sets the number of times that the switch restarts the authentication process before the port changes to the unauthorized state. The range is 0 to 10; the default is 2.

	Command	Purpose
8.	**authentication control-direction {both | in}**	(Optional) Enables 802.1x authentication with WoL on the port, and uses these keywords to configure the port as bidirectional or unidirectional. ■ **both**–Sets the port as bidirectional. The port cannot receive packets from or send packets to the host. By default, the port is bidirectional. ■ **in**–Sets the port as unidirectional. The port can send packets to the host but cannot receive packets from the host.
9.	**authentication order** [**mab**] {**webauth**}	(Optional) Sets the order of authentication methods. ■ **mab**–Adds MAC authentication bypass (MAB) to the order of authentication methods. ■ **webauth**–Adds web authentication to the order of authentication methods.
10.	**authentication order** [**dot1x | mab**] | {**webauth**}	(Optional) Sets the order of authentication methods used on a port.
11.	**authentication priority** [**dot1x** | **mab**] | {**webauth**}	(Optional) Adds an authentication method to the port-priority list.
12.	**dot1x default**	Resets the 802.1x parameters to the default values.
13.	**end**	Returns to privileged EXEC mode.
14.	**show authentication interface** *interface-id*	Verifies your entries.
15.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring 802.1x Accounting

Before You Begin

AAA must be enabled on your switch.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the port to be configured, and enter interface configuration mode.
3.	**aaa accounting dot1x default start-stop group radius**	Enables 802.1x accounting using the list of all RADIUS servers.
4.	**aaa accounting system default start-stop group radius**	(Optional) Enables system accounting (using the list of all RADIUS servers) and generates system accounting reload event messages when the switch reloads.
5.	**end**	Returns to privileged EXEc mode.
6.	**show running-config**	Verifies your entries.
7.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring a Guest VLAN

When you configure a guest VLAN, clients that are not 802.1x-capable are put into the guest VLAN when the server does not receive a response to its EAP request/identity frame. Clients that are 802.1x-capable but that fail authentication are not granted network access. The switch supports guest VLANs in single-host or multiple-hosts mode.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the port to be configured, and enters interface configuration mode.
3.	**switchport mode access** or **switchport mode private-vlan host**	Sets the port to access mode or Configures the Layer 2 port as a private-VLAN host port.
4.	**authentication port-control auto**	Enables 802.1x authentication on the port.
5.	**authentication event no-response action authorize vlan** *vlan-id*	Specifies an active VLAN as an 802.1x guest VLAN. The range is 1 to 4096. You can configure any active VLAN except an internal VLAN (routed port), an RSPAN VLAN, a primary private VLAN, or a voice VLAN as an 802.1x guest VLAN.
6.	**end**	Returns to privileged EXEC mode.
7.	**show authentication interface** *interface-id*	Verifies your entries.
8.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring a Restricted VLAN

When you configure a restricted VLAN on a switch, clients that are 802.1x-compliant are moved into the restricted VLAN when the authentication server does not receive a valid username and password. The switch supports restricted VLANs only in single-host mode.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the port to be configured, and enters interface configuration mode.
3.	**switchport mode access** or **switchport mode private-vlan host**	Sets the port to access mode, or Configures the Layer 2 port as a private-VLAN host port.
4.	**authentication port-control auto**	Enables 802.1x authentication on the port.
5.	*authentication event fail action authorize vlan-id*	Specifies an active VLAN as an 802.1x restricted VLAN. The range is 1 to 4096. You can configure any active VLAN except an internal VLAN (routed port), an RSPAN VLAN, a primary private VLAN, or a voice VLAN as an 802.1x restricted VLAN.

	Command	Purpose
6.	**end**	Returns to privileged EXEC mode.
7.	**show authentication interface** *interface-id*	(Optional) Verifies your entries.
8.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring the Maximum Number of Authentication Attempts

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the port to be configured, and enters interface configuration mode.
3.	**switchport mode access** or **switchport mode private-vlan host**	Sets the port to access mode, or Configures the Layer 2 port as a private-VLAN host port.
4.	**authentication port-control auto**	Enables 802.1x authentication on the port.
5.	**authentication event fail action authorize** *vlan-id*	Specifies an active VLAN as an 802.1x restricted VLAN. The range is 1 to 4096. You can configure any active VLAN except an internal VLAN (routed port), an RSPAN VLAN, a primary private VLAN, or a voice VLAN as an 802.1x restricted VLAN.
6.	**authentication event retry** *retry count*	Specifies a number of authentication attempts to allow before a port moves to the restricted VLAN. The range is 1 to 3, and the default is 3.
7.	**end**	Returns to privileged EXEC mode.
8.	**show authentication interface** *interface-id*	(Optional) Verifies your entries.
9.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Inaccessible Authentication Bypass

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**radius-server dead-criteria time** *time* **tries** *tries*	(Optional) Sets the conditions that are used to decide when a RADIUS server is considered unavailable or *dead*. The range for *time* is from 1 to 120 seconds. The switch dynamically determines the default *seconds* value that is 10 to 60 seconds. The range for *tries* is from 1 to 100. The switch dynamically determines the default *tries* parameter that is 10 to 100.
3.	**radius-server deadtime** *minutes*	(Optional) Sets the number of minutes that a RADIUS server is not sent requests. The range is from 0 to 1440 minutes (24 hours). The default is 0 minutes.

	Command	Purpose
4.	**radius-server host** *ip-address* **[acct-port** *udp-port*] [**auth-port** *udp-port*] [**test username** *name* [**idle-time** *time*] [**ignore-acct-port**] [**ignore-auth-port**]] [**key** *string*]	(Optional) Configures the RADIUS server parameters by using these keywords: ■ **acct-port** *udp-port*–Specifies the UDP port for the RADIUS accounting server. The range for the UDP port number is from 0 to 65536. The default is 1646. ■ **auth-port** *udp-port*–Specifies the UDP port for the RADIUS authentication server. The range for the UDP port number is from 0 to 65536. The default is 1645. **Note:** You should configure the UDP port for the RADIUS accounting server and the UDP port for the RADIUS authentication server to nondefault values. ■ **test username** *name*–Enables automated testing of the RADIUS server status, and specifies the username to be used. ■ **idle-time** *time*–Sets the interval of time in minutes after which the switch sends test packets to the server. The range is from 1 to 35791 minutes. The default is 60 minutes (1 hour). ■ **ignore-acct-port**–Disables testing on the RADIUS-server accounting port. ■ **ignore-auth-port**–Disables testing on the RADIUS-server authentication port. ■ **key** *string*–Specifies the authentication and encryption key for all RADIUS communication between the switch and the RADIUS daemon. **Note:** Always configure the key as the last item in the **radius-server host** command syntax because leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in the key, do not enclose the key in quotation marks unless the quotation marks are part of the key. This key must match the encryption used on the RADIUS daemon. You can also configure the authentication and encryption key by using the **radius-server key** {**0** *string* | **7** *string* | *string*} global configuration command.
5.	**dot1x critical** {**eapol** | **recovery delay** *milliseconds*}	(Optional) Configures the parameters for inaccessible authentication bypass. ■ **eapol**–Specifies that the switch sends an EAPOL-Success message when the switch successfully authenticates the critical port. ■ **recovery delay** *milliseconds*–Sets the recovery delay period during which the switch waits to reinitialize a critical port when a RADIUS server that was unavailable becomes available. The range is from 1 to 10000 milliseconds. The default is 1000 milliseconds (a port can be reinitialized every second).
6.	**interface** *interface-id*	Specifies the port to be configured, and enter interface configuration mode.
7.	**authentication event server dead action [authorize | reinitialize] vlan** *vlan-id*	Use these keywords to move hosts on the port if the RADIUS server is unreachable: ■ authorize–Moves any new hosts trying to authenticate to the user-specified critical VLAN. ■ **reinitialize**–Moves all authorized hosts on the port to the user-specified critical VLAN.
8.	**authentication event server dead action** {**authorize** | **reinitialize**} **vlan** *vlan-id*]	Enables the inaccessible authentication bypass feature and uses these keywords to configure the feature: ■ **authorize**–Authorizes the port. ■ **reinitialize**–Reinitializes all authorized clients.

	Command	Purpose
9.	**authentication server dead action authorize** [**vlan**]	Authorizes the switch in access VLAN or configured VLAN (if the VLAN is specified) when the ACS server is down.
10.	**end**	Returns to privileged EXEC mode.
11.	**show authentication interface** *interface-id*	(Optional) Verifies your entries.
12.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring 802.1x User Distribution

Beginning in global configuration, follow these steps to configure a VLAN group and to map a VLAN to it:

	Command	Purpose
1.	**vlan group** *vlan-group-name* **vlan-list** *vlan-list*	Configures a VLAN group, and maps a single VLAN or a range of VLANs to it.
2.	**show vlan group all** *vlan-group-name*	Verifies the configuration.
3.	**no vlan group** *vlan-group-name* **vlan-list** *vlan-list*	Clears the VLAN group configuration or elements of the VLAN group configuration.

Configuring NAC Layer 2 802.1x Validation

You can configure NAC Layer 2 802.1x validation, which is also referred to as 802.1x authentication with a RADIUS server.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the port to be configured, and enters interface configuration mode.
3.	**authentication event no-response action authorize vlan** *vlan-id*	Specifies an active VLAN as an 802.1x guest VLAN. The range is 1 to 4096. You can configure any active VLAN except an internal VLAN (routed port), an RSPAN VLAN, or a voice VLAN as an 802.1x guest VLAN.
4.	authentication periodic	Enables periodic reauthentication of the client, which is disabled by default.
5.	**authentication timer reauthenticate**	Sets reauthentication attempt for the client (set to one hour). This command affects the behavior of the switch only if periodic reauthentication is enabled.
6.	**end**	Returns to privileged EXEC mode.
7.	**show authentication interface** *interface-id*	Verifies your 802.1x authentication configuration.
8.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring an Authenticator and Supplicant

You can also use an Auto Smartports user-defined macro instead of the switch VSA to configure the authenticator switch. For information, seeConfiguring Smartports Macros, page 271.

Configuring an Authenticator

Before You Begin

One switch outside a wiring closet must be configured as a supplicant and be connected to an authenticator switch.

Note: The *cisco-av-pairs* must be configured as *device-traffic-class=switch* on the ACS, which sets the interface as a trunk after the supplicant is successfully authenticated.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	cisp enable	Enables CISP.
3.	**interface** *interface-id*	Specifies the port to be configured, and enters interface configuration mode.
4.	**switchport mode access**	Sets the port mode to **access**.
5.	authentication port-control auto	Sets the port-authentication mode to auto.
6.	dot1x pae authenticator	Configures the interface as a port access entity (PAE) authenticator.
7.	spanning-tree portfast	Enables Port Fast on an access port connected to a single workstation or server.
8.	**end**	Returns to privileged EXEC mode.
9.	**show running-config interface** *interface-id*	Verifies your configuration.
10.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring a Supplicant Switch with NEAT

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	cisp enable	Enables CISP.
3.	*dot1x credentials profile*	Creates 802.1x credentials profile. This must be attached to the port that is configured as supplicant.
4.	username *suppswitch*	Creates a username.
5.	password *password*	Creates a password for the new username.
6.	**dot1x supplicant force-multicast**	Forces the switch to send *only* multicast EAPOL packets when it receives either unicast or multicast packets. This also allows NEAT to work on the supplicant switch in all host modes.
7.	**interface** *interface-id*	Specifies the port to be configured, and enters interface configuration mode.
8.	switchport mode trunk	Configures the interface as a VLAN trunk port.
9.	dot1x pae supplicant	Configures the interface as a port access entity (PAE) supplicant.
10.	dot1x credentials *profile-name*	Attaches the 802.1x credentials profile to the interface.

	Command	Purpose
11.	**end**	Returns to privileged EXEC mode.
12.	**show running-config interface** *interface-id*	Verifies your configuration.
13.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring 802.1x Authentication with Downloadable ACLs and Redirect URLs

In addition to configuring 802.1x authentication on the switch, you need to configure the ACS. For more information, see the Cisco Secure ACS configuration guides.

Note: You must configure a downloadable ACL on the ACS before downloading it to the switch.

Configuring Downloadable ACLs

The policies take effect after client authentication and the client IP address addition to the IP device tracking table. The switch then applies the downloadable ACL to the port.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	ip device tracking	Configures the IP device tracking table.
3.	*aaa new-model*	Enables AAA.
4.	aaa authorization network default group radius	Sets the authorization method to local. To remove the authorization method, use the **no aaa authorization network default group radius** command.
5.	radius-server vsa send authentication	Configures the RADIUS VSA send authentication.
6.	**interface** *interface-id*	Specifies the port to be configured, and enters interface configuration mode.
7.	ip access-group *acl-id* in	Configures the default ACL on the port in the input direction. **Note:** The *acl-id* is an access list name or number.
8.	**show running-config interface** *interface-id*	Verifies your configuration.
9.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring a Downloadable Policy

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**access-list *access-list-number* deny** *source* **[*source-wildcard* log]**	Defines the default port ACL by using a source address and wildcard. The access-list-number is a decimal number from 1 to 99 or 1300 to 1999. **deny** or **permit**–Specifies whether to deny or permit access if conditions are matched. *source*–Specifies the source address of the network or host that sends a packet: ■ The 32-bit quantity in dotted-decimal format. ■ The keyword **any** as an abbreviation for **source** and *source-wildcard* value of 0.0.0.0 255.255.255.255. You do not need to enter a *source-wildcard* value. ■ The keyword host as an abbreviation for **source** and *source-wildcard* of source 0.0.0.0. (Optional) *source-wildcard*–Applies the wildcard bits to the source. (Optional) **log**–Creates an informational logging message about the packet that matches the entry to be sent to the console.
3.	**interface *interface-id***	Enters interface configuration mode.
4.	ip access-group *acl-id* in	Configures the default ACL on the port in the input direction. **Note:** The *acl-id* is an access list name or number.
5.	**exit**	Returns to global configuration mode.
6.	*aaa new-model*	Enables AAA.
7.	aaa authorization network default group radius	Sets the authorization method to local. To remove the authorization method, use the **no aaa authorization network default group radius** command.
8.	*ip device tracking*	Enables the IP device tracking table. To disable the IP device tracking table, use the **no ip device tracking** global configuration commands.
9.	**ip device tracking probe** [**count** | **interval** | **use-svi**]	(Optional) Configures the IP device tracking table: ■ **count** *count*–Sets the number of times that the switch sends the ARP probe. The range is from 1 to 5. The default is 3. ■ **interval** *interval*–Sets the number of seconds that the switch waits for a response before resending the ARP probe. The range is from 30 to 300 seconds. The default is 30 seconds. ■ **use-svi**–Uses the switch virtual interface (SVI) IP address as source of ARP probes.
10.	*radius-server vsa send authentication*	Configures the network access server to recognize and uses vendor-specific attributes. **Note:** The downloadable ACL must be operational.

	Command	Purpose
11.	*end*	Returns to privileged EXEC mode.
12.	***show ip device tracking all***	Displays information about the entries in the IP device tracking table.
13.	*copy running-config startup-config*	(Optional) Saves your entries in the configuration file.

Configuring Open1x

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the port to be configured, and enters interface configuration mode.
3.	**authentication control-direction {both** | **in}**	(Optional) Configures the port control as unidirectional or bidirectional.
4.	**authentication fallback** *name*	(Optional) Configures a port to use web authentication as a fallback method for clients that do not support 802.1x authentication.
5.	**authentication host-mode** [**multi-auth** | **multi-domain** | **multi-host** | **single-host**]	(Optional) Sets the authorization manager mode on a port.
6.	authentication open	(Optional) Enables or disables open access on a port.
7.	**authentication order** [**dot1x | mab**] | {**webauth**}	(Optional) Sets the order of authentication methods used on a port.
8.	authentication periodic	(Optional) Enables or disables reauthentication on a port.
9.	**authentication port-control** {**auto** | **force-authorized** | **force-un authorized**}	(Optional) Enables manual control of the port authorization state.
10.	**show** authentication	(Optional) Verifies your entries.
11.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Resetting the 802.1x Authentication Configuration to the Default Values

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Enters interface configuration mode, and specifies the port to be configured.
3.	**dot1x default**	Resets the 802.1x parameters to the default values.
4.	**end**	Returns to privileged EXEC mode.
5.	**show authentication interface** *interface-id*	Verifies your entries.
6.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Monitoring and Maintaining IEEE 802.1x Port-Based Authentication

Command	Purpose
show dot1x all statistics	Displays 802.1x statistics for all ports.
show dot1x statistics interface *interface-id*	Displays 802.1x statistics for a specific port.
show dot1x all [**details** | **statistics** | **summary**]	Displays the 802.1x administrative and operational status for the switch.
show dot1x interface *interface-id*	Displays the 802.1x administrative and operational status for a specific port.

Configuration Examples for Configuring IEEE 802.1x Port-Based Authentication

Enabling a Readiness Check: Example

This example shows how to enable a readiness check on a switch to query a port. It also shows the response received from the queried port verifying that the device connected to it is 802.1x-capable:

```
switch# dot1x test eapol-capable interface GigabitEthernet1/18

DOT1X_PORT_EAPOL_CAPABLE:DOT1X: MAC 00-01-02-4b-f1-a3 on GigabitEthernet1/18 is EAPOL capable
```

Enabling 802.1x Authentication: Example

This example shows how to enable 802.1x authentication and to allow multiple hosts:

```
Switch(config)# interface GigabitEthernet1/18
Switch(config-if)# authentication port-control auto
Switch(config-if)# authentication host-mode multi-host
Switch(config-if)# end
```

Enabling MDA: Example

This example shows how to enable MDA and to allow both a host and a voice device on the port:

```
Switch(config)# interface GigabitEthernet1/18
Switch(config-if)# authentication port-control auto
Switch(config-if)# authentication host-mode multi-domain
Switch(config-if)# switchport voice vlan 101
Switch(config-if)# end
```

Disabling the VLAN Upon Switch Violoation: Example

This example shows how to configure the switch to shut down any VLAN on which a security violation error occurs:

```
Switch(config)# errdisable detect cause security-violation shutdown vlan
```

This example shows how to reenable all VLANs that were error-disabled:

```
Switch# clear errdisable interface GigabitEthernet1/18 vlan
```

You can verify your settings by entering the **show errdisable detect** privileged EXEC command.

Configuring the Radius Server Parameters: Example

This example shows how to specify the server with IP address 172.20.39.46 as the RADIUS server, to use port 1612 as the authorization port, and to set the encryption key to *rad123*, matching the key on the RADIUS server:

```
Switch(config)# radius-server host 172.120.39.46 auth-port 1612 key rad123
```

Configuring 802.1x Accounting: Example

This example shows how to configure 802.1x accounting. The first command configures the RADIUS server, specifying 1813 as the UDP port for accounting:

```
Switch(config)# radius-server host 172.120.39.46 auth-port 1812 acct-port 1813 key rad123
Switch(config)# aaa accounting dot1x default start-stop group radius
Switch(config)# aaa accounting system default start-stop group radius
```

Enabling an 802.1x Guest VLAN: Example

This example shows how to enable VLAN 2 as an 802.1x guest VLAN:

```
Switch(config)# interface GigabitEthernet1/18
Switch(config-if)# authentication event no-response action authorize vlan 2
```

This example shows how to set 3 as the quiet time on the switch, to set 15 as the number of seconds that the switch waits for a response to an EAP-request/identity frame from the client before resending the request, and to enable VLAN 2 as an 802.1x guest VLAN when an 802.1x port is connected to a DHCP client:

```
Switch(config-if)# authentication timer inactivity 3
Switch(config-if)# authentication timer reauthenticate 15
Switch(config-if)# authentication event no-response action authorize vlan 2
```

Displaying Authentication Manager Common Session ID: Examples

This example shows how the session ID appears in the output of the **show authentication** command. The session ID in this example is 160000050000000B288508E5:

```
Switch# show authentication sessions

Interface  MAC Address     Method   Domain   Status         Session ID
Fa4/0/4    0000.0000.0203  mab      DATA     Authz Success  160000050000000B288508E5
```

This is an example of how the session ID appears in the syslog output. The session ID in this example is also 160000050000000B288508E5:

```
1w0d: %AUTHMGR-5-START: Starting 'mab' for client (0000.0000.0203) on Interface Fa4/0/4 AuditSessionID
160000050000000B288508E5
1w0d: %MAB-5-SUCCESS: Authentication successful for client (0000.0000.0203) on Interface Fa4/0/4
AuditSessionID 160000050000000B288508E5
1w0d: %AUTHMGR-7-RESULT: Authentication result 'success' from 'mab' for client (0000.0000.0203) on
Interface Fa4/0/4 AuditSessionID 160000050000000B288508E5
```

The session ID is used by the NAD, the AAA server, and other report-analyzing applications to identify the client. The ID appears automatically. No configuration is required.

Configuring Inaccessible Authentication Bypass: Example

This example shows how to configure the inaccessible authentication bypass feature:

```
Switch(config)# radius-server dead-criteria time 30 tries 20
Switch(config)# radius-server deadtime 60
Switch(config)# radius-server host 1.1.1.2 acct-port 1550 auth-port 1560 test username user1 idle-time
30 key abc1234
Switch(config)# dot1x critical eapol
Switch(config)# dot1x critical recovery delay 2000
Switch(config)# interface gigabitethernet 1/1
Switch(config)# radius-server deadtime 60
Switch(config-if)# dot1x critical
Switch(config-if)# dot1x critical recovery action reinitialize
Switch(config-if)# dot1x critical vlan 20
Switch(config-if)# end
```

Configuring VLAN Groups: Examples

This example shows how to configure the VLAN groups, to map the VLANs to the groups, and to verify the VLAN group configurations and mapping to the specified VLANs:

```
switch(config)# vlan group eng-dept vlan-list 10

switch(config)# show vlan group group-name eng-dept
Group Name                      Vlans Mapped
-------------                   --------------
eng-dept                        10
switch# show dot1x vlan-group all
Group Name                      Vlans Mapped
-------------                   --------------
eng-dept                        10
hr-dept                         20
```

This example shows how to add a VLAN to an existing VLAN group and to verify that the VLAN was added:

```
switch(config)# vlan group eng-dept vlan-list 30
switch(config)# show vlan group eng-dept
Group Name                      Vlans Mapped
-------------                   --------------
eng-dept                        10,30
```

This example shows how to remove a VLAN from a VLAN group:

```
switch# no vlan group eng-dept vlan-list 10
```

This example shows that when all the VLANs are cleared from a VLAN group, the VLAN group is cleared:

```
switch(config)# no vlan group eng-dept vlan-list 30
Vlan 30 is successfully cleared from vlan group eng-dept.

switch(config)# show vlan group group-name eng-dept
```

This example shows how to clear all the VLAN groups:

```
switch(config)# no vlan group end-dept vlan-list all
switch(config)# show vlan-group all
```

For more information about these commands, see the *Cisco IOS Security Command Reference.*

Configuring NAC Layer 2 802.1x Validation: Example

This example shows how to configure NAC Layer 2 802.1x validation:

```
Switch# configure terminal
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# authentication periodic
Switch(config-if)# authentication timer reauthenticate
```

Configuring an 802.1x Authenticator Switch: Example

This example shows how to configure a switch as an 802.1x authenticator:

```
Switch# configure terminal
Switch(config)# cisp enable
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# switchport mode access
Switch(config-if)# authentication port-control auto
Switch(config-if)# dot1x pae authenticator
Switch(config-if)# spanning-tree portfast trunk
```

Configuring an 802.1x Supplicant Switch: Example

This example shows how to configure a switch as a supplicant:

```
Switch# configure terminal
Switch(config)# cisp enable
Switch(config)# dot1x credentials test
Switch(config)# username suppswitch
Switch(config)# password myswitch
Switch(config)# dot1x supplicant force-multicast
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# switchport mode trunk
Switch(config-if)# dot1x pae supplicant
Switch(config-if)# dot1x credentials test
Switch(config-if)# end
```

Configuring a Downloadable Policy: Example

This example shows how to configure a switch for a downloadable policy:

```
Switch# config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# aaa new-model
Switch(config)# aaa authorization network default group radius
Switch(config)# ip device tracking
Switch(config)# ip access-list extended default_acl
Switch(config-ext-nacl)# permit ip any any
Switch(config-ext-nacl)# exit
Switch(config)# radius-server vsa send authentication
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# ip access-group default_acl in
Switch(config-if)# exit
```

Configuring Open 1x on a Port: Example

This example shows how to configure open 1x on a port:

```
Switch# configure terminal
Switch(config)# interface GigabitEthernet1/17
Switch(config)# authentication control-direction both
Switch(config)# au ten tic at ion fallback profile1
Switch(config)# authentication host-mode multi-auth
Switch(config)# authentication open
Switch(config)# authentication order dot1x webauth
Switch(config)# authentication periodic
Switch(config)# authentication port-control auto
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*
Radius commands	*Cisco IOS Security Command Reference*
Switch authentication configuration	Configuring Switch-Based Authentication, page 143
Authenticator switch information	Configuring Smartports Macros, page 271

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

MACsec

Media Access Control Security (MACsec) is the IEEE 802.1AE standard for authenticating and encrypting packets between two MACsec-capable devices.

For information about MACsec, including details about MACsec and MACsec Key Agreement (MKA), how to configure MKA and MACsec, and how to configure Cisco TrustSec MACsec, see Configuring MACsec Encryption.

This chapter includes the following information about MACsec specific to the IE 4000, IE 4010, and IE 5000 switches:

- PSK Based MKA Support for MACsec, page 243
- Certificate-based MACsec Encryption, page 247

Note: On the IE 4000, IE 4010, and the IE 5000, MACsec is included in the IP Services image only.

Guidelines and Limitations

MACsec on the IE5000 has the following guidelines and limitations:

- Both models of IE 5000 downlinks are fully interoperable with IE 4000, IE 4010, Catalyst 9300/3850, and Catalyst IE 3x00 platforms.
- On the IE-5000-16S12P, uplinks are fully functional when connected to another IE-5000-16S12P or a Catalyst 3850.
- On the IE-5000-12S12P-10G, uplinks when running at 10GE are fully functional when connected to another IE-5000-12S12P-10G running at 10GE or to a Catalyst 3850 running at 10GE.
- When an IE 5000 uplink is connected to a Catalyst 9300, the IE 5000 must be the key server. **CSCvs36043**
- IE-5000-12S12P-10G uplinks MACsec is not currently supported at GE speeds. **CSCvs41335**
- IE-5000-16S12P uplinks connected to downlinks of the IE 5000 and IE 4000 is not currently supported. **CSCvs44292**

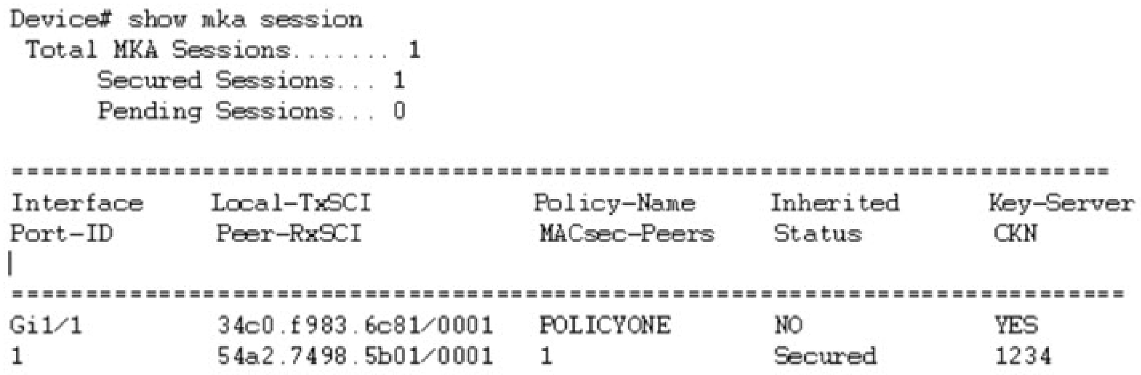
MKA-PSK: CKN Behavior Change

To interoperate with Cisco switches running IOS XE, the CKN configuration must be zero-padded. From Cisco IOS XE Everest Release 16.6.1 onwards, for MKA-PSK sessions, instead of fixed 32 bytes, the Connectivity Association Key name (CKN) uses exactly the same string as the CKN, which is configured as the hex-string for the key.

Example configuration:

```
configure terminal
 key chain KEYCHAINONE macsec
 key 1234
   cryptographic-algorithm aes-128-cmac
   key-string 123456789ABCDEF0123456789ABCDEF0
   lifetime local 12:21:00 Sep 9 2015 infinite
 end
```

For the above example, following is the output for the **show mka session** command:



```
Device# show mka session
 Total MKA Sessions....... 1
       Secured Sessions... 1
       Pending Sessions... 0

==========================================================================
Interface     Local-TxSCI           Policy-Name     Inherited      Key-Server
Port-ID        Peer-RxSCI            MACsec-Peers    Status         CKN
|
==========================================================================
Gi1/1          34c0.f983.6c81/0001   POLICYONE       NO             YES
1              54a2.7498.5b01/0001   1               Secured        1234
```

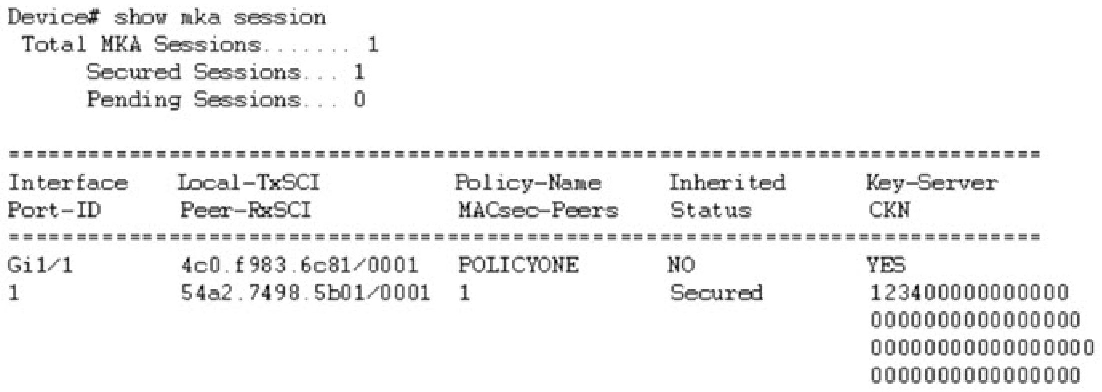
Note that the CKN key-string is exactly the same that has been configured for the key as hex-string.

For interoperability between two images, one having the CKN behavior change and one without the CKN behavior change, the hex-string for the key must be a 64-character hex-string padded with zeros to work on a device that has an image with the CKN behavior change. See the example below:

Configuration without CKN key-string behavior change:

```
config t
key chain KEYCHAINONE macsec
 key 1234
   cryptographic-algorithm aes-128-cmac
   key-string 123456789ABCDEF0123456789ABCDEF0
   lifetime local 12:21:00 Sep 9 2015 infinite
```

Output:



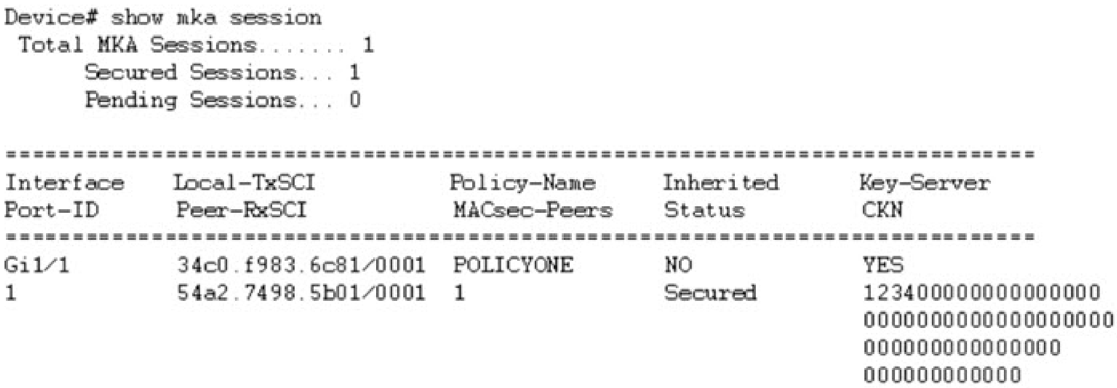
```
Device# show mka session
 Total MKA Sessions....... 1
       Secured Sessions... 1
       Pending Sessions... 0

==========================================================================
Interface    Local-TxSCI           Policy-Name     Inherited      Key-Server
Port-ID       Peer-RxSCI            MACsec-Peers    Status         CKN
==========================================================================
Gi1/1         4c0.f983.6c81/0001    POLICYONE       NO             YES
1             54a2.7498.5b01/0001   1               Secured        1234000000000000
                                                                   00000000000000000
                                                                   000000000000000000
                                                                   00000000000000000
```

Configuration with CKN key-string behavior change:

```
config t
key chain KEYCHAINONE macsec
 key 1234000000000000000000000000000000000000000000000000000000000000
   cryptographic-algorithm aes-128-cmac
   key-string 123456789ABCDEF0123456789ABCDEF0
   lifetime local 12:21:00 Sep 9 2015 infinite
```

Output:



```
Device# show mka session
 Total MKA Sessions....... 1
      Secured Sessions... 1
      Pending Sessions... 0

==============================================================================
Interface    Local-TxSCI          Policy-Name     Inherited      Key-Server
Port-ID      Peer-RxSCI           MACsec-Peers    Status         CKN
==============================================================================
Gi1/1        34c0.f983.6c81/0001  POLICYONE       NO             YES
1            54a2.7498.5b01/0001  1               Secured        123400000000000000
                                                                 0000000000000000000
                                                                 000000000000000
                                                                 000000000000
```

PSK Based MKA Support for MACsec

This section provides information about configuring pre-shared key (PSK) based MACsec Key Agreement (MKA) MACsec encryption on the switch. This feature applies to Cisco IOS Release 15.2(7)E1a and later.

Information about PSK Based MKA

IE switches support Pairwise Master Key (PMK) Security Association Protocol (SAP) based support for MACsec to interconnect links between the switches. The PMK keys can be either derived statically from the switch configuration (manual mode) or derived from the RADIUS server during dot1X negotiation (dynamic mode). Manual mode does not support switch-to-host MACsec connections because SAP is a Cisco proprietary protocol.

IE switches have MKA support for MACSec on switch-to-host links. Here the keys are derived from the RADIUS server after dot1x authentication. However, manually configured PSK keys were not supported on IE switch platforms (running Cisco IOS) prior to Cisco IOS Release 15.2(7)E1a. Catalyst IE 3x00 platforms (running Cisco IOS XE) have PSK based MKA support for MACsec for statically derived keys from the switch configuration for switch-to-switch connections as well as dynamically derived keys from RADIUS server for switch-to-host links.

Catalyst IE 3x00 platforms do not have PMK SAP based support for MACsec. Therefore, for interoperability with the Catalyst IE 3x00 platforms, the PSK functionality is added to MACsec for Cisco IOS based IE switches.

Configuring PSK Based MKA

Follow the procedures in this section to configure PSK based MKA on IE 4000, IE 4010, and IE 5000 switches.

Configuring MKA

The MACsec Key Agreement (MKA) enables configuration and control of keying parameters. Perform the following task to configure MKA.

	Command	Purpose
1.	**enable** Example: `Device> enable`	Enables privileged EXEC mode. - Enter your password if prompted.
2.	**configure terminal** Example: `Device# configure terminal`	Enters global configuration mode.
3.	**mka policy** *policy-name* Example: `Device(config)# mka policy MKAPolicy`	Configures an MKA policy.
4.	**key-server priority** *key-server-priority* Example: `Device(config-mka-policy)# key-server priority 200`	(Optional) Configures MKA key server priority.

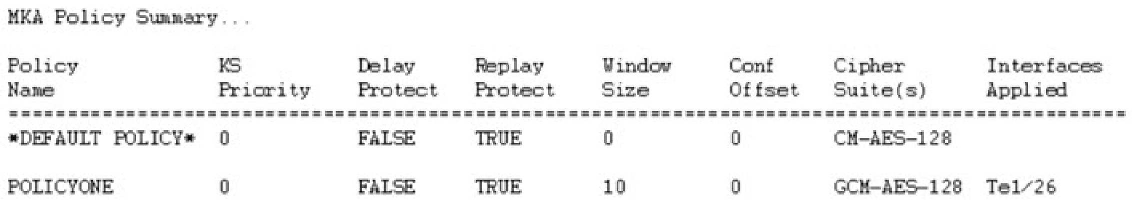
5.	**macsec-cipher-suite {gcm-aes-128 }**
	Example:
	`Device(config-mka-policy)# macsec-cipher-suite gcm-aes-128`
		(Optional) Configures cipher suite(s) for secure association key (SAK) derivation. Each of the cipher suite options can be repeated only once, but they can be used in any order.

6.	**replay-protection** Example: `Device(config-mka-policy)# replay-protection`	(Optional) Configure MKA to use replay protection for MACsec operation.
7.	**confidentiality-offset 30** Example: `Device(config-mka-policy)# confidentiality-offset 30`	(Optional) Configures confidentiality offset for MACsec operation.

8.	**end**
	Example:
	`Device(config-mka-policy)# end`
		Returns to privileged EXEC mode.

Example

You can use the **show mka policy** command to verify the configuration. Here's a sample output of the **show** command.



```
MKA Policy Summary...

Policy           KS          Delay    Replay   Window   Conf     Cipher       Interfaces
Name             Priority    Protect  Protect  Size     Offset   Suite(s)     Applied
==========================================================================================
*DEFAULT POLICY* 0           FALSE    TRUE     0        0        CM-AES-128

POLICYONE        0           FALSE    TRUE     10       0        GCM-AES-128  Te1/26
```

Configuring MACsec and MKA on Interfaces

Perform the following task to configure MACsec and MKA on an interface.

	Command	Purpose
1.	**enable** Example: `Device> enable`	Enables privileged EXEC mode. ■ Enter your password if prompted.
2.	**configure terminal** Example: `Device# configure terminal`	Enters global configuration mode.
3.	**interface** *type number* Example: `Device(config)# interface GigabitEthernet 1/1`	Enters interface configuration mode.
4.	**mka policy** *policy-name* Example: `Device(config-if)# mka policy MKAPolicy`	Configures an MKA policy.
5.	**mka pre-shared-key key-chain** *key-chain-name* Example: `Device(config-if)# mka pre-shared-key key-chain keychain1`	Configures an MKA pre-shared-key key-chain keychain1. **Note:** The MKA Pre-shared key can be configured on either physical interface or subinterfaces and not on both physical and subinterfaces.

	Command	Purpose
6.	**macsec network-link** Example: `Device(config-if)#macsec network-link`	Configures PSK MKA MACsec on this interface. This is mutually exclusive with macsec.
7.	**macsec replay-protection window-size** Example: `Device(config-if)# macsec replay-protection window-size 10`	Sets the MACsec window size for replay protection.

8.	**end**
	Example:
	`Device(config-mka-policy)# end`
		Returns to privileged EXEC mode.

Configuring MKA Pre-shared Key

Perform the following task to configure MACsec Key Agreement (MKA) pre-shared key.

	Command	Purpose
1.	**enable** Example: `Device> enable`	Enables privileged EXEC mode. ■ Enter your password if prompted.
2.	**configure terminal** Example: `Device# configure terminal`	Enters global configuration mode.
3.	**key chain** *key-chain-name* [**macsec**] Example: `Device(config)# Key chain keychain1 macsec`	Configures a key chain and enters keychain configuration mode
4.	**key** *hex-string* Example: `Device(config-keychain)# key 9ABCD`	Configures a key and enters keychain key configuration mode.

5.	**cryptographic-algorithm {gcm-aes-128 }**
	Example:
	`Device(config-keychain-key)# cryptographic-algorithm gcm-aes-128`
		Set cryptographic authentication algorithm.

	Command	Purpose

6.	**key-string** {[**0** | **6**] *pwd-string* | **7** | *pwd-string*}
	Example:
	`Device(config-keychain-key)# key-string 0 pwd`
		Sets the password for a key string.

7.	**lifetime local** {{*day month year* **duration** *seconds*}
	Example:
	`Device(config-keychain-key)# lifetime local 16:00:00 Nov 9 2014 duration 6000`
		Sets the lifetime for a key string.
		The range you can specify for the duration is between 1 and 864000 seconds.

8.	**end**
	Example:
	`Device(config-mka-policy)# end`
		Returns to privileged EXEC mode.

Certificate-based MACsec Encryption

This section provides information about Certificate-based MACsec Encryption. This feature applies to Cisco IOS Release 15.2(8)E and later.

Prerequisites for Certificate-based MACsec Encryption

- Certificate-based MACsec Encryption is supported on the IE4000, IE4010, and IE5000.
- Ensure that you have a Certificate Authority (CA) server configured for your network.
- Generate a CA certificate.
- Ensure that you have configured Cisco Identity Services Engine (ISE) Release 2.0. Refer to the Cisco Identity Services Engine Administrator Guide, Release 2.3.
- Ensure that both the participating devices, the CA server, and Cisco Identity Services Engine (ISE) are synchronized using Network Time Protocol (NTP). If time is not synchronized on all your devices, certificates will not be validated.
- Ensure that 802.1x authentication and AAA are configured on your device.

Restrictions for Certificate-based MACsec Encryption

- MKA is not supported on port-channels.
- High Availability for MKA is not supported.
- When you remove **dot1x pae both** from an interface, all configuration related to dot1x is removed from the interface.
- Certificate-based MACsec is supported only if the access-session host-mode is configured in multiple-host mode. The other configuration modes (multi-auth, multi-domain, or single-host) are not supported.

Information About Certificate-based MACsec Encryption

MKA MACsec is supported on switch-to-switch links. Using IEEE 802.1X Port-based Authentication with Extensible Authentication Protocol (EAP-TLS), you can configure MKA MACsec between device ports. EAP-TLS allows mutual authentication and obtains an MSK (master session key) from which the connectivity association key (CAK) is derived for MKA protocol. Device certificates are carried, using EAP-TLS, for authentication to the AAA server.

Refer to Certificate-based MACsec Encryption For more information about Certificate-based MACsec Encryption, including how to configure Certificate-based MACsec Encryption using Remote Authentication.

Configuring Certificate-based MACsec Encryption using Remote Authentication

Follow these procedures to configure MACsec encryption using remote authentication:

- Configure Certificate Enrollment Manually
- Configure an Authentication Policy
- Configure EAP-TLS Profiles and IEEE 802.1x Credentials
- Configure MKA MACsec using EAP-TLS on Interfaces

Configuring Certificate Enrollment Manually

If network connection between the router and CA is not possible, perform the following task to set up manual certificate enrollment:

	Command or Action	Purpose
1.	`enable`	Enables privileged EXEC mode. - Enter your password if prompted.
2.	`configure terminal`	Enters global configuration mode.
3.	**`crypto pki trustpoint`** *`server name`*	Declares the trustpoint and a given name and enters ca-trustpoint configuration mode.
4.	`enrollment terminal`	Enroll via the terminal (cut-and-paste).
5.	**`rsakeypair`** *`label`*	Specifies which key pair to associate with the certificate.
6.	`serial-number`	Specifies the router serial number in the certificate request.
7.	**`Subject-name`** *`Line`*	Declares the subject name. For example: subject-name cn=MUSTS.mkadt.cisco.com ,OU=CSG Security,O=Cisco Systems,L=Bengaluru,ST=KA,C=IN

8.	**subject-alt-name** *Line*	include subject alternative name.
9.	**fqdn** *Line*	include fully-qualified domain name.
10.	revocation-check none	The **none** keyword specifies to ignore revocation check.
11.	exit	Exits global configuration mode.
12.	**crypto pki authenticate** *name*	Retrieves the CA certificate and authenticates it.
13.	**crypto pki enroll** *name*	Generates certificate request and displays the request for copying and pasting into the certificate server. Enter enrollment information when you are prompted. For example, specify whether to include the device FQDN and IP address in the certificate request. You are also given the choice about displaying the certificate request to the console terminal. The base-64 encoded certificate with or without PEM headers as requested is displayed.
14.	**crypto pki import** *name* certificate	Imports a certificate via TFTP at the console terminal, which retrieves the granted certificate. The device attempts to retrieve the granted certificate via TFTP using the same filename used to send the request, except the extension is changed from ".req" to ".crt". For usage key certificates, the extensions "-sign.crt" and "-encr.crt" are used. The device parses the received files, verifies the certificates, and inserts the certificates into the internal certificate database on the switch. **Note:** Some CAs ignore the usage key information in the certificate request and issue general purpose usage certificates. If your CA ignores the usage key information in the certificate request, only import the general purpose certificate. The router will not use one of the two key pairs generated.
15.	exit	Exits global configuration mode.
16.	show crypto pki certificate trustpoint name	Displays information about the certificate for the trust point.
17.	copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Enabling 802.1x Authentication and Configuring AAA

	Command or Action	Purpose
1.	`enable`	Enables privileged EXEC mode. ■ Enter your password if prompted.
2.	`configure terminal`	Enters global configuration mode.
3.	`aaa new-model`	Enables AAA.
4.	`dot1x system-auth-control`	Enables 802.1X on your device.
5.	**radius server** *name*	Specifies the name of the RADIUS server configuration for Protected Access Credential (PAC) provisioning and enters RADIUS server configuration mode.
6.	**address** *ip-address* **auth-port** *port-number* **acct-port** *port-number*	Configures the IPv4 address for the RADIUS server accounting and authentication parameters.
7.	**automate-tester username** *username*	Enables the automated testing feature for the RADIUS server. With this practice, the device sends periodic test authentication messages to the RADIUS server. It looks for a RADIUS response from the server. A success message is not necessary - a failed authentication suffices, because it shows that the server is alive.
8.	**key** *string*	Configures the authentication and encryption key for all RADIUS communications between the device and the RADIUS server.
9.	**radius-server deadtime** *minutes*	Improves RADIUS response time when some servers might be unavailable and skips unavailable servers immediately.
10.	`exit`	Returns to global configuration mode.
11.	**aaa group server radius** *group-name*	Groups different RADIUS server hosts into distinct lists and distinct methods, and enters server group configuration mode.
12.	**server** *name*	Assigns the RADIUS server name.
13.	`exit`	Returns to global configuration mode.
14.	**aaa authentication dot1x default group** *group-name*	Sets the default authentication server group for IEEE 802.1x.
15.	**aaa authorization network default group** *group-name*	Sets the network authorization default group.

Configuring EAP-TLS Profile and 802.1x Credentials

	Command or Action	Purpose
1.	`enable`	Enables privileged EXEC mode. ■ Enter your password if prompted.
2.	`configure terminal`	Enters global configuration mode.
3.	**`eap profile`** *`profile-name`*	Configures EAP profile and enters EAP profile configuration mode.
4.	`method tls`	Enables EAP-TLS method on the device.
5.	**`pki-trustpoint`** *`name`*	Sets the default PKI trustpoint.
6.	`exit`	Returns to global configuration mode.
7.	**`dot1x credentials`** *`profile-name`*	Configures 802.1x credentials profile and enters dot1x credentials configuration mode.
8.	**`username`** *`username`*	Sets the authentication user ID.
9.	`end`	Returns to privileged EXEC mode.

Applying the 802.1x MKA MACsec Configuration on Interfaces

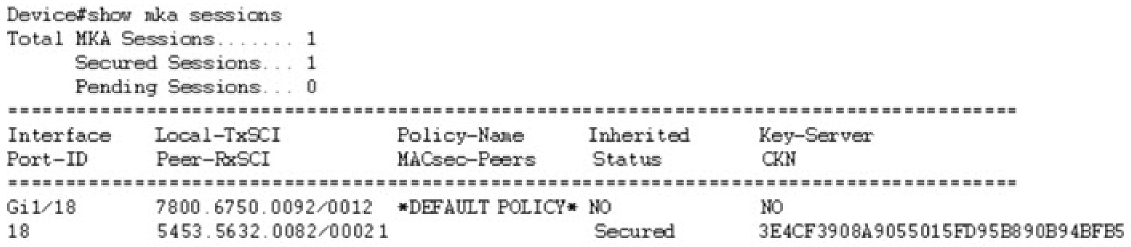
To apply MKA MACsec using EAP-TLS to interfaces, perform the following task:

	Command or Action	Purpose
1.	`enable`	Enables privileged EXEC mode. ■ Enter your password if prompted.
2.	`configure terminal`	Enters global configuration mode.
3.	**`interface`** *`interface-id`*	Identifies the MACsec interface, and enter interface configuration mode. The interface must be a physical interface.
4.	`macsec network-link`	Enables MACsec on the interface.
5.	`authentication periodic`	Enables reauthentication for this port.
6.	`access-session host-mode multi-host`	Allows hosts to gain access to the interface.
7.	`access-session closed`	Prevents preauthentication access on the interface.
8.	`access-session port-control auto`	Sets the authorization state of a port.
9.	`dot1x pae both`	Configures the port as an 802.1X port access entity (PAE) supplicant and authenticator.
10.	`dot1x credentials profile`	Assigns a 802.1x credentials profile to the interface.

11.	**dot1x supplicant eap profile** *name*	Assigns the EAP-TLS profile to the interface.
	dot1x authenticator eap profile *name*	Assigns the EAP-TLS profile to the interface
12.	service-policy type control subscriber control-policy name	Applies a subscriber control policy to the interface.
13.	exit	Returns to privileged EXEC mode.
14.	show macsec interface	Displays MACsec details for the interface.
15.	copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Verifying Certificate-based MACsec Encryption

Use the following **show** commands to verify the configuration of certificate-based MACsec encryption. Sample output is shown below.



```
Device#show mka sessions
Total MKA Sessions....... 1
      Secured Sessions... 1
      Pending Sessions... 0
========================================================================================
Interface    Local-TxSCI          Policy-Name     Inherited      Key-Server
Port-ID      Peer-RxSCI           MACsec-Peers    Status         CKN
========================================================================================
Gi1/18       7800.6750.0092/0012  *DEFAULT POLICY* NO             NO
18           5453.5632.0082/00021                 Secured        3E4CF3908A9055015FD95B890B94BFB5
```

The **show access-session interface** *interface-id* details displays detailed information about the access session for the given interface.

```
Device#show access-session interface gi 1/18 details
Interface:  GigabitEthernet1/18
          MAC Address:  5453.5632.0082
         IPv6 Address:  Unknown
         IPv4 Address:  Unknown
            User-Name:  scepen.mkadt.cisco.com
               Status:  Authorized
               Domain:  DATA
       Oper host mode:  multi-host
     Oper control dir:  both
      Session timeout:  N/A
      Restart timeout:  N/A
Periodic Acct timeout:  N/A
       Session Uptime:  25s
    Common Session ID:  000000000000000C0011E814
      Acct Session ID:  0x00000001
               Handle:  0xC0000001
       Current Policy:  MUSTS_1
Local Policies:
        Service Template: DEFAULT_LINKSEC_POLICY_MUST_SECURE (priority 150)
      Security Policy:  Must Secure
      Security Status:  Link Secured
Server Policies:
Method status list:
      Method            State
      dot1xSupp          Authc Success
      dot1x              Authc Success
```

Configuration examples for Certificate-based MACsec Encryption

Example: Enrolling theCertificate

Configure Crypto PKI Trustpoint:

```
crypto pki trustpoint demo
 enrollment terminal
 serial-number
 fqdn MUSTS.mkadt.cisco.com
 subject-name cn=MUSTS.mkadt.cisco.com,OU=CSG Security,O=Cisco Systems,L=Bengaluru,ST=KA,C=IN
 subject-alt-name MUSTS.mkadt.cisco.com
 revocation-check none
 rsakeypair demo 2048
 !
```

Manual Installation of Root CA certificate:

```
crypto pki authenticate demo
```

Example: Enabling 802.1x Authentication and AAA Configuration

```
aaa new-model
dot1x system-auth-control
radius server ISE
address ipv4 <ISE ipv4 address> auth-port 1645 acct-port 1646
key <secret configured on ise>
!
aaa group server radius ISEGRP
server name ISE
!
aaa authentication dot1x default group ISEGRP
aaa authorization network default group ISEGRP
!
```

Example: Configuring EAP-TLS Profile and 802.1X Credentials

```
eap profile scepen
 method tls
 pki-trustpoint demo
!
dot1x system-auth-control
dot1x credentials mis
 username scepen.mkadt.cisco.com
!
```

Example: Applying 802.1X, PKI, and MACsec Configuration on the Interface

```
interface GigabitEthernet1/2
 switchport mode access
 macsec network-link
 authentication periodic
 access-session host-mode multi-host
 access-session closed
 access-session port-control auto
 dot1x pae both
 dot1x authenticator eap profile scepen
 dot1x credentials mis
 dot1x supplicant eap profile scepen
 service-policy type control subscriber MUSTS_1
```

```
!
```

Configuring Web-Based Authentication

Prerequisites for Configuring Web-Based Authentication

- By default, the IP device tracking feature is disabled on a switch. You must enable the IP device tracking feature to use web-based authentication.
- You must configure at least one IP address to run the switch HTTP server. You must also configure routes to reach each host IP address. The HTTP server sends the HTTP login page to the host.
- You must configure the default ACL on the interface before configuring web-based authentication. Configure a port ACL for a Layer 2 interface.

Restrictions for Configuring Web-Based Authentication

- Web-based authentication is an ingress-only feature.
- You can configure web-based authentication only on access ports. Web-based authentication is not supported on trunk ports, EtherChannel member ports, or dynamic trunk ports.
- You cannot authenticate hosts on Layer 2 interfaces with static ARP cache assignment. These hosts are not detected by the web-based authentication feature because they do not send ARP messages.
- Hosts that are more than one hop away might experience traffic disruption if an STP topology change results in the host traffic arriving on a different port. This occurs because the ARP and DHCP updates might not be sent after a Layer 2 (STP) topology change.
- Web-based authentication does not support VLAN assignment as a downloadable-host policy.
- Web-based authentication is not supported for IPv6 traffic.
- Web-based authentication and Network Edge Access Topology (NEAT) are mutually exclusive. You cannot use web-based authentication when NEAT is enabled on an interface, and you cannot use NEAT when web-based authentication is running on an interface.
- Web-based authentication supports only RADIUS authorization servers. You cannot use TACACS+ servers or local authorization.

Information About Configuring Web-Based Authentication

Web-Based Authentication

Use the web-based authentication feature, known as *web authentication proxy*, to authenticate end users on host systems that do not run the IEEE 802.1x supplicant.

Note: You can configure web-based authentication on Layer 2 interfaces.

When you initiate an HTTP session, web-based authentication intercepts ingress HTTP packets from the host and sends an HTML login page to the users. The users enter their credentials, which the web-based authentication feature sends to the authentication, authorization, and accounting (AAA) server for authentication.

If authentication succeeds, web-based authentication sends a Login-Successful HTML page to the host and applies the access policies returned by the AAA server.

If authentication fails, web-based authentication forwards a Login-Fail HTML page to the user, prompting the user to retry the login. If the user exceeds the maximum number of attempts, web-based authentication forwards a Login-Expired HTML page to the host, and the user is placed on a watch list for a waiting period.

These sections describe the role of web-based authentication as part of AAA:

- Device Roles, page 256
- Host Detection, page 256
- Session Creation, page 257
- Authentication Process, page 257
- Web Authentication Customizable Web Pages, page 260
- Web-Based Authentication Interactions with Other Features, page 261

Device Roles

With web-based authentication, the devices in the network have these specific roles:

- Client—The device (workstation) that requests access to the LAN and the services and responds to requests from the switch. The workstation must be running an HTML browser with Java Script enabled.
- Authentication server—Authenticates the client. The authentication server validates the identity of the client and notifies the switch that the client is authorized to access the LAN and the switch services or that the client is denied.
- Switch—Controls the physical access to the network based on the authentication status of the client. The switch acts as an intermediary (proxy) between the client and the authentication server, requesting identity information from the client, verifying that information with the authentication server, and relaying a response to the client.

Figure 24 Web-Based Authentication Device Roles

Host Detection

The switch maintains an IP device tracking table to store information about detected hosts.

Note: By default, the IP device tracking feature is disabled on a switch. You must enable the IP device tracking feature to use web-based authentication.

For Layer 2 interfaces, web-based authentication detects IP hosts by using these mechanisms:

- ARP-based trigger—ARP redirect ACL allows web-based authentication to detect hosts with a static IP address or a dynamic IP address.
- Dynamic ARP inspection
- DHCP snooping—Web-based authentication is notified when the switch creates a DHCP-binding entry for the host.

Session Creation

When web-based authentication detects a new host, it creates a session as follows:

- Reviews the exception list.

  If the host IP is included in the exception list, the policy from the exception list entry is applied, and the session is established.

- Reviews for authorization bypass.

  If the host IP is not on the exception list, web-based authentication sends a nonresponsive-host (NRH) request to the server.

  If the server response is *access accepted*, authorization is bypassed for this host. The session is established.

- Sets up the HTTP intercept ACL.

  If the server response to the NRH request is *access rejected***,** the HTTP intercept ACL is activated, and the session waits for HTTP traffic from the host.

Authentication Process

When you enable web-based authentication, these events occur:

- The user initiates an HTTP session.
- The HTTP traffic is intercepted, and authorization is initiated. The switch sends the login page to the user. The user enters a username and password, and the switch sends the entries to the authentication server.
- If the authentication succeeds, the switch downloads and activates the user’s access policy from the authentication server. The login success page is sent to the user.
- If the authentication fails, the switch sends the login fail page. The user retries the login. If the maximum number of attempts fails, the switch sends the login expired page, and the host is placed in a watch list. After the watch list times out, the user can retry the authentication process.
- If the authentication server does not respond to the switch, and if an AAA fail policy is configured, the switch applies the failure access policy to the host. The login success page is sent to the user. (See Local Web Authentication Banner, page 258.)
- The switch reauthenticates a client when the host does not respond to an ARP probe on a Layer 2 interface, or when the host does not send any traffic within the idle timeout on a Layer 3 interface.
- The feature applies the downloaded timeout or the locally configured session timeout.
- If the terminate action is RADIUS, the feature sends a nonresponsive host (NRH) request to the server. The terminate action is included in the response from the server.
- If the terminate action is default, the session is dismantled, and the applied policy is removed.

Local Web Authentication Banner

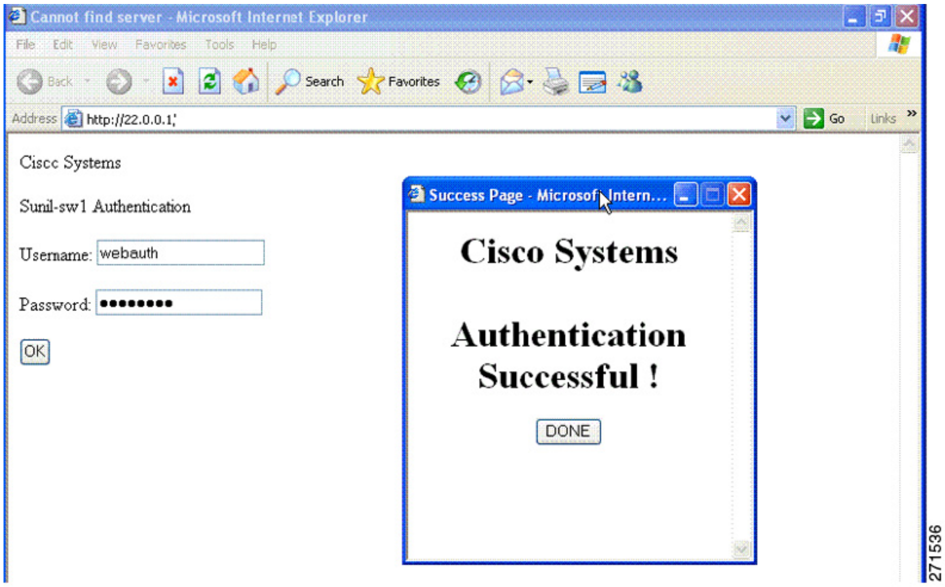
You can create a banner that will appear when you log in to a switch by using web authentication.

The banner appears on both the login page and the authentication-result pop-up pages:

- Authentication Successful
- Authentication Failed
- Authentication Expired

You create a banner by using the **ip admission auth-proxy-banner http** global configuration command. The default banner Cisco Systems and Switch host-name Authentication appear on the Login Page. Cisco Systems appears on the authentication result pop-up page, as shown in Figure 25 on page 258.

Figure 25 Authentication Successful Banner



Cisco Systems

Sunil-sw1 Authentication

Username: webauth

Password: ••••••••

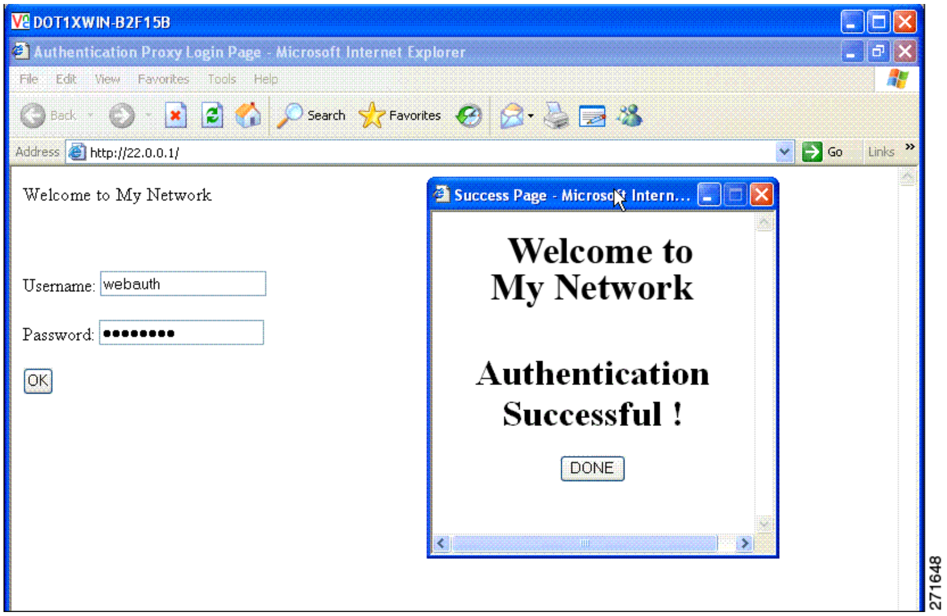
Cisco Systems

Authentication Successful !

You can also customize the banner, as shown in Figure 26 on page 259.

- Add a switch, router, or company name to the banner by using the **ip admission auth-proxy-banner http** *banner-text* global configuration command.
- Add a logo or text file to the banner by using the **ip admission auth-proxy-banner http** *file-path* global configuration command.

Figure 26 Customized Web Banner



DOT1XWIN-B2F15B

Authentication Proxy Login Page - Microsoft Internet Explorer

Address http://22.0.0.1/

Welcome to My Network

Username: webauth

Password: ••••••••

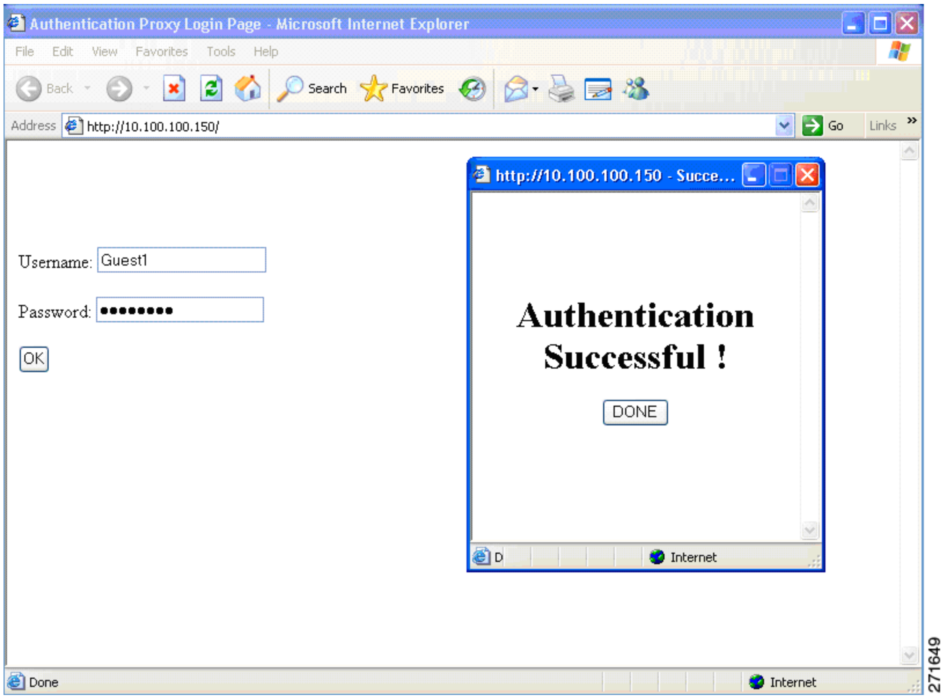
Success Page - Microsoft Intern...

Welcome to My Network

Authentication Successful !

If you do not enable a banner, only the username and password dialog boxes appear in the web authentication login screen, and no banner appears when you log into the switch, as shown in Figure 27.

Figure 27 Login Screen with No Banner



Authentication Proxy Login Page - Microsoft Internet Explorer

Address http://10.100.100.150/

Username: Guest1

Password: •••••••••

http://10.100.100.150 - Succe...

Authentication Successful !

For more information, see the *Cisco IOS Security Command Reference* and Configuring a Web Authentication Local Banner, page 266.

Web Authentication Customizable Web Pages

During the web-based authentication process, the switch internal HTTP server hosts four HTML pages to deliver to an authenticating client. The server uses these pages to notify you of these four-authentication process states:

- Login—Your credentials are requested.
- Success—The login was successful.
- Fail—The login failed.
- Expire—The login session has expired because of excessive login failures.

Web Authentication Guidelines

- You can substitute your own HTML pages for the default internal HTML pages.
- You can use a logo or specify text in the login, success, failure, and expire web pages.
- On the banner page, you can specify text in the login page.
- The pages are in HTML.
- You must include an HTML redirect command in the success page to access a specific URL.
- The URL string must be a valid URL (for example, http://www.cisco.com). An incomplete URL might cause page not found error or similar errors on a web browser.
- If you configure web pages for HTTP authentication, they must include the appropriate HTML commands (for example, to set the page time out, to set a hidden password, or to confirm that the same page is not submitted twice).
- The CLI command to redirect users to a specific URL is not available when the configured login form is enabled. The administrator should ensure that the redirection is configured in the web page.
- If the CLI command redirecting users to a specific URL after authentication occurs is entered and then the command configuring web pages is entered, the CLI command redirecting users to a specific URL does not take effect.
- Configured web pages can be copied to the switch boot flash or flash.
- Configured pages can be accessed from the flash on the stack master or members.
- The login page can be on one flash, and the success and failure pages can be another flash (for example, the flash on the stack master or a member).
- You must configure all four pages.
- The banner page has no effect if it is configured with the web page.
- All of the logo files (image, flash, audio, video, and so on) that are stored in the system directory (for example, flash, disk0, or disk) and that must be displayed on the login page must use web_auth_*filename* as the filename.
- The configured authentication proxy feature supports both HTTP and SSL.

When configuring customized authentication proxy web pages, follow these guidelines:

- To enable the custom web pages feature, specify all four custom HTML files. If you specify fewer than four files, the internal default HTML pages are used.
- The four custom HTML files must be present on the flash memory of the switch. The maximum size of each HTML file is 8 KB.

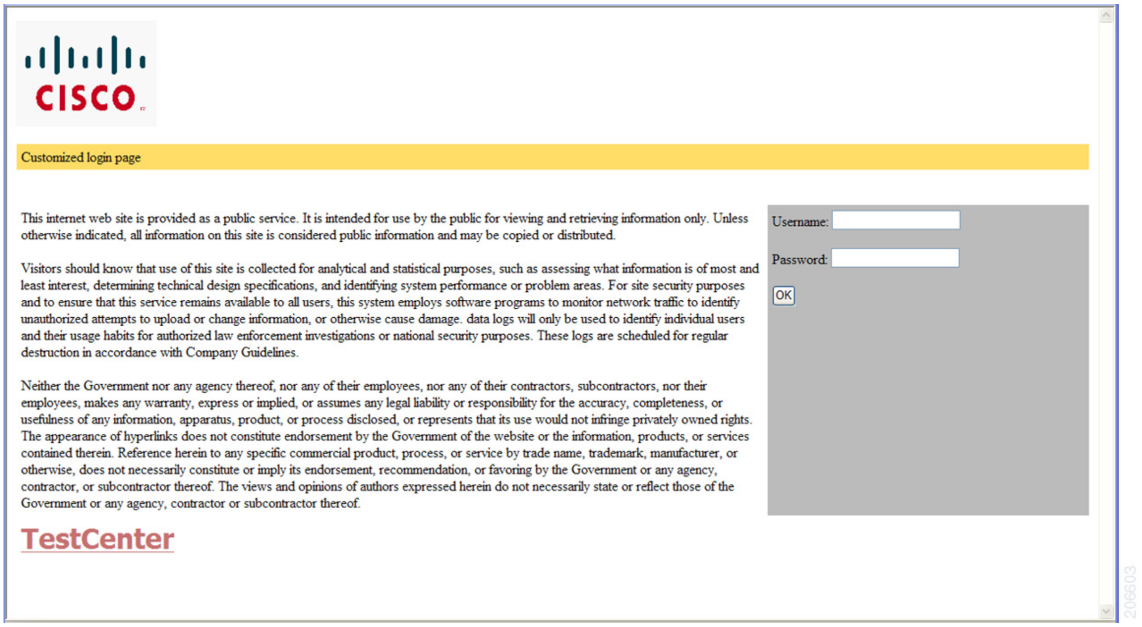
- Any images on the custom pages must be on an accessible HTTP server. Configure an intercept ACL within the admission rule.
- Any external link from a custom page requires configuration of an intercept ACL within the admission rule.
- To access a valid DNS server, any name resolution required for external links or images requires configuration of an intercept ACL within the admission rule.
- If the custom web pages feature is enabled, a configured auth-proxy-banner is not used.
- If the custom web pages feature is enabled, the redirection URL for successful login feature is not available.
- To remove the specification of a custom file, use the **no** form of the command.

Because the custom login page is a public web form, consider these guidelines for the page:

- The login form must accept user entries for the username and password and must show them as **uname** and **pwd**.
- The custom login page should follow best practices for a web form, such as page timeout, hidden password, and prevention of redundant submissions.

You can substitute your HTML pages, as shown in Figure 28, for the default internal HTML pages. You can also specify a URL to which users are redirected after authentication occurs, which replaces the internal Success page.

Figure 28 Customizeable Authentication Page



Customized login page

This internet web site is provided as a public service. It is intended for use by the public for viewing and retrieving information only. Unless otherwise indicated, all information on this site is considered public information and may be copied or distributed.

Visitors should know that use of this site is collected for analytical and statistical purposes, such as assessing what information is of most and least interest, determining technical design specifications, and identifying system performance or problem areas. For site security purposes and to ensure that this service remains available to all users, this system employs software programs to monitor network traffic to identify unauthorized attempts to upload or change information, or otherwise cause damage. data logs will only be used to identify individual users and their usage habits for authorized law enforcement investigations or national security purposes. These logs are scheduled for regular destruction in accordance with Company Guidelines.

Neither the Government nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, nor their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. The appearance of hyperlinks does not constitute endorsement by the Government of the website or the information, products, or services contained therein. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the Government or any agency, contractor, or subcontractor thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the Government or any agency, contractor or subcontractor thereof.

Web-Based Authentication Interactions with Other Features

- Port Security, page 262
- LAN Port IP, page 262
- Gateway IP, page 262
- ACLs, page 262

- Context-Based Access Control, page 262
- 802.1x Authentication, page 262
- EtherChannel, page 263

Port Security

You can configure web-based authentication and port security on the same port. Web-based authentication authenticates the port, and port security manages network access for all MAC addresses, including that of the client. You can then limit the number or group of clients that can access the network through the port.

LAN Port IP

You can configure LAN port IP (LPIP) and Layer 2 web-based authentication on the same port. The host is authenticated by using web-based authentication first, followed by LPIP posture validation. The LPIP host policy overrides the web-based authentication host policy.

If the web-based authentication idle timer expires, the NAC policy is removed. The host is authenticated, and posture is validated again.

Gateway IP

You cannot configure Gateway IP (GWIP) on a Layer 3 VLAN interface if web-based authentication is configured on any of the switch ports in the VLAN.

You can configure web-based authentication on the same Layer 3 interface as Gateway IP. The host policies for both features are applied in software. The GWIP policy overrides the web-based authentication host policy.

ACLs

If you configure a VLAN ACL or a Cisco IOS ACL on an interface, the ACL is applied to the host traffic only after the web-based authentication host policy is applied.

For Layer 2 web-based authentication, you must configure a port ACL (PACL) as the default access policy for ingress traffic from hosts connected to the port. After authentication, the web-based authentication host policy overrides the PACL.

Note: When a proxy ACL is configured for a web-based authentication client, the proxy ACL is downloaded and applied as part of the authorization process. Hence, the PACL displays the proxy ACL access control entry (ACE).

You cannot configure a MAC ACL and web-based authentication on the same interface.

You cannot configure web-based authentication on a port whose access VLAN is configured for VACL capture.

Context-Based Access Control

Web-based authentication cannot be configured on a Layer 2 port if context-based access control (CBAC) is configured on the Layer 3 VLAN interface of the port VLAN.

802.1x Authentication

You cannot configure web-based authentication on the same port as 802.1x authentication except as a fallback authentication method.

EtherChannel

You can configure web-based authentication on a Layer 2 EtherChannel interface. The web-based authentication configuration applies to all member channels.

Default Web-Based Authentication Settings

Feature	Default Settings
AAA	Disabled
RADIUS server ■ IP address ■ UDP authentication port ■ Key	 ■ None specified ■ 1812 ■ None specified
Default value of inactivity timeout	3600 seconds
Inactivity timeout	Enabled

Configuring Switch-to-RADIUS-Server Communication

RADIUS security servers identification:

- Host name
- Host IP address
- Host name and specific UDP port numbers
- IP address and specific UDP port numbers

The combination of the IP address and UDP port number creates a unique identifier, that enables RADIUS requests to be sent to multiple UDP ports on a server at the same IP address. If two different host entries on the same RADIUS server are configured for the same service (for example, authentication) the second host entry that is configured functions as the failover backup to the first one. The RADIUS host entries are chosen in the order that they were configured.

How to Configure Web-Based Authentication

Configuring the Authentication Rule and Interfaces

	Command	Purpose
1.	**ip admission name** *name* **proxy http**	Configures an authentication rule for web-based authorization.
2.	**interface** *type slot/port*	Enters interface configuration mode and specifies the ingress Layer 2 interface to be enabled for web-based authentication. *type* can be Gigabit Ethernet, or 10-Gigabit Ethernet.
3.	**ip access-group** *name*	Applies the default ACL.
4.	**ip admission** *name*	Configures web-based authentication on the specified interface.

	Command	Purpose
5.	**exit**	Returns to configuration mode.
6.	**ip device tracking**	Enables the IP device tracking table.
7.	**end**	Returns to privileged EXEC mode.
8.	**show ip admission configuration**	Displays the configuration.

Configuring AAA Authentication

	Command	Purpose
1.	**aaa new-model**	Enables AAA functionality.
2.	**aaa authentication login default group** {*tacacs+* | *radius*}	Defines the list of authentication methods at login.
3.	**aaa authorization auth-proxy default group** {*tacacs+* | *radius*}	Creates an authorization method list for web-based authorization.
4.	**radius-server host** {*hostname* | *ip-address*} **test username** *username*	Specifies an AAA server. Specifies the host name or IP address of the remote RADIUS server. The **test username** *username* option enables automated testing of the RADIUS server connection. The specified username does not need to be a valid user name.
5.	**radius-server key** *string*	Configures the authorization and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. To use multiple RADIUS servers, reenter this command for each server.

Configuring Switch-to-RADIUS-Server Communication

	Command	Purpose
1.	**ip radius source-interface** *interface_name*	Specifies that the RADIUS packets have the IP address of the indicated interface.
2.	**radius-server host** {*hostname* | *ip-address*} **test username** *username*	Specifies the host name or IP address of the remote RADIUS server. The **test username** *username* option enables automated testing of the RADIUS server connection. The specified *username* does not need to be a valid user name. The **key** option specifies an authentication and encryption key to use between the switch and the RADIUS server. To use multiple RADIUS servers, reenter this command for each server.

	Command	Purpose
3.	**radius-server key** *string*	Configures the authorization and encryption key used between the switch and the RADIUS daemon running on the RADIUS server.
4.	**radius-server vsa send authentication**	Enables downloading of an ACL from the RADIUS server. This feature is supported in Cisco IOS Release 12.2(50)SG.
5.	**radius-server dead-criteria tries** *num-tries*	Specifies the number of unanswered sent messages to a RADIUS server before considering the server to be inactive. The range of *num-tries* is 1 to 100.

Configuring the HTTP Server

	Command	Purpose
1.	**ip http server**	Enables the HTTP server. The web-based authentication feature uses the HTTP server to communicate with the hosts for user authentication.
2.	**ip http secure-server**	Enables HTTPS.

Customizing the Authentication Proxy Web Pages

Before You Begin

You can configure web authentication to display four substitute HTML pages to the user in place of the switch default HTML pages during web-based authentication.

To specify the use of your custom authentication proxy web pages, first store your custom HTML files on the switch flash memory, then perform this task in global configuration mode:

	Command	Purpose
1.	**ip admission proxy http login page file** *device:login-filename*	Specifies the location in the switch memory file system of the custom HTML file to use in place of the default login page. The *device:* is flash memory.
2.	**ip admission proxy http success page file** *device:success-filename*	Specifies the location of the custom HTML file to use in place of the default login success page.
3.	**ip admission proxy http failure page file** *device:fail-filename*	Specifies the location of the custom HTML file to use in place of the default login failure page.
4.	**ip admission proxy http login expired page file** *device:expired-filename*	Specifies the location of the custom HTML file to use in place of the default login expired page.

Specifying a Redirection URL for Successful Login

You can specify a URL to which the user is redirected after authentication, effectively replacing the internal *Success* HTML page.

Command	Purpose
ip admission proxy http success redirect *url-string*	Specifies a URL for redirection of the user in place of the default login success page.

Configuring the Web-Based Authentication Parameters

You can configure the maximum number of failed login attempts before the client is placed in a watch list for a waiting period.

	Command	Purpose
1.	**ip admission max-login-attempts** *number*	Sets the maximum number of failed login attempts. The range is 1 to 2147483647 attempts. The default is 5.
2.	**end**	Returns to privileged EXEC mode.
3.	**show ip admission configuration**	Displays the authentication proxy configuration.
4.	**show ip admission cache**	Displays the list of authentication entries.
5.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring a Web Authentication Local Banner

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**ip admission auth-proxy-banner http** [*banner-text* | *file-path*]	Enables the local banner. (Optional) Creates a custom banner by entering *C banner-text C,* where *C* is a delimiting character or a file-path indicates a file (for example, a logo or text file) that appears in the banner.
3.	end	Returns to privileged EXEC mode.
4.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Removing Web-Based Authentication Cache Entries

Enter a specific IP address to delete the entry for a single host. Use an asterisk to delete all cache entries.

Command	Purpose
clear ip auth-proxy cache {* | *host ip address*}	Clears authentication proxy entries from the switch.
clear ip admission cache {* | *host ip address*}	Clears IP admission cache entries from the switch.

Monitoring and Maintaining Web-Based Authentication

Command	Purpose
show authentication sessions	Displays the web-based authentication settings.
show ip admission configuration	Displays the authentication proxy configuration.
show ip admission cache	Displays the list of authentication entries.

Configuration Examples for Configuring Web-Based Authentication

Enabling and Displaying Web-Based Authentication: Examples

This example shows how to verify the configuration:

```
Switch# show ip admission configuration
Authentication Proxy Banner not configured
Authentication global cache time is 60 minutes
Authentication global absolute time is 0 minutes
Authentication global init state time is 2 minutes
Authentication Proxy Watch-list is disabled

Authentication Proxy Rule Configuration
 Auth-proxy name webauth1
    http list not specified inactivity-time 60 minutes

Authentication Proxy Auditing is disabled
Max Login attempts per user is 5
```

Enabling AAA: Example

This example shows how to enable AAA:

```
Switch(config)# aaa new-model
Switch(config)# aaa authentication login default group radius
Switch(config)# aaa authorization auth-proxy default group radius
```

Configuring the RADIUS Server Parameters: Example

This example shows how to configure the RADIUS server parameters on a switch:

```
Switch(config)# ip radius source-interface Vlan80
Switch(config)# radius-server host 172.120.39.46 test username user1
Switch(config)# radius-server key rad123
Switch(config)# radius-server dead-criteria tries 2
```

Configuring a Custom Authentication Proxy Web Page: Example

This example shows how to configure custom authentication proxy web pages:

```
Switch(config)# ip admission proxy http login page file flash:login.htm
Switch(config)# ip admission proxy http success page file flash:success.htm
Switch(config)# ip admission proxy http fail page file flash:fail.htm
Switch(config)# ip admission proxy http login expired page flash flash:expired.htm
```

Verifying a Custom Authentication Proxy Web Page: Example

This example shows how to verify the configuration of a custom authentication proxy web pages:

```
Switch# show ip admission configuration
Authentication proxy webpage
 Login page           : flash:login.htm
 Success page         : flash:success.htm
 Fail Page            : flash:fail.htm
 Login expired Page   : flash:expired.htm

Authentication global cache time is 60 minutes
Authentication global absolute time is 0 minutes
Authentication global init state time is 2 minutes
Authentication Proxy Session ratelimit is 100
Authentication Proxy Watch-list is disabled
Authentication Proxy Auditing is disabled
Max Login attempts per user is 5
```

Configuring a Redirection URL: Example

This example shows how to configure a redirection URL for successful login:

```
Switch(config)# ip admission proxy http success redirect www.cisco.com
```

Verifying a Redirection URL: Example

This example shows how to verify the redirection URL for successful login:

```
Switch# show ip admission configuration
Authentication Proxy Banner not configured
Customizable Authentication Proxy webpage not configured
HTTP Authentication success redirect to URL: http://www.cisco.com
Authentication global cache time is 60 minutes
Authentication global absolute time is 0 minutes
Authentication global init state time is 2 minutes
Authentication Proxy Watch-list is disabled
Authentication Proxy Max HTTP process is 7
Authentication Proxy Auditing is disabled
Max Login attempts per user is 5
```

Configuring a Local Banner: Example

This example shows how to configure a local banner with the custom message *My Switch*:

```
Switch(config) configure terminal
Switch(config)# aaa new-model
Switch(config)# aaa ip auth-proxy auth-proxy-banner C My Switch C
Switch(config) end
```

Clearing the Web-Based Authentication Session: Example

This example shows how to remove the web-based authentication session for the client at the IP address 209.165.201.1:

```
Switch# clear ip auth-proxy cache 209.165.201.1
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*
Authentication proxy commands Radius server commands	Cisco IOS Security Command Reference
Authentication proxy configuration Radius server configuration	*Cisco IOS Security Configuration Guide*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Configuring Smartports Macros

Information About Configuring Smartports Macros

Smartports macros provide a convenient way to save and share common configurations. You can use Smartports macros to enable features and settings based on the location of a switch in the network and for mass configuration deployments across the network.

Each Smartports macro is a set of CLI commands that you define. Smartports macros do not contain new CLI commands; they are simply a group of existing CLI commands.

When you apply a Smartports macro to an interface, the CLI commands within the macro are configured on the interface. When the macro is applied to an interface, the existing interface configurations are not lost. The new commands are added to the interface and are saved in the running configuration file.

How to Configure Smartports Macros

Default Smartports Settings

There are no Smartports macros enabled on the switch.

Table 31 Default Smartports Macros

Macro Name[1]	Description
Global Configuration Macros	
cisco-cg-global	Use this global configuration macro to configure the switch settings for the industrial Ethernet environment. This macro is automatically applied when you use Express Setup to initially configure the switch. **Note:** You must first apply the **cisco-cg-global** macro for the interface configuration macros to work properly.
cisco-cg-password	Use this global configuration macro to configure the password settings for the switch.
no-cisco-cg-password	Use the **no** form of this global configuration macro to delete the macro from the switch.
cisco-sniffer	Use this global configuration macro to configure SPAN functionality to analyze traffic on another port of the switch.
no-cisco-sniffer	Use the **no** form of this global configuration macro to delete the macro from the interface.
Interface Configuration Macros	
cisco-ethernetip	Use this interface configuration macro when connecting the switch to an EtherNet IP device. **Note:** You must first apply the **cisco-ie-global** macro for the **cisco-ethernetip** macro to work properly.
no-cisco-ethernetip	Use the **no** form of this global configuration macro to delete the macro from the interface.

Table 31 Default Smartports Macros (continued)

Macro Name[1]	Description
Global Configuration Macros	
cisco-cg-hmi	Use this interface configuration macro for increased network security and reliability when connecting a desktop device, such as a PC, to a switch port. This macro is optimized for utility deployments.
no-cisco-cg-hmi	Use the **no** form of this interface configuration macro to delete the macro from the switch.
cisco-cg-ied	Use this interface configuration macro when connecting the switch to an IED.
no-cisco-cg-ied	Use the **no** form of this interface configuration macro to delete the macro from the switch.
cisco-ie-phone	Use this interface configuration macro when connecting a desktop device such as a PC with a Cisco IP Phone to a switch port. This macro is an extension of the **cisco-desktop** macro and provides the same security and resiliency features, but with the addition of dedicated voice VLANs to ensure proper treatment of delay-sensitive voice traffic.
no-cisco-ie-phone	Use the **no** form of this interface configuration macro to delete the macro from the switch.
cisco-cg-router	Use this interface configuration macro when connecting the switch and a WAN router. This macro is optimized for utility deployments.
no-cisco-cg-router	Use the **no** form of this interface configuration macro to delete the macro from the switch.
cisco-cg-switch	Use this interface configuration macro when connecting a ring of switches. This macro is optimized for utility deployments.
no-cisco-cg-switch	Use the **no** form of this interface configuration macro to delete the macro from the switch.
cisco-cg-wireless	Use this interface configuration macro when connecting the switch and a wireless access point. This macro is optimized for utility deployments.
no-cisco-cg-wireless	Use the **no** form of this interface configuration macro to delete the macro from the switch.
cisco-desktop	Use this interface configuration macro for increased network security and reliability when connecting a desktop device, such as a PC, to a switch port. This macro is optimized for utility deployments.
no-cisco-desktop	Use the **no** form of this interface configuration macro to delete the macro from the interface.
cisco-ie-none	The None Smartport is used to clear all Smartport configurations from the port.
cisco-ethernetip	Use this interface configuration macro when connecting the switch to an EtherNet IP device. **Note:** You must first apply the **cisco-ie-global** macro for the **cisco-ethernetip** macro to work properly.
cisco-ie-global	Use this global configuration macro to configure the switch settings for the industrial Ethernet environment. This macro is automatically applied when you use Express Setup to initially configure the switch. **Note:** You must first apply the **cisco-ie-global** macro for the **cisco-ethernetip** macro to work properly.
cisco-ie-desktop	Use this interface configuration macro for increased network security and reliability when connecting a desktop device, such as a PC, to a switch port. This macro is optimized for industrial automation traffic.
cisco-ie-phone	Use this interface configuration macro when connecting a desktop device such as a PC with a Cisco IP Phone to a switch port. This macro is an extension of the **cisco-ie-desktop** macro and provides the same security and resiliency features, but with the addition of dedicated voice VLANs to ensure proper treatment of delay-sensitive voice traffic. This macro is optimized for industrial automation traffic.

Table 31 Default Smartports Macros (continued)

Macro Name[1]	Description
Global Configuration Macros	
cisco-ie-router	Use this interface configuration macro when connecting the switch and a WAN router. This macro is optimized for industrial automation traffic.
cisco-ie-switch	Use this interface configuration macro when connecting an access switch and a distribution switch or between access switches connected using small form-factor pluggable (SFP) modules. This macro is optimized for industrial automation traffic.
cisco-ie-wireless	Use this interface configuration macro when connecting the switch and a wireless access point. This macro is optimized for industrial automation traffic.

[1] Cisco-default Smartports macros vary, depending on the software version running on your switch.

Smartports Configuration Guidelines

- When a macro is applied globally to a switch or to a switch interface, all of the existing configurations on the interface are retained. This is helpful when applying an incremental configuration.
- If a command fails because of a syntax or a configuration error, the macro continues to apply the remaining commands. You can use the **macro global trace** *macro-name* global configuration command or the **macro trace** *macro-name* interface configuration command to apply and debug a macro to find any syntax or configuration errors.
- Some CLI commands are specific to certain interface types. If you apply a macro to an interface that does not accept the configuration, the macro fails the syntax or the configuration check, and the switch returns an error message.
- Applying a macro to an interface range is the same as applying a macro to a single interface. When you use an interface range, the macro is applied sequentially to each interface within the range. If a macro command fails on one interface, it is still applied to the remaining interfaces.
- When you apply a macro to a switch or a switch interface, the macro name is automatically added to the switch or interface. You can display the applied commands and macro names by using the **show running-config** user EXEC command.

Applying Smartports Macros

	Command	Purpose
1.	**show parser macro**	Displays the Cisco-default Smartports macros embedded in the switch software.
2.	**show parser macro name** *macro-name*	Displays the specific macro that you want to apply.
3.	**configure terminal**	Enters global configuration mode.

	Command	Purpose
4.	**macro global** {**apply** | **trace**} *macro-name* [**parameter** {*value*}] [**parameter** {*value*}] [**parameter** {*value*}]	Applies each individual command defined in the macro to the switch by entering **macro global apply** *macro-name*. Specifies **macro global trace** *macro-name* to apply and to debug a macro to find any syntax or configuration errors. Appends the macro with the required values by using the **parameter** *value* keywords. Keywords that begin with **$** require a unique parameter value. You can use the **macro global apply** *macro-name* **?** command to display a list of any required values for the macro. If you apply a macro without entering the keyword values, the commands are invalid and are not applied. (Optional) Specifies unique parameter values that are specific to the switch. You can enter up to three keyword-value pairs. Parameter keyword matching is case sensitive. The corresponding value replaces all matching occurrences of the keyword.
5.	**interface** *interface-id*	(Optional) Enters interface configuration mode and specifies the interface on which to apply the macro.
6.	**default interface** *interface-id*	(Optional) Clears all configuration from the specified interface.
7.	**macro** {**apply** | **trace**} *macro-name* [**parameter** {*value*}] [**parameter** {*value*}] [**parameter** {*value*}]	Applies each individual command defined in the macro to the port by entering **macro global apply** *macro-name*. Specifies **macro global trace** *macro-name* to apply and to debug a macro to find any syntax or configuration errors. Appends the macro with the required values by using the **parameter** *value* keywords. Keywords that begin with **$** require a unique parameter value. You can use the **macro global apply** *macro-name* **?** command to display a list of any required values for the macro. If you apply a macro without entering the keyword values, the commands are invalid and are not applied. (Optional) Specifies unique parameter values that are specific to the switch. You can enter up to three keyword-value pairs. Parameter keyword matching is case sensitive. The corresponding value replaces all matching occurrences of the keyword.
8.	**end**	Returns to privileged EXEC mode.
9.	**show running-config interface** *interface-id*	Verifies that the macro is applied to an interface.
10.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Monitoring and Maintaining Smartports Macros

Table 32 Commands for Displaying Smartports Macros

Command	Purpose
show parser macro	Displays all Smartports macros.

Table 32 Commands for Displaying Smartports Macros (continued)

Command	Purpose
show parser macro name *macro-name*	Displays a specific Smartports macro.
show parser macro brief	Displays the Smartports macro names.
show parser macro description [**interface** *interface-id*]	Displays the Smartports macro description for all interfaces or for a specified interface.

Configuration Examples for Smartports Macros

Applying the Smartports Macro: Examples

This example shows how to display the **cisco-ie-desktop** macro, how to apply the macro and to set the access VLAN ID to 25 on an interface:

```
Switch# show parser macro name cisco-ie-desktop
--------------------------------------------------------------
Macro name : cisco-ie-desktop
Macro type : default interface
# macro keywords ACCESS_VLAN
#macro name cisco-ie-desktop
switchport mode access
switchport access vlan ACCESS_VLAN
switchport port-security
switchport port-security maximum 1
switchport port-security aging time 2
switchport port-security violation restrict
switchport port-security aging type inactivity
spanning-tree portfast
spanning-tree bpduguard enable
no macro description
macro description cisco-ie-desktop
--------------------------------------------------------------
Switch#
Switch# configure terminal
Switch(config)# interface GigabitEthernet1/20
Switch(config-if)# macro apply cisco-ie-desktop $AVID 25
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Configuring SGACL Monitor Mode and SGACL Logging

SGACL Monitor Mode and SGACL Logging are supported on IE 4000, IE 4010, and IE 5000 Series Switches in Cisco IOS Release 15.2(8)E and later.

Security group-based access control is a component of the Cisco TrustSec security architecture, which builds secure networks by establishing domains of trusted network devices. For comprehensive information about TrustSec, including TrustSec prerequisites, guidelines and limitations, and configuration procedures, see Cisco TrustSec Switch Configuration Guide. For information about Configuring SGT Exchange Protocol over TCP (SXP) and Layer 3 Transport, see Configuring SGT Exchange Protocol over TCP (SXP) and Layer 3 Transport, page 281.

Restrictions for Configuring SGACL Policies

The following restrictions apply to the Cisco IE 4000, IE 4010, and IE 5000 Series Switches when configuring SGACL policies:

- Cisco TrustSec can be configured only on physical interfaces, not on logical interfaces.
- When SXP is configured between a Cisco IE 4000, IE 4010, or IE 5000 switch and another switch, SGACL policies are not enforced on Cisco IE 4000, IE 4010, or IE 5000 series switches. SGACL policies are downloaded for the destination SGT, but policy statements are not applied to the traffic that is initiated from the source SGT.

  IP device tracking must be enabled on both switches and these switches should have Layer2 adjacency configured between them so that a Cisco IE 4000, IE 4010, or IE 5000 can tag packets with the corresponding SGT learned via the SXP protocol.

  You can enable IP device tracking on Cisco IE 4000, IE 4010, or IE 5000 series switches by using the **ip device tracking maximum <** ***number*** **>** command. Based on your topology, configure the number of IP clients using the number argument. We do not recommend configuring a high number of IP clients on ports/interfaces.

  IP device tracking is enabled by default on all ports in Cisco IOS Release 15.2(1)E, and in Cisco IE 4000, IE 4010, and IE 5000 switches using this release image, SGACL policies are enforced.
- CTS SGACLs cannot be enforced for punt (CPU bound) traffic due to hardware limitations.

The following restrictions apply to IPv6 SGACL enforcement:

- SGACL enforcement is bypassed for IPv6 multicast traffic.
- SGACL enforcement is bypassed for IPv6 packets with Link-Local IPv6 source/destination addresses.

SGACL Monitor Mode

During the pre-deployment phase of Cisco TrustSec, you can use monitor mode to test security policies without enforcing them to make sure that the policies function as intended. If the security policies do not function as intended, monitor mode identifies that and allows you to correct the policy before enabling security group access control list (SGACL) enforcement. Seeing the outcome of the policy actions before enforcing them lets you confirm that the subject policy meets the security requirements (access is denied to resources if users are not authorized).

The monitoring capability is provided at the SGT-DGT pair level. When you enable the SGACL monitoring mode feature, the deny action is implemented as an ACL permit on the line cards. This allows the SGACL counters and logging to display how connections are handled by the SGACL policy. Since all the monitored traffic is permitted, there is no disruption of service due to SGACLs while in the SGACL monitor mode.

The IPServices license is required to enable SGACL Monitor Mode.

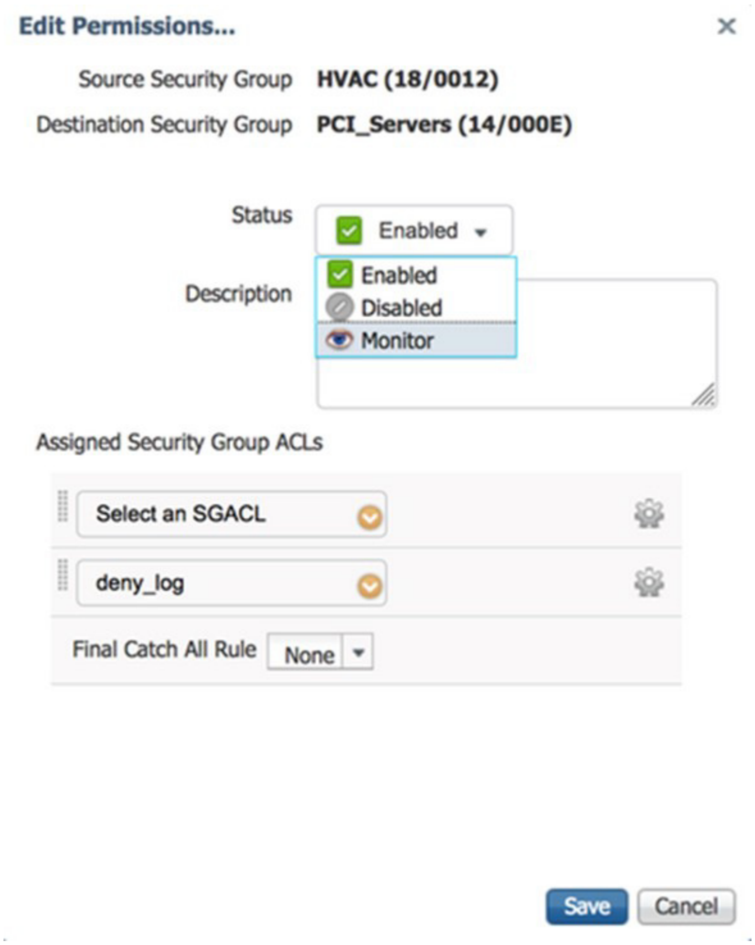
Configuring SGACL Monitor Mode - CLI

To configure SGACL Monitor Mode through the CLI, follow these steps:

	Command	Purpose
1.	Switch# **configure terminal**	Enters global configuration mode.
2.	Switch(config)#**cts role-based monitor enable**	Enables monitor mode.
3.	Switch(config)# **cts role-based monitor permissions from** { ***sgt_num*** } to { ***dgt_num*** } **ipv4**	Enables monitor mode for IPv4 RBACL (SGT-DGT pair).
4.	Switch(config)# **exit**	Exits configuration mode.
5.	Switch# **show cts role-based permissions from** { ***sgt_num*** } to { ***dgt_num*** } **ipv4** [**details**]	(Optional) Displays the SGACL policies and details about the monitor mode feature for each pair. The command output displays “monitored” if per cell monitor mode is enabled for the <SGT-DGT> pair.

Configuring SGACL Monitor Mode - Radius (ISE)

To enable SGACL Monitor Mode using the Cisco Identity Services Engine (ISE) GUI, select Monitor as shown below:



Source Security Group **HVAC (18/0012)**

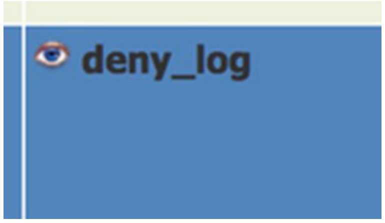
Destination Security Group **PCI_Servers (14/000E)**

Status: Enabled

deny_log

Final Catch All Rule: None

An eye icon indicates that Monitor mode is enabled.



The policy matrix change needs to be pushed to network devices by using the Deploy function at the top of the matrix. This utilizes RADIUS CoA to inform the devices that a change has been made.

After the update is downloaded to the switch, use the **show cts role-based permissions** command to verify the configuration. The policy permissions show the specific policy in Monitor Mode by appending the term “monitored”.

Verifying Configuration

The following examples are output from the **show cts role-based permissions** and **show cts role-based counters** commands, which you can use to display SGACL Monitor Mode status.

```
Switch#show cts role-based permissions
IPv4 Role-based permissions default:
        Permit IP-00
IPv4 Role-based permissions from group 18:HVAC to group 14:PCI_Servers (monitored):
        deny_log-10
RBACL Monitor All for Dynamic Policies : FALSE
RBACL Monitor All for Configured Policies : FALSE
```

The HW-Monitor column of **show cts role-based counters** displays the count of enforcement events that are being monitored in hardware and not actually enforced.

```
Switch#show cts role-based counters
Role-based IPv4 counters
From    To      SW-Denied  HW-Denied  SW-Permitt HW-Permitt SW-Monitor HW-Monitor
*       *       0          0          5378613    6291011    0          0
18      14      0          0          0          0          0          84
```

SGACL Logging

The **log** option in **cts** applies to individual ACEs and causes packets that match the ACE to be logged. The first packet logged by the **log** keyword generates a syslog message.

SGACL Logging is only triggered when the Cisco ACE Application Control Engine has the **logging** keyword.

When logging is enabled in SGACL, the switch logs the following information:

- The source security group tag (SGT) and destination SGT
- The SGACL policy name
- The packet protocol type
- The action performed on the packet

To enable Cisco TrustSec role-based (security group) access control enforcement, use the **cts role-based enforcement** command in global configuration mode. To configure a logging interval for an SGACL, enter:

cts role-based enforcement [**logging-interval** *interval*]

The valid values for the *interval* argument are from 5 to 86400 seconds. The default is 300 seconds.

To enable logging, use the **log** keyword before the ACE definition in the SGACL configuration. For example, **permit ip log**.

The following is a sample log, displaying source and destination SGTs, ACE matches for deny action). The **logging rate-limit** command can be used to limit the rate of messages logged per second.

```
Switch(config)# cts role-based enforcement logging-interval 90
Switch(config)# logging rate-limit

May 27 10:19:21.509: %RBM-6-SGACLHIT:
ingress_interface='GigabitEthernet1/0/2' sgacl_name='sgacl2' action='Deny'
protocol='icmp' src-ip='16.16.1.3' src-port='8' dest-ip='17.17.1.2' dest-port='0'
sgt='101' dgt='202' logging_interval_hits='5'
```

Configuring SGT Exchange Protocol over TCP (SXP) and Layer 3 Transport

You can use the SGT Exchange Protocol (SXP) to propagate the SGTs across network devices that do not have hardware support for Cisco TrustSec. This section describes how to configure Cisco TrustSec SXP on switches in your network.

This section includes the following topics:

Cisco TrustSec SGT Exchange Protocol Feature Histories

For a list of supported TrustSec features per platform and the minimum required Cisco IOS release, see the *Cisco TrustSec Platform Support Matrix* at the following URL: (final URL posted with TS 4.0)

http://www.cisco.com/en/US/solutions/ns170/ns896/ns1051/trustsec_matrix.html

Otherwise, see product release notes for detailed feature introduction information.

Note: **vrf aware sgt** is currently not supported.

Configuring Cisco TrustSec SXP

To configure Cisco TrustSec SXP, follow these steps:

1. Enable the Cisco TrustSec feature (see the “Configuring Identities, Connections, and SGTs” chapter in the Cisco TrustSec Switch Configuration Guide at: http://www.cisco.com/c/en/us/td/docs/switches/lan/trustsec/configuration/guide/trustsec/ident-conn_config.html#wpxref29406).
2. Enable Cisco TrustSec SXP (see Enabling Cisco TrustSec SXP, page 282).

3. Configure SXP peer connections (see Configuring an SXP Peer Connection, page 282).

Enabling Cisco TrustSec SXP

You must enable Cisco TrustSec SXP before you can configure peer connections. To enable Cisco TrustSec SXP, perform this task:

	Command	Purpose
1.	`Router#` **`configure terminal`**	Enters global configuration mode.
2.	`Router(config)# [`**`no`**`]` **`cts sxp enable`**	Enables SXP for Cisco TrustSec.
3.	`Router(config)#` **`exit`**	Exits configuration mode.

Configuring an SXP Peer Connection

You must configure the SXP peer connection on both of the devices. One device is the speaker and the other is the listener. When using password protection, make sure to use the same password on both ends.

Note: If a default SXP source IP address is not configured and you do not configure an SXP source address in the connection, the Cisco TrustSec software derives the SXP source IP address from existing local IP addresses. The SXP source address might be different for each TCP connection initiated from the switch.

To configure the SXP peer connection, perform this task:

	Command	Purpose
1.	Router# **configure terminal**	Enters global configuration mode.
2.	Router(config)# **cts sxp connection peer** *peer-ipv4-addr* [**source** *src-ipv4-addr*] **password** {**default** | **none**] **mode** {**local** | **peer**} {**speaker** | **listener**} [**vrf** *vrf-name*]	Configures the SXP address connection. The optional **source** keyword specifies the IPv4 address of the source device. If no address is specified, the connection will use the default source address, if configured, or the address of the port. The **password** keyword specifies the password that SXP will use for the connection using the following options: ■ **default**–Use the default SXP password you configured using the **cts sxp default password** command. ■ **none**–Do not use a password. The **mode** keyword specifies the role of the remote peer device: ■ **local**–The specified mode refers to the local device. ■ **peer**–The specified mode refers to the peer device. ■ **speaker**–Default. Specifies that the device is the speaker in the connection. ■ **listener**–Specifies that the device is the listener in the connection. The optional **vrf** keyword specifies the VRF to the peer. The default is the default VRF.
3.	Router(config)# **exit**	Exits configuration mode.
4.	Router# **show cts sxp connections**	(Optional) Displays the SXP connection information.

This example shows how to enable SXP and configure the SXP peer connection on Switch A, a speaker, for connection to Switch B, a listener:

```
Router# configure terminal
Router(config)# cts sxp enable
Router(config)# cts sxp default password Cisco123
Router(config)# cts sxp default source-ip 10.10.1.1
Router(config)# cts sxp connection peer 10.20.2.2 password default mode local speaker
```

This example shows how to configure the SXP peer connection on Switch B, a listener, for connection to Switch A, a speaker:

```
Router# configure terminal
Router(config)# cts sxp enable
Router(config)# cts sxp default password Cisco123
Router(config)# cts sxp default source-ip 10.20.2.2
Router(config)# cts sxp connection peer 10.10.1.1 password default mode local listener
```

Configuring the Default SXP Password

By default, SXP uses no password when setting up connections. You can configure a default SXP password for the switch. In Cisco IOS Release 12.2(50)SY and later releases, you can specify an encrypted password for the SXP default password.

To configure a default SXP password, perform this task:

	Command	Purpose
1.	`Router# configure terminal`	Enters configuration mode.
2.	`Router(config)# cts sxp default password [0 | 6 | 7] password`	Configures the SXP default password. You can enter either a clear text password (using the **0** or no option) or an encrypted password (using the **6** or **7** option). The maximum password length is 32 characters.
3.	`Router(config)# exit#`	Exits configuration mode.

This example shows how to configure a default SXP password:

```
Router# configure terminal
Router(config)# cts sxp default password Cisco123
```

Configuring the Default SXP Source IP Address

SXP uses the default source IP address for all new TCP connections where a source IP address is not specified. There is no effect on existing TCP connections when you configure the default SXP source IP address.

To configure a default SXP source IP address, perform this task:

	Command	Purpose
1.	`Router# configure terminal`	Enters configuration mode.
2.	`Router(config)# cts sxp default source-ip src-ip-addr`	Configures the SXP default source IP address.
3.	`Router(config)# exit`	Exits configuration mode.

This example shows how to configure an SXP default source IP address:

```
Router# configure terminal
Router(config)# cts sxp default source-ip 10.20.2.2
```

Changing the SXP Reconciliation Period

After a peer terminates an SXP connection, an internal hold-down timer starts. If the peer reconnects before the internal hold-down timer expires, the SXP reconciliation period timer starts. While the SXP reconciliation period timer is active, the Cisco TrustSec software retains the SGT mapping entries learned from the previous connection and removes invalid entries. The default value is 120 seconds (2 minutes). Setting the SXP reconciliation period to 0 seconds disables the timer and causes all entries from the previous connection to be removed.

To change the SXP reconciliation period, perform this task:

	Command	Purpose
1.	`Router# configure terminal`	Enters configuration mode.
2.	`Router(config)# cts sxp reconciliation period seconds`	Changes the SXP reconciliation timer. The default value is 120 seconds (2 minutes). The range is from 0 to 64000.
3.	`Router(config)# exit`	Exits configuration mode.

Changing the SXP Retry Period

The SXP retry period determines how often the Cisco TrustSec software retries an SXP connection. When an SXP connection is not successfully set up, the Cisco TrustSec software makes a new attempt to set up the connection after the SXP retry period timer expires. The default value is 120 seconds. Setting the SXP retry period to 0 seconds disables the timer and retries are not attempted.

To change the SXP retry period, perform this task:

	Command	Purpose
1.	`Router# configure terminal`	Enters configuration mode.
2.	`Router(config)# cts sxp retry period seconds`	Changes the SXP retry timer. The default value is 120 seconds (2 minutes). The range is from 0 to 64000.
3.	`Router(config)# exit`	Exits configuration mode.

Creating Syslogs to Capture Changes of IP Address to SGT Mapping Learned Through SXP

When the **cts sxp log binding-changes** global configuration command is executed, SXP syslogs (sev 5 syslog) are generated whenever a change to IP address to SGT binding occurs (add, delete, change). These changes are learned and propagated on the SXP connection.

The default is **no cts sxp log binding-changes**.

To enable logging of binding changes, perform the following task:

	Command	Purpose
1.	`Router# configure terminal`	Enters configuration mode.
2.	`Router(config)# cts sxp log binding-changes`	Turns on logging for IP to SGT binding changes.

Verifying the SXP Connections

To view the SXP connections, perform this task:

	Command	Purpose
1.	Router# **show cts sxp connections** [**brief**]	Displays SXP status and connections.

This example shows how to view the SXP connections:

```
Router# show cts sxp connections

SXP              : Enabled
Default Password : Set
Default Source IP: 10.10.1.1
Connection retry open period: 10 secs
Reconcile period: 120 secs
Retry open timer is not running
----------------------------------------------
Peer IP          : 10.20.2.2
Source IP        : 10.10.1.1
Conn status      : On
Conn Version     : 2
Connection mode  : SXP Listener
Connection inst# : 1
TCP conn fd      : 1
TCP conn password: default SXP password
Duration since last state change: 0:00:21:25 (dd:hr:mm:sec)
Total num of SXP Connections = 1
```

Configuring Cisco TrustSec Caching

Feature Name	Releases	Feature Information
TrustSec Caching	12.2(50) SY	This feature was introduced on the Catalyst 6500 series switches.

Enabling Cisco TrustSec Caching

For quick recovery from brief outages, you can enable caching of authentication, authorization, and policy information for Cisco TrustSec connections. Caching allows Cisco TrustSec devices to use unexpired security information to restore links after an outage without requiring a full reauthentication of the Cisco TrustSec domain. The Cisco TrustSec devices will cache security information in DRAM. If non-volatile (NV) storage is also enabled, the DRAM cache information will also be stored to the NV memory. The contents of NV memory populate DRAM during a reboot.

Note: During extended outages, the Cisco TrustSec cache information is likely to become outdated.

To enable Cisco TrustSec caching, perform this task:

	Command	Purpose
1.	`Router# configure terminal`	Enters configuration mode.
2.	`Router(config)# [no] cts cache enable`	Enables caching of authentication, authorization and environment-data information to DRAM. The default is disabled. The **no** form of this command deletes all cached information from DRAM and non-volatile storage.
3.	`Router(config)# [no] cts cache nv-storage {bootdisk: | bootflash: | disk0:} [directory dir-name]`	When DRAM caching is enabled, enables DRAM cache updates to be written to non-volatile storage. Also enables DRAM cache to be initially populated from non-volatile storage when the device boots.
4.	`Router(config)# exit`	Exits configuration mode.

This example shows how to configure Cisco TrustSec caching, including non-volatile storage:

```
Router# configure terminal
Router(config)# cts cache enable
Router(config)# cts cache nv-storage bootdisk:
Router(config)# exit
```

Clearing the Cisco TrustSec Cache

To clear the cache for Cisco TrustSec connections, perform this task:

	Command	Purpose
1.	`Router# clear cts cache [authorization-policies [peer] | environment-data | filename filename | interface-controller [type slot/port]]`	Clears the cache for Cisco TrustSec connection information.

This example shows how to clear the Cisco TrustSec cache:

```
Router# clear cts cache
```

Configuring VLANs

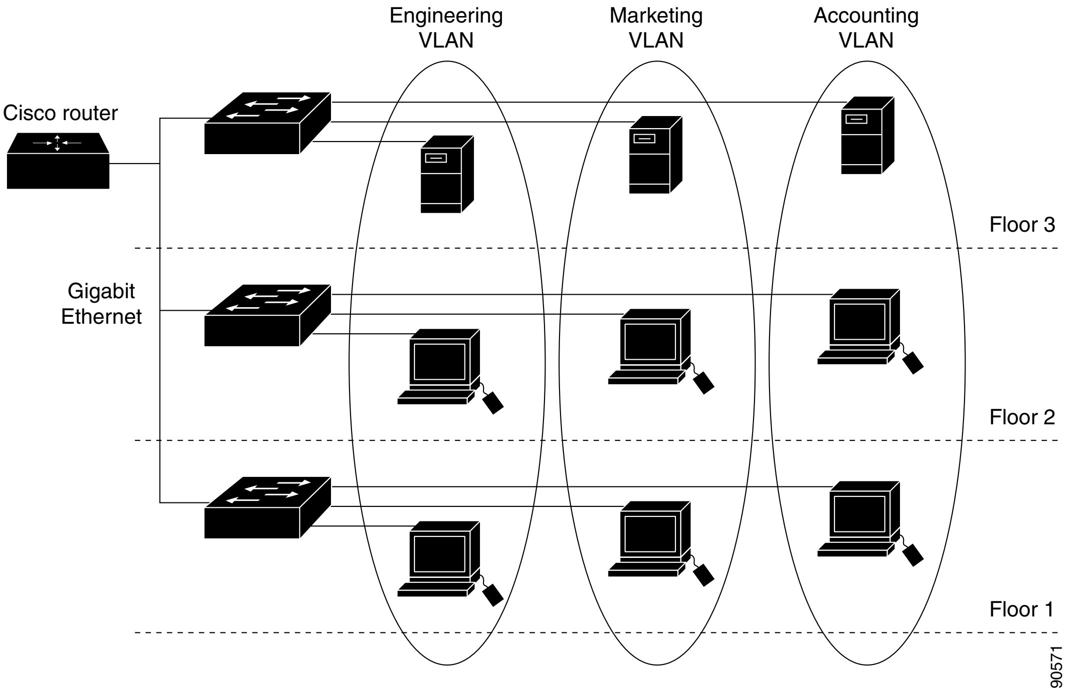
Information About Configuring VLANs

VLANs

A VLAN is a switched network that is logically segmented by function, project team, or application, without regard to the physical locations of the users. VLANs have the same attributes as physical LANs, but you can group end stations even if they are not physically located on the same LAN segment. Any switch port can belong to a VLAN, and unicast, broadcast, and multicast packets are forwarded and flooded only to end stations in the VLAN. Each VLAN is considered a logical network, and packets destined for stations that do not belong to the VLAN must be forwarded through a router or a switch supporting fallback bridging, as shown in Figure 29 on page 289. Because a VLAN is considered a separate logical network, it contains its own bridge Management Information Base (MIB) information and can support its own implementation of spanning tree. See Configuring STP, page 333

Note: Before you create VLANs, you must decide whether to use VLAN Trunking Protocol (VTP) to maintain global VLAN configuration for your network.

Figure 29 VLANs as Logically Defined Networks



VLANs are often associated with IP subnetworks. For example, all the end stations in a particular IP subnet belong to the same VLAN. Interface VLAN membership on the switch is assigned manually on an interface-by-interface basis. When you assign switch interfaces to VLANs by using this method, it is known as interface-based, or static, VLAN membership.

Traffic between VLANs must be routed or fallback bridged. The switch can route traffic between VLANs by using switch virtual interfaces (SVIs). An SVI must be explicitly configured and assigned an IP address to route traffic between VLANs.

Note: If you plan to configure many VLANs on the switch and to not enable routing, you can use the **sdm prefer vlan** global configuration command to set the Switch Database Management (sdm) feature to the VLAN template, which configures system resources to support the maximum number of unicast MAC addresses. For more information on the SDM templates, see Configuring SDM Templates, page 137

Supported VLANs

The switch supports VLANs in VTP client, server, and transparent modes. VLANs are identified by a number from 1 to 4096. VLAN IDs 1002 through 1005 are reserved for Token Ring and FDDI VLANs.

VTP version 1 and version 2 support only normal-range VLANs (VLAN IDs 1 to 1005). In these versions, the switch must be in VTP transparent mode when you create VLAN IDs from 1006 to 4096.

This release supports VTP version 3. VTP version 3 supports the entire VLAN range (VLANs 1 to 4096). Extended range VLANs (VLANs 1006 to 4096) are supported only in VTP version 3. You cannot convert from VTP version 3 to VTP version 2 if extended VLANs are configured in the domain.

Although the switch supports a total of 1005 (normal range and extended range) VLANs, the number of routed ports, SVIs, and other configured features affects the use of the switch hardware.

The switch supports per-VLAN spanning-tree plus (PVST+) or rapid PVST+ with a maximum of 128 spanning-tree instances. One spanning-tree instance is allowed per VLAN. See Normal-Range VLAN Configuration Guidelines, page 293 for more information about the number of spanning-tree instances and the number of VLANs.

VLAN Port Membership Modes

You configure a port to belong to a VLAN by assigning a membership mode that specifies the kind of traffic the port carries and the number of VLANs to which it can belong. Table 33 on page 291 lists the membership modes and membership and VTP characteristics.

Table 33 Port Membership Modes and Characteristics

Membership Mode	VLAN Membership Characteristics	VTP Characteristics
Static-access	A static-access port can belong to one VLAN and is manually assigned to that VLAN. For more information, see Assigning Static-Access Ports to a VLAN, page 303.	VTP is not required. If you do not want VTP to globally propagate information, set the VTP mode to transparent. To participate in VTP, there must be at least one trunk port on the switch connected to a trunk port of a second switch.
Trunk (ISL or IEEE 802.1Q)	A trunk port is a member of all VLANs by default, including extended-range VLANs, but membership can be limited by configuring the allowed-VLAN list. You can also modify the pruning-eligible list to block flooded traffic to VLANs on trunk ports that are included in the list. For information about configuring trunk ports, see Configuring an Ethernet Interface as a Trunk Port, page 304.	VTP is recommended but not required. VTP maintains VLAN configuration consistency by managing the addition, deletion, and renaming of VLANs on a network-wide basis. VTP exchanges VLAN configuration messages with other switches over trunk links.
Dynamic access	A dynamic-access port can belong to one VLAN and is dynamically assigned by a VMPS (VLAN Membership Policy Server). The VMPS can be a Catalyst 5000 or Catalyst 6500 series switch, for example, but never an IE 2000 switch. The IE 2000 switch is a VMPS client. You can have dynamic-access ports and trunk ports on the same switch, but you must connect the dynamic-access port to an end station or hub and not to another switch. For configuration information, see Configuring Dynamic-Access Ports on VMPS Clients, page 308.	VTP is required. Configure the VMPS and the client with the same VTP domain name. To participate in VTP, at least one trunk port on the switch must be connected to a trunk port of a second switch.
Voice VLAN	A voice VLAN port is an access port attached to a Cisco IP Phone, configured to use one VLAN for voice traffic and another VLAN for data traffic from a device attached to the phone. For more information about voice VLAN ports, see Configuring Voice VLAN, page 327	VTP is not required; it has no effect on a voice VLAN.

For more detailed definitions of access and trunk modes and their functions, see Table 35 on page 296.

When a port belongs to a VLAN, the switch learns and manages the addresses associated with the port on a per-VLAN basis. For more information, see Changing the Address Aging Time, page 115.

Normal-Range VLANs

Normal-range VLANs are VLANs with VLAN IDs 1 to 1005. If the switch is in VTP server or VTP transparent mode, you can add, modify or remove configurations for VLANs 2 to 1001 in the VLAN database. (VLAN IDs 1 and 1002 to 1005 are automatically created and cannot be removed.)

Configurations for VLAN IDs 1 to 1005 are written to the *vlan.dat* file (VLAN database), and you can display them by entering the **show vlan** privileged EXEC command. The *vlan.dat* file is stored in flash memory.

Caution: You can cause inconsistency in the VLAN database if you attempt to manually delete the *vlan.dat* file. If you want to modify the VLAN configuration, use the commands described in these sections.

You use the interface configuration mode to define the port membership mode and to add and remove ports from VLANs. The results of these commands are written to the running-configuration file, and you can display the file by entering the **show running-config** privileged EXEC command.

You can set these parameters when you create a new normal-range VLAN or modify an existing VLAN in the VLAN database:

- VLAN ID
- VLAN name
- VLAN type (Ethernet, Fiber Distributed Data Interface [FDDI], FDDI network entity title [NET], TrBRF, or TrCRF, Token Ring, Token Ring-Net)
- VLAN state (active or suspended)
- Maximum transmission unit (MTU) for the VLAN
- Security Association Identifier (SAID)
- Bridge identification number for TrBRF VLANs
- Ring number for FDDI and TrCRF VLANs
- Parent VLAN number for TrCRF VLANs
- Spanning Tree Protocol (STP) type for TrCRF VLANs
- VLAN number to use when translating from one VLAN type to another

You configure VLANs in **vlan** global configuration command by entering a VLAN ID. Enter a new VLAN ID to create a VLAN, or enter an existing VLAN ID to modify that VLAN. You can use the default VLAN configuration (Table 34 on page 294) or enter multiple commands to configure the VLAN. When you have finished the configuration, you must exit VLAN configuration mode for the configuration to take effect. To display the VLAN configuration, enter the **show vlan** privileged EXEC command.

The configurations of VLAN IDs 1 to 1005 are always saved in the VLAN database (vlan.dat file). If the VTP mode is transparent, they are also saved in the switch running configuration file. You can enter the **copy running-config startup-config** privileged EXEC command to save the configuration in the startup configuration file. To display the VLAN configuration, enter the **show vlan** privileged EXEC command.

When you save VLAN and VTP information (including extended-range VLAN configuration information) in the startup configuration file and reboot the switch, the switch configuration is selected as follows:

- If the VTP mode is transparent in the startup configuration, and the VLAN database and the VTP domain name from the VLAN database matches that in the startup configuration file, the VLAN database is ignored (cleared), and the VTP and VLAN configurations in the startup configuration file are used. The VLAN database revision number remains unchanged in the VLAN database.
- If the VTP mode or domain name in the startup configuration does not match the VLAN database, the domain name and VTP mode and configuration for the first 1005 VLANs use the VLAN database information.
- In VTP versions 1 and 2, if VTP mode is server, the domain name and VLAN configuration for only the first 1005 VLANs use the VLAN database information. VTP version 3 also supports VLANs 1006 to 4096.

Token Ring VLANs

Although the switch does not support Token Ring connections, a remote device such as a Catalyst 6500 series switch with Token Ring connections could be managed from one of the supported switches. Switches running VTP Version 2 advertise information about these Token Ring VLANs:

- Token Ring TrBRF VLANs

- Token Ring TrCRF VLANs

For more information on configuring Token Ring VLANs, see the *Catalyst 6500 Series Software Configuration Guide*.

Normal-Range VLAN Configuration Guidelines

Follow these guidelines when creating and modifying normal-range VLANs in your network:

- The switch supports 1005 VLANs in VTP client, server, and transparent modes.
- Normal-range VLANs are identified with a number between 1 and 1001. VLAN numbers 1002 through 1005 are reserved for Token Ring and FDDI VLANs.
- VLAN configuration for VLANs 1 to 1005 are always saved in the VLAN database. If the VTP mode is transparent, VTP and VLAN configuration are also saved in the switch running configuration file.
- With VTP versions 1 and 2, the switch supports VLAN IDs 1006 through 4096 only in VTP transparent mode (VTP disabled). These are extended-range VLANs and configuration options are limited. Extended-range VLANs created in VTP transparent mode are not saved in the VLAN database and are not propagated. VTP version 3 supports extended range VLAN (VLANs 1006 to 4096) database propagation. If extended VLANs are configured, you cannot convert from VTP version 3 to version 1 or 2. See Creating an Extended-Range VLAN, page 303.
- Before you can create a VLAN, the switch must be in VTP server mode or VTP transparent mode. If the switch is a VTP server, you must define a VTP domain or VTP will not function.
- The switch does not support Token Ring or FDDI media. The switch does not forward FDDI, FDDI-Net, TrCRF, or TrBRF traffic, but it does propagate the VLAN configuration through VTP.
- The switch supports 128 spanning-tree instances. If a switch has more active VLANs than supported spanning-tree instances, spanning tree can be enabled on 128 VLANs and is disabled on the remaining VLANs. If you have already used all available spanning-tree instances on a switch, adding another VLAN anywhere in the VTP domain creates a VLAN on that switch that is not running spanning-tree. If you have the default allowed list on the trunk ports of that switch (which is to allow all VLANs), the new VLAN is carried on all trunk ports. Depending on the topology of the network, this could create a loop in the new VLAN that would not be broken, particularly if there are several adjacent switches that all have run out of spanning-tree instances. You can prevent this possibility by setting allowed lists on the trunk ports of switches that have used up their allocation of spanning-tree instances.

  If the number of VLANs on the switch exceeds the number of supported spanning-tree instances, we recommend that you configure the IEEE 802.1s Multiple STP (MSTP) on your switch to map multiple VLANs to a single spanning-tree instance. For more information about MSTP, see Configuring MSTP, page 351

Default Ethernet VLAN Configuration

Note: The switch supports Ethernet interfaces exclusively. Because FDDI and Token Ring VLANs are not locally supported, you only configure FDDI and Token Ring media-specific characteristics for VTP global advertisements to other switches.

Table 34 Ethernet VLAN Defaults and Ranges

Parameter	Default	Range
VLAN ID	1	1 to 4096. **Note:** Extended-range VLANs (VLAN IDs 1006 to 4096) are only saved in the VLAN database in VTP version 3.
VLAN name	*VLANxxxx*, where *xxxx* represents four numeric digits (including leading zeros) equal to the VLAN ID number	No range
IEEE 802.10 SAID	100001 (100000 plus the VLAN ID)	1 to 4294967294
MTU size	1500	1500 to 18190
Translational bridge 1	0	0 to 1005
Translational bridge 2	0	0 to 1005
VLAN state	active	active, suspend
Remote SPAN	disabled	enabled, disabled

Ethernet VLANs

Each Ethernet VLAN in the VLAN database has a unique, 4-digit ID that can be a number from 1 to 1001. VLAN IDs 1002 to 1005 are reserved for Token Ring and FDDI VLANs. To create a normal-range VLAN to be added to the VLAN database, assign a number and name to the VLAN.

Note: With VTP version 1 and 2, if the switch is in VTP transparent mode, you can assign VLAN IDs greater than 1006, but they are not added to the VLAN database. See Creating an Extended-Range VLAN, page 303.

For the list of default parameters that are assigned when you add a VLAN, see Normal-Range VLANs, page 291.

VLAN Removal

When you delete a VLAN from a switch that is in VTP server mode, the VLAN is removed from the VLAN database for all switches in the VTP domain. When you delete a VLAN from a switch that is in VTP transparent mode, the VLAN is deleted only on that specific switch.

You cannot delete the default VLANs for the different media types: Ethernet VLAN 1 and FDDI or Token Ring VLANs 1002 to 1005.

Caution: When you delete a VLAN, any ports assigned to that VLAN become inactive. They remain associated with the VLAN (and thus inactive) until you assign them to a new VLAN.

Static-Access Ports for a VLAN

You can assign a static-access port to a VLAN without having VTP globally propagate VLAN configuration information by disabling VTP (VTP transparent mode).

If you are assigning a port on a cluster member switch to a VLAN, first use the **rcommand** privileged EXEC command to log in to the cluster member switch.

Note: If you assign an interface to a VLAN that does not exist, the new VLAN is created. (See Creating or Modifying an Ethernet VLAN, page 302.)

Extended-Range VLANs

With VTP version 1 and version 2, when the switch is in VTP transparent mode (VTP disabled), you can create extended-range VLANs (in the range 1006 to 4096). VTP version supports extended-range VLANs in server or transparent move. Extended-range VLANs enable service providers to extend their infrastructure to a greater number of customers. The extended-range VLAN IDs are allowed for any switchport commands that allow VLAN IDs.

With VTP version 1 or 2, extended-range VLAN configurations are not stored in the VLAN database, but because VTP mode is transparent, they are stored in the switch running configuration file, and you can save the configuration in the startup configuration file by using the **copy running-config startup-config** privileged EXEC command. Extended-range VLANs created in VTP version 3 are stored in the VLAN database.

Default VLAN Configuration

See Table 34 on page 294 for the default configuration for Ethernet VLANs. You can change only the MTU size, private VLAN, and the remote SPAN configuration state on extended-range VLANs; all other characteristics must remain at the default state.

Extended-Range VLAN Configuration Guidelines

Follow these guidelines when creating extended-range VLANs:

- VLAN IDs in the extended range are not saved in the VLAN database and are not recognized by VTP unless the switch is running VTP version 3.
- You cannot include extended-range VLANs in the pruning eligible range.
- In VTP version 1 and 2, a switch must be in VTP transparent mode when you create extended-range VLANs. If VTP mode is server or client, an error message is generated, and the extended-range VLAN is rejected. VTP version 3 supports extended VLANs in server and transparent modes.
- For VTP version 1 or 2, you can set the VTP mode to transparent in global configuration mode. See Adding a VTP Client Switch to a VTP Domain, page 321. You should save this configuration to the startup configuration so that the switch boots up in VTP transparent mode. Otherwise, you lose the extended-range VLAN configuration if the switch resets. If you create extended-range VLANs in VTP version 3, you cannot convert to VTP version 1 or 2.
- STP is enabled by default on extended-range VLANs, but you can disable it by using the **no spanning-tree vlan** *vlan-id* global configuration command. When the maximum number of spanning-tree instances are on the switch, spanning tree is disabled on any newly created VLANs. If the number of VLANs on the switch exceeds the maximum number of spanning-tree instances, we recommend that you configure the IEEE 802.1s Multiple STP (MSTP) on your switch to map multiple VLANs to a single spanning-tree instance.
- Each routed port on the switch creates an internal VLAN for its use. These internal VLANs use extended-range VLAN numbers, and the internal VLAN ID cannot be used for an extended-range VLAN. If you try to create an extended-range VLAN with a VLAN ID that is already allocated as an internal VLAN, an error message is generated, and the command is rejected.
  - Because internal VLAN IDs are in the lower part of the extended range, we recommend that you create extended-range VLANs beginning from the highest number (4096) and moving to the lowest (1006) to reduce the possibility of using an internal VLAN ID.
  - Before configuring extended-range VLANs, enter the **show vlan internal usage** privileged EXEC command to see which VLANs have been allocated as internal VLANs.
  - If necessary, you can shut down the routed port assigned to the internal VLAN, which frees up the internal VLAN, and then create the extended-range VLAN and re-enable the port, which then uses another VLAN as its internal VLAN. See Creating an Extended-Range VLAN with an Internal VLAN ID, page 304.

- Although the switch supports a total of 1005 (normal-range and extended-range) VLANs, the number of routed ports, SVIs, and other configured features affects the use of the switch hardware. If you try to create an extended-range VLAN and there are not enough hardware resources available, an error message is generated, and the extended-range VLAN is rejected.

VLAN Trunks

Trunking Overview

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

You can configure a trunk on a single Ethernet interface or on an EtherChannel bundle.

Ethernet trunk interfaces support different trunking modes (see Table 35 on page 296). You can set an interface as trunking or nontrunking or to negotiate trunking with the neighboring interface. To autonegotiate trunking, the interfaces must be in the same VTP domain.

Trunk negotiation is managed by the Dynamic Trunking Protocol (DTP), which is a Point-to-Point Protocol. However, some internetworking devices might forward DTP frames improperly, which could cause misconfigurations.

To avoid this, you should configure interfaces connected to devices that do not support DTP to not forward DTP frames, that is, to turn off DTP.

- If you do not intend to trunk across those links, use the **switchport mode access** interface configuration command to disable trunking.

- To enable trunking to a device that does not support DTP, use the **switchport mode trunk** and **switchport nonegotiate** interface configuration commands to cause the interface to become a trunk but to not generate DTP frames.

Table 35 Layer 2 Interface Modes

Mode	Function
switchport mode access	Puts the interface (access port) into permanent nontrunking mode and negotiates to convert the link into a nontrunk link. The interface becomes a nontrunk interface regardless of whether or not the neighboring interface is a trunk interface.
switchport mode dynamic auto	Makes the interface able to convert the link to a trunk link. The interface becomes a trunk interface if the neighboring interface is set to *trunk* or *desirable* mode. The default switch port mode for all Ethernet interfaces is dynamic auto.
switchport mode dynamic desirable	Makes the interface actively attempt to convert the link to a trunk link. The interface becomes a trunk interface if the neighboring interface is set to *trunk*, *desirable*, or *auto* mode.
switchport mode trunk	Puts the interface into permanent trunking mode and negotiates to convert the neighboring link into a trunk link. The interface becomes a trunk interface even if the neighboring interface is not a trunk interface.
switchport nonegotiate	Prevents the interface from generating DTP frames. You can use this command only when the interface switchport mode is access or trunk. You must manually configure the neighboring interface as a trunk interface to establish a trunk link.

IEEE 802.1Q Configuration Guidelines

The IEEE 802.1Q trunks impose these restrictions on the trunking strategy for a network:

- In a network of Cisco switches connected through IEEE 802.1Q trunks, the switches maintain one spanning-tree instance for each VLAN allowed on the trunks. Non-Cisco devices might support one spanning-tree instance for all VLANs.

  When you connect a Cisco switch to a non-Cisco device through an IEEE 802.1Q trunk, the Cisco switch combines the spanning-tree instance of the VLAN of the trunk with the spanning-tree instance of the non-Cisco IEEE 802.1Q switch. However, spanning-tree information for each VLAN is maintained by Cisco switches separated by a cloud of non-Cisco IEEE 802.1Q switches. The non-Cisco IEEE 802.1Q cloud separating the Cisco switches is treated as a single trunk link between the switches.
- Make sure the native VLAN for an IEEE 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 IEEE 802.1Q trunk without disabling spanning tree on every VLAN in the network can potentially cause spanning-tree loops. We recommend that you leave spanning tree enabled on the native VLAN of an IEEE 802.1Q trunk or disable spanning tree on every VLAN in the network. Make sure your network is loop-free before you disable spanning tree.

Default Layer 2 Ethernet Interface VLAN Settings

Feature	Default Setting
Interface mode	switchport mode dynamic auto
Allowed VLAN range	VLANs 1 to 4096
VLAN range eligible for pruning	VLANs 2 to 1001
Default VLAN (for access ports)	VLAN 1
Native VLAN (for IEEE 802.1Q trunks)	VLAN 1

Ethernet Interface as a Trunk Port

Because trunk ports send and receive VTP advertisements, to use VTP you must ensure that at least one trunk port is configured on the switch and that this trunk port is connected to the trunk port of a second switch. Otherwise, the switch cannot receive any VTP advertisements.

Note: By default, an interface is in Layer 2 mode. The default mode for Layer 2 interfaces is **switchport mode dynamic auto**. If the neighboring interface supports trunking and is configured to allow trunking, the link is a Layer 2 trunk or, if the interface is in Layer 3 mode, it becomes a Layer 2 trunk when you enter the **switchport** interface configuration command.

Trunking Interaction with Other Features

Trunking interacts with other features in these ways:

- A trunk port cannot be a secure port.
- A trunk port cannot be a tunnel port.
- Trunk ports can be grouped into EtherChannel port groups, but all trunks in the group must have the same configuration. When a group is first created, all ports follow the parameters set for the first port to be added to the group. If you change the configuration of one of these parameters, the switch propagates the setting you entered to all ports in the group:
  - Allowed-VLAN list.
  - STP port priority for each VLAN.

  - STP Port Fast setting.
  - Trunk status. If one port in a port group ceases to be a trunk, all ports cease to be trunks.
- We recommend that you configure no more than 24 trunk ports in PVST mode and no more than 40 trunk ports in MST mode.
- If you try to enable IEEE 802.1x on a trunk port, an error message appears, and IEEE 802.1x is not enabled. If you try to change the mode of an IEEE 802.1x-enabled port to trunk, the port mode is not changed.
- A port in dynamic mode can negotiate with its neighbor to become a trunk port. If you try to enable IEEE 802.1x on a dynamic port, an error message appears, and IEEE 802.1x is not enabled. If you try to change the mode of an IEEE 802.1x-enabled port to dynamic, the port mode is not changed.

Allowed VLANs on a Trunk

By default, a trunk port sends traffic to and receives traffic from all VLANs. All VLAN IDs, 1 to 4096, are allowed on each trunk. However, you can remove VLANs from the allowed list, preventing traffic from those VLANs from passing over the trunk. To restrict the traffic a trunk carries, use the **switchport trunk allowed vlan remove** *vlan-list* interface configuration command to remove specific VLANs from the allowed list.

Note: VLAN 1 is the default VLAN on all trunk ports in all Cisco switches, and it has previously been a requirement that VLAN 1 always be enabled on every trunk link. You can use the VLAN 1 minimization feature to disable VLAN 1 on any individual VLAN trunk link so that no user traffic (including spanning-tree advertisements) is sent or received on VLAN 1.

To reduce the risk of spanning-tree loops or storms, you can disable VLAN 1 on any individual VLAN trunk port by removing VLAN 1 from the allowed list. When you remove VLAN 1 from a trunk port, the interface continues to send and receive management traffic, for example, Cisco Discovery Protocol (CDP), Port Aggregation Protocol (PAgP), Link Aggregation Control Protocol (LACP), DTP, and VTP in VLAN 1.

If a trunk port with VLAN 1 disabled is converted to a nontrunk port, it is added to the access VLAN. If the access VLAN is set to 1, the port will be added to VLAN 1, regardless of the **switchport trunk allowed** setting. The same situation applies for any VLAN that has been disabled on the port.

A trunk port can become a member of a VLAN if the VLAN is enabled, if VTP knows of the VLAN, and if the VLAN is in the allowed list for the port. When VTP detects a newly enabled VLAN and the VLAN is in the allowed list for a trunk port, the trunk port automatically becomes a member of the enabled VLAN. When VTP detects a new VLAN and the VLAN is not in the allowed list for a trunk port, the trunk port does not become a member of the new VLAN.

Native VLAN for Untagged Traffic

A trunk port configured with IEEE 802.1Q tagging can receive both tagged and untagged traffic. By default, the switch forwards untagged traffic in the native VLAN configured for the port. The native VLAN is VLAN 1 by default.

Note: The native VLAN can be assigned any VLAN ID.

For information about IEEE 802.1Q configuration issues, see IEEE 802.1Q Configuration Guidelines, page 296.

Load Sharing Using Trunk Ports

Load sharing divides the bandwidth supplied by parallel trunks connecting switches. To avoid loops, STP normally blocks all but one parallel link between switches. Using load sharing, you divide the traffic between the links according to which VLAN the traffic belongs.

You configure load sharing on trunk ports by using STP port priorities or STP path costs. For load sharing using STP port priorities, both load-sharing links must be connected to the same switch. For load sharing using STP path costs, each load-sharing link can be connected to the same switch or to two different switches.

Load Sharing Using STP Port Priorities

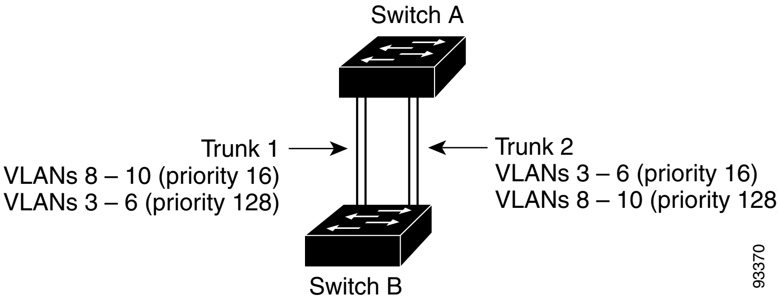
When two ports on the same switch form a loop, the switch uses the STP port priority to decide which port is enabled and which port is in a blocking state. You can set the priorities on a parallel trunk port so that the port carries all the traffic for a given VLAN. The trunk port with the higher priority (lower values) for a VLAN is forwarding traffic for that VLAN. The trunk port with the lower priority (higher values) for the same VLAN remains in a blocking state for that VLAN. One trunk port sends or receives all traffic for the VLAN.

Figure 30 on page 299 shows two trunks connecting supported switches. In this example, the switches are configured as follows:

- VLANs 8 through 10 are assigned a port priority of 16 on Trunk 1.
- VLANs 3 through 6 retain the default port priority of 128 on Trunk 1.
- VLANs 3 through 6 are assigned a port priority of 16 on Trunk 2.
- VLANs 8 through 10 retain the default port priority of 128 on Trunk 2.

In this way, Trunk 1 carries traffic for VLANs 8 through 10, and Trunk 2 carries traffic for VLANs 3 through 6. If the active trunk fails, the trunk with the lower priority takes over and carries the traffic for all of the VLANs. No duplication of traffic occurs over any trunk port.

Figure 30 Load Sharing by Using STP Port Priorities


Switch A
Trunk 1
VLANs 8 – 10 (priority 16)
VLANs 3 – 6 (priority 128)
Trunk 2
VLANs 3 – 6 (priority 16)
VLANs 8 – 10 (priority 128)
Switch B
93370

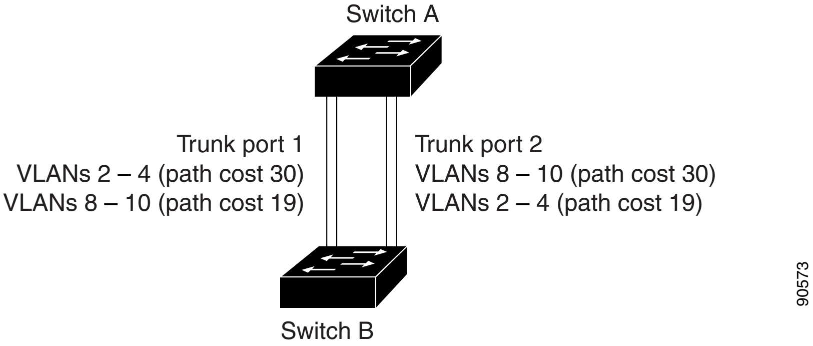
Load Sharing Using STP Path Cost

You can configure parallel trunks to share VLAN traffic by setting different path costs on a trunk and associating the path costs with different sets of VLANs, blocking different ports for different VLANs. The VLANs keep the traffic separate and maintain redundancy in the event of a lost link.

In Figure 31 on page 300, Trunk ports 1 and 2 are configured as 100BASE-T ports. These VLAN path costs are assigned:

- VLANs 2 through 4 are assigned a path cost of 30 on Trunk port 1.
- VLANs 8 through 10 retain the default 100BASE-T path cost on Trunk port 1 of 19.
- VLANs 8 through 10 are assigned a path cost of 30 on Trunk port 2.
- VLANs 2 through 4 retain the default 100BASE-T path cost on Trunk port 2 of 19.

Figure 31 Load-Sharing Trunks with Traffic Distributed by Path Cost


Switch A
Trunk port 1
VLANs 2 – 4 (path cost 30)
VLANs 8 – 10 (path cost 19)
Trunk port 2
VLANs 8 – 10 (path cost 30)
VLANs 2 – 4 (path cost 19)
Switch B
90573

See Configuring Load Sharing Using STP Path Cost, page 306.

VMPS

The VLAN Query Protocol (VQP) is used to support dynamic-access ports, which are not permanently assigned to a VLAN, but give VLAN assignments based on the MAC source addresses seen on the port. Each time an unknown MAC address is seen, the switch sends a VQP query to a remote VMPS; the query includes the newly seen MAC address and the port on which it was seen. The VMPS responds with a VLAN assignment for the port. The switch cannot be a VMPS server but can act as a client to the VMPS and communicate with it through VQP.

Each time the client switch receives the MAC address of a new host, it sends a VQP query to the VMPS. When the VMPS receives this query, it searches its database for a MAC-address-to-VLAN mapping. The server response is based on this mapping and whether or not the server is in open or secure mode. In secure mode, the server shuts down the port when an illegal host is detected. In open mode, the server simply denies the host access to the port.

If the port is currently *unassigned* (that is, it does not yet have a VLAN assignment), the VMPS provides one of these responses:

- If the host is allowed on the port, the VMPS sends the client a *vlan-assignment* response containing the assigned VLAN name and allowing access to the host.
- If the host is not allowed on the port and the VMPS is in open mode, the VMPS sends an *access-denied* response.
- If the VLAN is not allowed on the port and the VMPS is in secure mode, the VMPS sends a *port-shutdown* response.

If the port already has a VLAN assignment, the VMPS provides one of these responses:

- If the VLAN in the database matches the current VLAN on the port, the VMPS sends a *success* response, allowing access to the host.
- If the VLAN in the database does not match the current VLAN on the port and active hosts exist on the port, the VMPS sends an *access-denied* or a *port-shutdown* response, depending on the secure mode of the VMPS.

If the switch receives an *access-denied* response from the VMPS, it continues to block traffic to and from the host MAC address. The switch continues to monitor the packets directed to the port and sends a query to the VMPS when it identifies a new host address. If the switch receives a *port-shutdown* response from the VMPS, it disables the port. The port must be manually reenabled by using Network Assistant, the CLI or SNMP.

Dynamic-Access Port VLAN Membership

A dynamic-access port can belong to only one VLAN with an ID from 1 to 4096. When the link comes up, the switch does not forward traffic to or from this port until the VMPS provides the VLAN assignment. The VMPS receives the source MAC address from the first packet of a new host connected to the dynamic-access port and attempts to match the MAC address to a VLAN in the VMPS database.

If there is a match, the VMPS sends the VLAN number for that port. If the client switch was not previously configured, it uses the domain name from the first VTP packet it receives on its trunk port from the VMPS. If the client switch was previously configured, it includes its domain name in the query packet to the VMPS to obtain its VLAN number. The VMPS verifies that the domain name in the packet matches its own domain name before accepting the request and responds to the client with the assigned VLAN number for the client. If there is no match, the 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-access port if they are all in the same VLAN; however, the VMPS shuts down a dynamic-access port if more than 20 hosts are active on the port.

If the link goes down on a dynamic-access port, the port returns to an isolated state and does not belong to a VLAN. Any hosts that come online through the port are checked again through the VQP with the VMPS before the port is assigned to a VLAN.

Dynamic-access ports can be used for direct host connections, or they can connect to a network. A maximum of 20 MAC addresses are allowed per port on the switch. A dynamic-access port can belong to only one VLAN at a time, but the VLAN can change over time, depending on the MAC addresses seen.

Default VMPS Client Settings

Feature	Default Setting
VMPS domain server	None
VMPS reconfirm interval	60 minutes
VMPS server retry count	3
Dynamic-access ports	None configured

VMPS Configuration Guidelines

These guidelines and restrictions apply to dynamic-access port VLAN membership:

- You should configure the VMPS before you configure ports as dynamic-access ports.
- When you configure a port as a dynamic-access port, the spanning-tree Port Fast feature is automatically enabled for that port. The Port Fast mode accelerates the process of bringing the port into the forwarding state.
- IEEE 802.1x ports cannot be configured as dynamic-access ports. If you try to enable IEEE 802.1x on a dynamic-access (VQP) port, an error message appears, and IEEE 802.1x is not enabled. If you try to change an IEEE 802.1x-enabled port to dynamic VLAN assignment, an error message appears, and the VLAN configuration is not changed.
- Trunk ports cannot be dynamic-access ports, but you can enter the **switchport access vlan dynamic** interface configuration command for a trunk port. In this case, the switch retains the setting and applies it if the port is later configured as an access port.

  You must turn off trunking on the port before the dynamic-access setting takes effect.
- Dynamic-access ports cannot be monitor ports.
- Secure ports cannot be dynamic-access ports. You must disable port security on a port before it becomes dynamic.
- Private VLAN ports cannot be dynamic-access ports.
- Dynamic-access ports cannot be members of an EtherChannel group.
- Port channels cannot be configured as dynamic-access ports.

- A dynamic-access port can participate in fallback bridging.
- The VTP management domain of the VMPS client and the VMPS server must be the same.
- The VLAN configured on the VMPS server should not be a voice VLAN.

VMPS Reconfirmation Interval

VMPS clients periodically reconfirm the VLAN membership information received from the VMPS.You can set the number of minutes after which reconfirmation occurs.

If you are configuring a member switch in a cluster, this parameter must be equal to or greater than the reconfirmation setting on the command switch. You must also first use the **rcommand** privileged EXEC command to log in to the member switch.

Dynamic-Access Port VLAN Membership

The VMPS shuts down a dynamic-access port under these conditions:

- The VMPS is in secure mode, and it does not allow the host to connect to the port. The VMPS shuts down the port to prevent the host from connecting to the network.
- More than 20 active hosts reside on a dynamic-access port.

To reenable a disabled dynamic-access port, enter the **shutdown** interface configuration command followed by the **no shutdown** interface configuration command.

How to Configure VLANs

Creating or Modifying an Ethernet VLAN

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**vlan** *vlan-id*	Enters a VLAN ID, and enters VLAN configuration mode. **Note:** The available VLAN ID range for this command is 1 to 4096. For information about adding VLAN IDs greater than 1005 (extended-range VLANs), see Creating an Extended-Range VLAN, page 303.
3.	**name** *vlan-name*	(Optional) Enters a name for the VLAN. If no name is entered for the VLAN, the default is to append the *vlan-id* with leading zeros to the word VLAN. For example, VLAN0004 is a default VLAN name for VLAN 4.
4.	**mtu** *mtu-size*	(Optional) Changes the MTU size (or other VLAN characteristic).
5.	**remote-span**	(Optional) Configures the VLAN as the RSPAN VLAN for a remote SPAN session. **Note:** For more information on remote SPAN, see Configuring SPAN and RSPAN, page 491
6.	**end**	Returns to privileged EXEC mode.

Deleting a VLAN

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**no vlan** *vlan-id*	Removes the VLAN by entering the VLAN ID.
3.	**end**	Returns to privileged EXEC mode.

Assigning Static-Access Ports to a VLAN

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode
2.	**interface** *interface-id*	Enters the interface to be added to the VLAN.
3.	**switchport mode access**	Defines the VLAN membership mode for the port (Layer 2 access port).
4.	**switchport access vlan** *vlan-id*	Assigns the port to a VLAN. Valid VLAN IDs are 1 to 4096.
5.	**end**	Returns to privileged EXEC mode.

Creating an Extended-Range VLAN

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**vtp mode transparent**	Configures the switch for VTP transparent mode and disables VTP. **Note:** This step is not required for VTP version 3.
3.	**vlan** *vlan-id*	Enters an extended-range VLAN ID and enters VLAN configuration mode. The range is 1006 to 4096.
4.	**mtu** *mtu-size*	(Optional) Modifies the VLAN by changing the MTU size. **Note:** Although all VLAN commands appear in the CLI help, only the **mtu** *mtu-size*, **private-vlan**, and **remote-span** commands are supported for extended-range VLANs.
5.	**remote-span**	(Optional) Configures the VLAN as the RSPAN VLAN. See Configuring a VLAN as an RSPAN VLAN, page 505.
6.	**end**	Returns to privileged EXEC mode.

Creating an Extended-Range VLAN with an Internal VLAN ID

	Command	Purpose
1.	**show vlan internal usage**	Displays the VLAN IDs being used internally by the switch. If the VLAN ID that you want to use is an internal VLAN, the display shows the routed port that is using the VLAN ID. Enter that port number in Step 3.
2.	**configure terminal**	Enters global configuration mode.
3.	**interface** *interface-id*	Specifies the interface ID for the routed port that is using the VLAN ID, and enters interface configuration mode.
4.	**shutdown**	Shuts down the port to free the internal VLAN ID.
5.	**exit**	Returns to global configuration mode.
6.	**vtp mode transparent**	Sets the VTP mode to transparent for creating extended-range VLANs. **Note:** This step is not required for VTP version 3.
7.	**vlan** *vlan-id*	Enters the new extended-range VLAN ID, and enters VLAN configuration mode.
8.	**exit**	Exits from VLAN configuration mode, and returns to global configuration mode.
9.	**interface** *interface-id*	Specifies the interface ID for the routed port that you shut down in Step 4, and enters interface configuration mode.
10.	**no shutdown**	Reenables the routed port. It will be assigned a new internal VLAN ID.
11.	**end**	Returns to privileged EXEC mode.

Configuring an Ethernet Interface as a Trunk Port

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the port to be configured for trunking, and enters interface configuration mode.
3.	**switchport mode** {**dynamic** {**auto** | **desirable**} | **trunk**}	Configures the interface as a Layer 2 trunk (required only if the interface is a Layer 2 access port or tunnel port or to specify the trunking mode). ■ **dynamic auto**—Sets the interface to a trunk link if the neighboring interface is set to trunk or desirable mode. This is the default. ■ **dynamic desirable**—Sets the interface to a trunk link if the neighboring interface is set to trunk, desirable, or auto mode. ■ **trunk**—Sets the interface in permanent trunking mode and negotiate to convert the link to a trunk link even if the neighboring interface is not a trunk interface.
4.	**switchport access vlan** *vlan-id*	(Optional) Specifies the default VLAN, which is used if the interface stops trunking.
5.	**switchport trunk native vlan** *vlan-id*	Specifies the native VLAN for IEEE 802.1Q trunks.
6.	**end**	Returns to privileged EXEC mode.

Defining the Allowed VLANs on a Trunk

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the port to be configured, and enters interface configuration mode.
3.	**switchport mode trunk**	Configures the interface as a VLAN trunk port.
4.	**switchport trunk allowed vlan** {**add** | **all** | **except** | **remove**} *vlan-list*	(Optional) Configures the list of VLANs allowed on the trunk.
5.	**end**	Returns to privileged EXEC mode.

Changing the Pruning-Eligible List

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Selects the trunk port for which VLANs should be pruned, and enters interface configuration mode.
3.	**switchport trunk pruning vlan** {**add** | **except** | **none** | **remove**} *vlan-list* [,*vlan*[,*vlan*[,,,]]	Configures the list of VLANs allowed to be pruned from the trunk. (See VTP Pruning, page 318.)
4.	**end**	Returns to privileged EXEC mode.

Configuring the Native VLAN for Untagged Traffic

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Defines the interface that is configured as the IEEE 802.1Q trunk, and enters interface configuration mode.
3.	**switchport trunk native vlan** *vlan-id*	Configures the VLAN that is sending and receiving untagged traffic on the trunk port.
4.	**end**	Returns to privileged EXEC mode.

Load Sharing Using STP Port Priorities

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode on Switch A.
2.	**vtp domain** *domain-name*	Configures a VTP administrative domain. The domain name can be 1 to 32 characters.
3.	**vtp mode server**	Configures Switch A as the VTP server.
4.	**end**	Returns to privileged EXEC mode.
5.	**show vtp status**	Verifies the VTP configuration on both Switch A and Switch B.
6.	**show vlan**	Verifies that the VLANs exist in the database on Switch A.
7.	**configure terminal**	Enters global configuration mode.
8.	**interface** *interface-id_1*	Defines the interface to be configured as a trunk, and enters interface configuration mode.
9.	**switchport mode trunk**	Configures the port as a trunk port.
10.	**end**	Returns to privileged EXEC mode.
11.	**show interfaces** *interface-id_1* **switchport**	Verifes the VLAN configuration.
12.	Repeat Steps 7 through 10 on Switch A for a second port in the switch.	
13.	Repeat Steps 7 through 10 on Switch B to configure the trunk ports that connect to the trunk ports configured on Switch A.	
14.	**show vlan**	When the trunk links come up, VTP passes the VTP and VLAN information to Switch B. Verifies that Switch B has learned the VLAN configuration.
15.	**configure terminal**	Enters global configuration mode on Switch A.
16.	**interface** *interface-id_1*	Defines the interface to set the STP port priority, and enters interface configuration mode.
17.	**spanning-tree vlan 8-10 port-priority 16**	Assigns the port priority of 16 for VLANs 8 through 10.
18.	**exit**	Returns to global configuration mode.
19.	**interface** *interface-id_2*	Defines the interface to set the STP port priority, and enters interface configuration mode.
20.	**spanning-tree vlan 3-6 port-priority 16**	Assigns the port priority of 16 for VLANs 3 through 6.
21.	**end**	Returns to privileged EXEC mode.

Configuring Load Sharing Using STP Path Cost

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode on Switch A.
2.	**interface** *interface-id_1*	Defines the interface to be configured as a trunk, and enters interface configuration mode.
3.	**switchport mode trunk**	Configures the port as a trunk port.
4.	**exit**	Returns to global configuration mode.
5.		Repeat Steps 2 through 4 on a second interface in Switch A.
6.	**end**	Returns to privileged EXEC mode.

	Command	Purpose
7.	**show running-config**	Verifies your entries. In the display, make sure that the interfaces are configured as trunk ports.
8.	**show vlan**	When the trunk links come up, Switch A receives the VTP information from the other switches. Verifies that Switch A has learned the VLAN configuration.
9.	**configure terminal**	Enters global configuration mode.
10.	**interface** *interface-id_1*	Defines the interface on which to set the STP cost, and enters interface configuration mode.
11.	**spanning-tree vlan 2-4 cost 30**	Sets the spanning-tree path cost to 30 for VLANs 2 through 4.
12.	**end**	Returns to global configuration mode.
13.	Repeat Steps 9 through 12 on the other configured trunk interface on Switch A, and set the spanning-tree path cost to 30 for VLANs 8, 9, and 10.	
14.	**exit**	Returns to privileged EXEC mode.
15.	**show running-config**	Verifies your entries. In the display, verify that the path costs are set correctly for both trunk interfaces.

Configuring the VMPS Client

You configure dynamic VLANs by using the VMPS (VLAN Membership Policy Server). The switch can be a VMPS client; it cannot be a VMPS server.

Entering the IP Address of the VMPS

Before You Begin

- You must first enter the IP address of the server to configure the switch as a client.
- You must have IP connectivity to the VMPS for dynamic-access ports to work. You can test for IP connectivity by pinging the IP address of the VMPS and verifying that you get a response.
- If the VMPS is being defined for a cluster of switches, enter the address on the command switch.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**vmps server** *ipaddress* **primary**	Enters the IP address of the switch acting as the primary VMPS server.
3.	**vmps server** *ipaddress*	(Optional) Enters the IP address of the switch acting as a secondary VMPS server. You can enter up to three secondary server addresses.
4.	**vmps reconfirm**	(Optional) Reconfirms dynamic-access port VLAN membership.
5.	**vmps retry** *count*	(Optional) Changes the retry count.
6.	**end**	Returns to privileged EXEC mode.

Configuring Dynamic-Access Ports on VMPS Clients

Before You Begin

If you are configuring a port on a cluster member switch as a dynamic-access port, first use the **rcommand** privileged EXEC command to log in to the cluster member switch.

Caution: Dynamic-access port VLAN membership is for end stations or hubs connected to end stations. Connecting dynamic-access ports to other switches can cause a loss of connectivity.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the switch port that is connected to the end station, and enters interface configuration mode.
3.	**switchport mode access**	Sets the port to access mode.
4.	**switchport access vlan dynamic**	Configures the port as eligible for dynamic VLAN membership. The dynamic-access port must be connected to an end station.
5.	**end**	Returns to privileged EXEC mode.

Monitoring and Maintaining VLANs

Command	Purpose
copy running-config startup config	Saves your entries in the configuration file ■ To save an extended-range VLAN configuration, you need to save the VTP transparent mode configuration and the extended-range VLAN configuration in the switch startup configuration file. Otherwise, if the switch resets, it will default to VTP server mode, and the extended-range VLAN IDs will not be saved. ■ This step is not required for VTP version 3 because VLANs are saved in the VLAN database.
show interfaces *interface-id* **switchport**	Displays the switch port configuration of the interface.
show interfaces *interface-id* **trunk**	Displays the trunk configuration of the interface.
show running-config interface *interface-id*	Verifies the VLAN membership mode of the interface.
show vmps	Verifies your VMPS entries.
show vlan	Verifies your VLAN entries.

Configuration Examples for Configuring VLANs

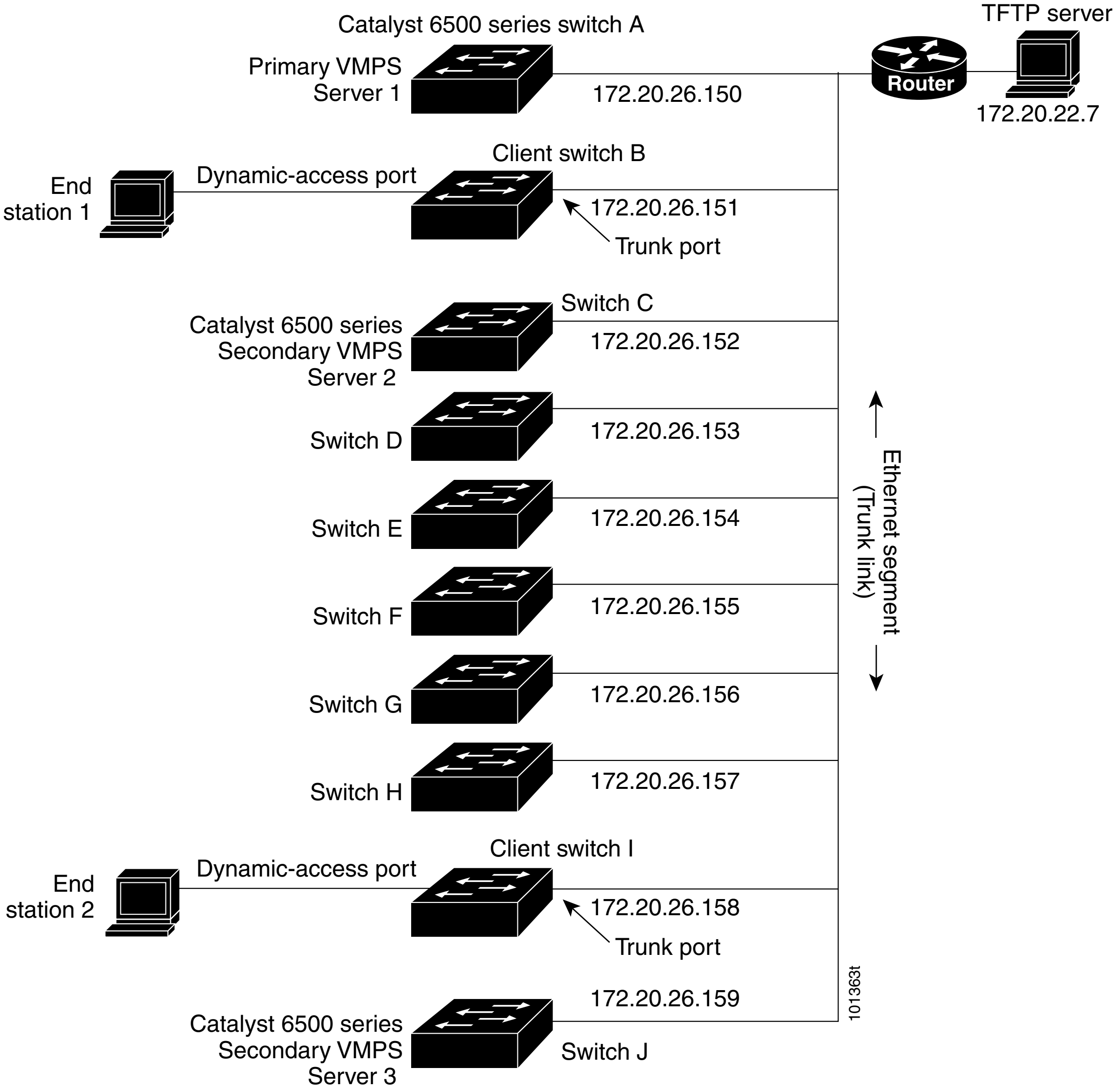
VMPS Network: Example

Figure 32 on page 309 shows a network with a VMPS server switch and VMPS client switches with dynamic-access ports. In this example, these assumptions apply:

- The VMPS server and the VMPS client are separate switches.
- The Catalyst 6500 series Switch A is the primary VMPS server.

- The Catalyst 6500 series Switch C and Switch J are secondary VMPS servers.
- End stations are connected to the clients, Switch B and Switch I.
- The database configuration file is stored on the TFTP server with the IP address 172.20.22.7.

Figure 32 Dynamic Port VLAN Membership Configuration



Configuring a VLAN: Example

This example shows how to create Ethernet VLAN 20, name it *test20,* and add it to the VLAN database:

```
Switch# configure terminal
Switch(config)# vlan 20
Switch(config-vlan)# name test20
Switch(config-vlan)# end
```

Configuring an Access Port in a VLAN: Example

This example shows how to configure a port as an access port in VLAN 2:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 2
Switch(config-if)# end
```

Configuring an Extended-Range VLAN: Example

This example shows how to create a new extended-range VLAN with all default characteristics:

```
Switch(config)# vtp mode transparent
Switch(config)# vlan 2000
Switch(config-vlan)# end
Switch# copy running-config startup config
```

Configuring a Trunk Port: Example

This example shows how to configure a port as an IEEE 802.1Q trunk. The example assumes that the neighbor interface is configured to support IEEE 802.1Q trunking.

```
Switch(config)# interface GigabitEthernet1/18
Switch(config-if)# switchport mode dynamic desirable
Switch(config-if)# end
```

Removing a VLAN: Example

This example shows how to remove VLAN 2 from the allowed VLAN list on a port:

```
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# switchport trunk allowed vlan remove 2
Switch(config-if)# end
```

Show VMPS Output: Example

This is an example of output for the **show vmps** privileged EXEC command:

```
Switch# show vmps
VQP Client Status:
--------------------
VMPS VQP Version:   1
Reconfirm Interval: 60 min
Server Retry Count: 3
VMPS domain server: 172.20.128.86 (primary, current)
                    172.20.128.87

Reconfirmation status
---------------------
VMPS Action:         other
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Configuring VTP

Prerequisites for Configuring VTP

- When you configure VTP, you must configure a trunk port so that the switch can send and receive VTP advertisements to and from other switches in the domain.
- Before adding a VTP client switch to a VTP domain, always verify that its VTP configuration revision number is *lower* than the configuration revision number of the other switches in the VTP domain. Switches in a VTP domain always use the VLAN configuration of the switch with the highest VTP configuration revision number. If you add a switch that has a revision number higher than the revision number in the VTP domain, it can erase all VLAN information from the VTP server and VTP domain. See Adding a VTP Client Switch to a VTP Domain, page 323 for the procedure for verifying and resetting the VTP configuration revision number.

Restrictions for Configuring VTP

- VTP version 1 and VTP version 2 are not interoperable on switches in the same VTP domain. Do not enable VTP version 2 unless every switch in the VTP domain supports version 2.
- In VTP versions 1 and 2, when you configure extended-range VLANs on the switch, the switch must be in VTP transparent mode. VTP version 3 also supports creating extended-range VLANs in client or server mode.

Information About Configuring VTP

VTP

A VLAN Trunking Protocol (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 cause several problems, such as duplicate VLAN names, incorrect VLAN-type specifications, and security violations.

Before you create VLANs, you must decide whether to use VTP in your network. Using VTP, you can make configuration changes centrally on one or more switches and have those changes automatically communicated to all the other switches in the network. Without VTP, you cannot send information about VLANs to other switches.

VTP is designed to work in an environment where updates are made on a single switch and are sent through VTP to other switches in the domain. It does not work well in a situation where multiple updates to the VLAN database occur simultaneously on switches in the same domain, which would result in an inconsistency in the VLAN database.

The switch supports 1005 VLANs, but the number of configured features affects the usage of the switch hardware. If the switch is notified by VTP of a new VLAN and the switch is already using the maximum available hardware resources, it sends a message that there are not enough hardware resources available and shuts down the VLAN. The output of the **show vlan** user EXEC command shows the VLAN in a suspended state.

VTP version 1 and version 2 support only normal-range VLANs (VLAN IDs 1 to 1005). VTP version 3 supports the entire VLAN range (VLANs 1 to 4096). Extended range VLANs (VLANs 1006 to 4096) are supported only in VTP version 3. You cannot convert from VTP version 3 to VTP version 2 if extended VLANs are configured in the domain.

VTP Domain

A VTP domain (also called a VLAN management domain) consists of one switch or several interconnected switches under the same administrative responsibility sharing the same VTP domain name. A switch can be in only one VTP domain. You make global VLAN configuration changes for the domain.

By default, the switch is in the VTP no-management-domain state until it receives an advertisement for a domain over a trunk link (a link that carries the traffic of multiple VLANs) or until you configure a domain name. Until the management domain name is specified or learned, you cannot create or modify VLANs on a VTP server, and VLAN information is not propagated over the network.

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 then ignores advertisements with a different domain name or an earlier configuration revision number.

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 sent over all IEEE trunk connections, including IEEE 802.1Q. VTP dynamically maps VLANs with unique names and internal index associates across multiple LAN types. Mapping eliminates excessive device administration required from network administrators.

If you configure a switch for VTP transparent mode, you can create and modify VLANs, but the changes are not sent to other switches in the domain, and they affect only the individual switch. However, configuration changes made when the switch is in this mode are saved in the switch running configuration and can be saved to the switch startup configuration file.

For domain name and password configuration guidelines, see VTP Configuration Guidelines, page 320.

VTP Modes

Table 36 VTP Modes

VTP Mode	Description
VTP server	In VTP server mode, you can create, modify, and delete VLANs, and specify other configuration parameters (such as the VTP version) for the entire VTP domain. VTP servers advertise their VLAN configurations to other switches in the same VTP domain and synchronize their VLAN configurations with other switches based on advertisements received over trunk links. VTP server is the default mode. **Note:** In VTP server mode, VLAN configurations are saved in NVRAM. If the switch detects a failure while writing a configuration to NVRAM, VTP mode automatically changes from server mode to client mode. If this happens, the switch cannot be returned to VTP server mode until the NVRAM is functioning.
VTP client	A VTP client behaves like a VTP server and transmits and receives VTP updates on its trunks, but you cannot create, change, or delete VLANs on a VTP client. VLANs are configured on another switch in the domain that is in server mode. In VTP versions 1 and 2, in VTP client mode, VLAN configurations are not saved in NVRAM. In VTP version 3, VLAN configurations are saved in NVRAM in client mode.
VTP 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 or version 3, transparent switches do forward VTP advertisements that they receive from other switches through their trunk interfaces. You can create, modify, and delete VLANs on a switch in VTP transparent mode. In VTP versions 1 and 2, the switch must be in VTP transparent mode when you create extended-range VLANs. VTP version 3 also supports creating extended-range VLANs in client or server mode. When the switch is in VTP transparent mode, the VTP and VLAN configurations are saved in NVRAM, but they are not advertised to other switches. In this mode, VTP mode and domain name are saved in the switch running configuration, and you can save this information in the switch startup configuration file by using the **copy running-config startup-config** privileged EXEC command.
VTP off	A switch in VTP off mode functions in the same manner as a VTP transparent switch, except that it does not forward VTP advertisements on trunks.

VTP Mode Guidelines

- For VTP version 1 and version 2, if extended-range VLANs are configured on the switch, you cannot change VTP mode to client or server. You receive an error message, and the configuration is not allowed. VTP version 1 and version 2 do not propagate configuration information for extended range VLANs (VLANs 1006 to 4096). You must manually configure these VLANs on each device.

  Note: For VTP version 1 and 2, before you create extended-range VLANs (VLAN IDs 1006 to 4096), you must set VTP mode to transparent by using the **vtp mode transparent** global configuration command. Save this configuration to the startup configuration so that the switch starts in VTP transparent mode. Otherwise, you lose the extended-range VLAN configuration if the switch resets and boots up in VTP server mode (the default).

- VTP version 3 supports extended-range VLANs. If extended VLANs are configured, you cannot convert from VTP version 3 to VTP version 2.

- If you configure the switch for VTP client mode, the switch does not create the VLAN database file (vlan.dat). If the switch is then powered off, it resets the VTP configuration to the default. To keep the VTP configuration with VTP client mode after the switch restarts, you must first configure the VTP domain name before the VTP mode.

- When a switch is in VTP server mode, you can change the VLAN configuration and have it propagated throughout the network.
- When a switch is in VTP client mode, you cannot change its VLAN configuration. The client switch receives VTP updates from a VTP server in the VTP domain and then modifies its configuration accordingly.
- When you configure the switch for VTP transparent mode, VTP is disabled on the switch. The switch does not send VTP updates and does not act on VTP updates received from other switches. However, a VTP transparent switch running VTP version 2 does forward received VTP advertisements on its trunk links.
- VTP off mode is the same as VTP transparent mode except that VTP advertisements are not forwarded.

Caution: If all switches are operating in VTP client mode, do not configure a VTP domain name. If you do, it is impossible to make changes to the VLAN configuration of that domain. Therefore, make sure you configure at least one switch as a VTP server.

VTP Advertisements

Each switch in the VTP domain sends periodic global configuration advertisements from each trunk port to a reserved multicast address. Neighboring switches receive these advertisements and update their VTP and VLAN configurations as necessary.

VTP advertisements distribute this global domain information:

- VTP domain name
- VTP configuration revision number
- Update identity and update timestamp
- MD5 digest VLAN configuration, including maximum transmission unit (MTU) size for each VLAN
- Frame format

VTP advertisements distribute this VLAN information for each configured VLAN:

- VLAN IDs (IEEE 802.1Q)
- VLAN name
- VLAN type
- VLAN state
- Additional VLAN configuration information specific to the VLAN type

In VTP version 3, VTP advertisements also include the primary server ID, an instance number, and a start index.

VTP Version 2

If you use VTP in your network, you must decide which version of VTP to use. By default, VTP operates in version 1.

VTP version 2 supports these features that are not supported in version 1:

- Token Ring support—VTP version 2 supports Token Ring Bridge Relay Function (TrBRF) and Token Ring Concentrator Relay Function (TrCRF) VLANs. For more information about Token Ring VLANs, see Normal-Range VLANs, page 291.
- 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 when the switch is operating in VTP server mode.

- 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. Although VTP version 2 supports only one domain, a VTP version 2 transparent switch forwards a message only when the domain name matches.

- 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 MD5 digest on a received VTP message is correct, its information is accepted.

VTP Version 3

VTP version 3 supports these features that are not supported in version 1 or version 2:

- Enhanced authentication—You can configure the authentication as **hidden** or **secret**. When **hidden**, the secret key from the password string is saved in the VLAN database file, but it does not appear in plain text in the configuration. Instead, the key associated with the password is saved in hexadecimal format in the running configuration. You must reenter the password if you enter a takeover command in the domain. When you enter the **secret** keyword, you can directly configure the password secret key.

- Support for extended range VLAN (VLANs 1006 to 4096) database propagation. VTP versions 1 and 2 propagate only VLANs 1 to 1005. If extended VLANs are configured, you cannot convert from VTP version 3 to version 1 or 2.

  VTP pruning still applies only to VLANs 1 to 1005, and VLANs 1002 to 1005 are still reserved and cannot be modified.

- Support for any database in a domain. In addition to propagating VTP information, version 3 can propagate Multiple Spanning Tree (MST) protocol database information. A separate instance of the VTP protocol runs for each application that uses VTP.

- VTP primary server and VTP secondary servers. A VTP primary server updates the database information and sends updates that are honored by all devices in the system. A VTP secondary server can only back up the updated VTP configurations received from the primary server to its NVRAM.

  By default, all devices come up as secondary servers. You can enter the **vtp primary** privileged EXEC command to specify a primary server. Primary server status is only needed for database updates when the administrator issues a takeover message in the domain. You can have a working VTP domain without any primary servers. Primary server status is lost if the device reloads or domain parameters change, even when a password is configured on the switch.

- The option to turn VTP on or off on a per-trunk (per-port) basis. You can enable or disable VTP per port by entering the [**no**] **vtp** interface configuration command. When you disable VTP on trunking ports, all VTP instances for that port are disabled. You cannot set VTP to *off* for the MST database and *on* for the VLAN database on the same port.

  When you globally set VTP mode to off, it applies to all the trunking ports in the system. However, you can specify on or off on a per-VTP instance basis. For example, you can configure the switch as a VTP server for the VLAN database but with VTP *off* for the MST database.

VTP Version Guidelines

Follow these guidelines when deciding which VTP version to implement:

- All switches in a VTP domain must have the same domain name, but they do not need to run the same VTP version.

- A VTP version 2-capable switch can operate in the same VTP domain as a switch running VTP version 1 if version 2 is disabled on the version 2-capable switch (version 2 is disabled by default).

- If a switch running VTP version 1 but capable of running VTP version 2 receives VTP version 3 advertisements, it automatically moves to VTP version 2.

- If a switch running VTP version 3 is connected to a switch running VTP version 1, the VTP version 1 switch moves to VTP version 2, and the VTP version 3 switch sends scaled-down versions of the VTP packets so that the VTP version 2 switch can update its database.
- A switch running VTP version 3 cannot move to version 1 or 2 if it has extended VLANs.
- 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 version 2 on a switch, all of the version-2-capable switches in the domain enable version 2. If there is a version 1-only switch, it does not exchange VTP information with switches that have version 2 enabled.
- We recommend placing VTP version 1 and 2 switches at the edge of the network because they do not forward VTP version 3 advertisements.
- If there are TrBRF and TrCRF Token Ring networks in your environment, you must enable VTP version 2 or version 3 for Token Ring VLAN switching to function properly. To run Token Ring and Token Ring-Net, disable VTP version 2.
- VTP version 1 and version 2 do not propagate configuration information for extended range VLANs (VLANs 1006 to 4096). You must configure these VLANs manually on each device. VTP version 3 supports extended-range VLANs. You cannot convert from VTP version 3 to VTP version 2 if extended VLANs are configured.
- When a VTP version 3 device trunk port receives messages from a VTP version 2 device, it sends a scaled-down version of the VLAN database on that particular trunk in VTP version 2 format. A VTP version 3 device does not send VTP version 2-formatted packets on a trunk unless it first receives VTP version 2 packets on that trunk port.
- When a VTP version 3 device detects a VTP version 2 device on a trunk port, it continues to send VTP version 3 packets, in addition to VTP version 2 packets, to allow both kinds of neighbors to coexist on the same trunk.
- A VTP version 3 device does not accept configuration information from a VTP version 2 or version 1 device.
- Two VTP version 3 regions can only communicate in transparent mode over a VTP version 1 or version 2 region.
- Devices that are only VTP version 1 capable cannot interoperate with VTP version 3 devices.
- VTP version 2 and version 3 are disabled by default.
- When you enable VTP version 2 on a switch, every VTP version 2-capable switch in the VTP domain enables version 2. To enable VTP version 3, you must manually configure it on each switch.
- With VTP versions 1 and 2, you can configure the version only on switches in VTP server or transparent mode. If a switch is running VTP version 3, you can change to version 2 when the switch is in client mode if no extended VLANs exist, no private VLANs exist, and no hidden password was configured.

Caution: In VTP version 3, both the primary and secondary servers can exist on an instance in the domain.

VTP Pruning

VTP pruning increases network available bandwidth by restricting flooded traffic to those trunk links that the traffic must use to reach the destination devices. Without VTP pruning, a switch floods broadcast, multicast, and unknown unicast traffic across all trunk links within a VTP domain even though receiving switches might discard them. VTP pruning is disabled by default.

VTP pruning blocks unneeded flooded traffic to VLANs on trunk ports that are included in the pruning-eligible list. Only VLANs included in the pruning-eligible list can be pruned. By default, VLANs 2 through 1001 are pruning eligible switch trunk ports. If the VLANs are configured as pruning-ineligible, the flooding continues. VTP pruning is supported in all VTP versions.

Figure 33 on page 319 shows a switched network without VTP pruning enabled. Port 1 on Switch A and Port 2 on Switch D are assigned to the Red VLAN. If a broadcast is sent from the host connected to Switch A, Switch A floods the broadcast and every switch in the network receives it, even though Switches C, E, and F have no ports in the Red VLAN.

Figure 33 Flooding Traffic without VTP Pruning

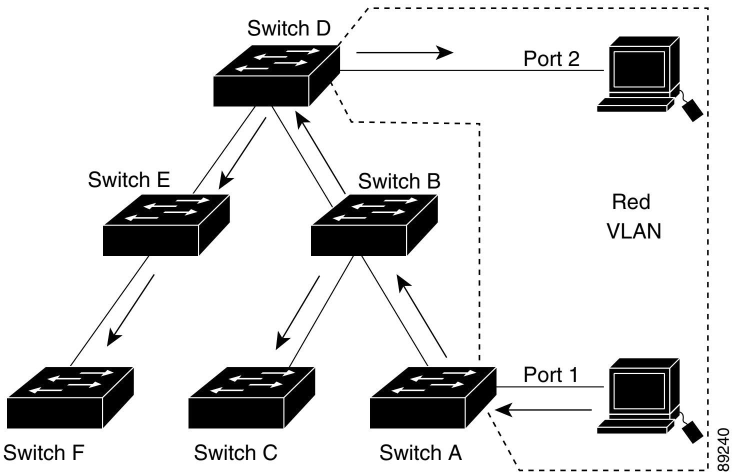
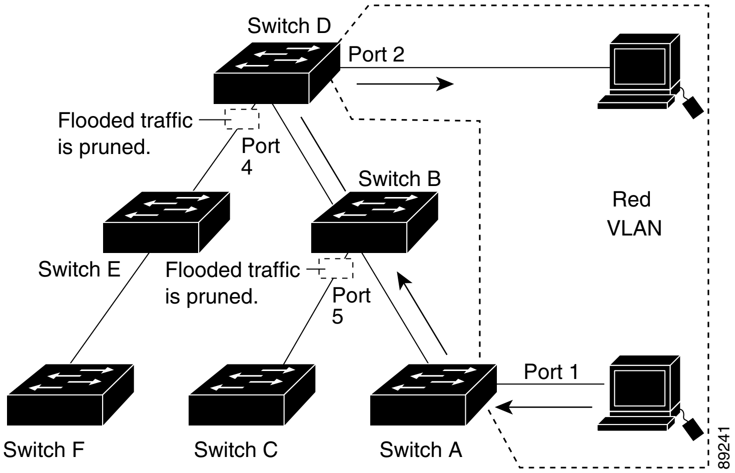


Figure 34 on page 319 shows a switched network with VTP pruning enabled. The broadcast traffic from Switch A is not forwarded to Switches C, E, and F because traffic for the Red VLAN has been pruned on the links shown (Port 5 on Switch B and Port 4 on Switch D).

Figure 34 Optimized Flooded Traffic with VTP Pruning



With VTP versions 1 and 2, enabling VTP pruning on a VTP server enables pruning for the entire management domain. Making VLANs pruning-eligible or pruning-ineligible affects pruning eligibility for those VLANs on that trunk only (not on all switches in the VTP domain). In VTP version 3, you must manually enable pruning on each switch in the domain.

See Enabling VTP Pruning, page 323. VTP pruning takes effect several seconds after you enable it. VTP pruning does not prune traffic from VLANs that are pruning-ineligible. VLAN 1 and VLANs 1002 to 1005 are always pruning-ineligible; traffic from these VLANs cannot be pruned. Extended-range VLANs (VLAN IDs higher than 1005) are also pruning-ineligible.

VTP pruning is not designed to function in VTP transparent mode. If one or more switches in the network are in VTP transparent mode, you should do one of these:

- Turn off VTP pruning in the entire network.

- Turn off VTP pruning by making all VLANs on the trunk of the switch upstream to the VTP transparent switch pruning ineligible.

To configure VTP pruning on an interface, use the **switchport trunk pruning vlan** interface configuration command. VTP pruning operates when an interface is trunking. You can set VLAN pruning-eligibility, whether or not VTP pruning is enabled for the VTP domain, whether or not any given VLAN exists, and whether or not the interface is currently trunking.

Default VTP Settings

Feature	Default Setting
VTP domain name	Null.
VTP mode (VTP version 1 and version 2)	Server.
VTP mode (VTP version 3)	The mode is the same as the mode in VTP version 1 or 2 before conversion to version 3.
VTP version	Version 1.
MST database mode	Transparent.
VTP version 3 server type	Secondary.
VTP password	None.
VTP pruning	Disabled.

VTP Configuration Guidelines

You use the **vtp** global configuration command to set the VTP password, the version, the VTP filename, the interface providing updated VTP information, the domain name, and the mode, and to disable or enable pruning. The VTP information is saved in the VTP VLAN database. When VTP mode is transparent, the VTP domain name and mode are also saved in the switch running configuration file, and you can save it in the switch startup configuration file by entering the **copy running-config startup-config** privileged EXEC command. You must use this command if you want to save VTP mode as transparent if the switch resets.

When you save VTP information in the switch startup configuration file and restart the switch, the configuration is selected as follows:

- If the VTP mode is transparent in both the startup configuration and the VLAN database and the VTP domain name from the VLAN database matches that in the startup configuration file, the VLAN database is ignored (cleared). The VTP and VLAN configurations in the startup configuration file are used. The VLAN database revision number remains unchanged in the VLAN database.

- If the VTP mode or the domain name in the startup configuration do not match the VLAN database, the domain name and the VTP mode and configuration for the first 1005 VLANs use the VLAN database information.

Domain Names

When configuring VTP for the first time, you must always assign a domain name. You must configure all switches in the VTP domain with the same domain name. Switches in VTP transparent mode do not exchange VTP messages with other switches, and you do not need to configure a VTP domain name for them.

Note: If NVRAM and DRAM storage is sufficient, all switches in a VTP domain should be in VTP server mode.

Caution: Do not configure a VTP domain if all switches are operating in VTP client mode. If you configure the domain, it is impossible to make changes to the VLAN configuration of that domain. Make sure that you configure at least one switch in the VTP domain for VTP server mode.

Passwords

You can configure a password for the VTP domain, but it is not required. If you do configure a domain password, all domain switches must share the same password and you must configure the password on each switch in the management domain. Switches without a password or with the wrong password reject VTP advertisements.

If you configure a VTP password for a domain, a switch that is booted without a VTP configuration does not accept VTP advertisements until you configure it with the correct password. After the configuration, the switch accepts the next VTP advertisement that uses the same password and domain name in the advertisement.

If you are adding a new switch to an existing network with VTP capability, the new switch learns the domain name only after the applicable password has been configured on it.

Caution: When you configure a VTP domain password, the management domain does not function properly if you do not assign a management domain password to each switch in the domain.

Adding a VTP Client Switch to a VTP Domain

Before adding a VTP client to a VTP domain, always verify that its VTP configuration revision number is *lower* than the configuration revision number of the other switches in the VTP domain. Switches in a VTP domain always use the VLAN configuration of the switch with the highest VTP configuration revision number. With VTP versions 1 and 2, adding a switch that has a revision number higher than the revision number in the VTP domain can erase all VLAN information from the VTP server and VTP domain. With VTP version 3, the VLAN information is not erased.

How to Configure VTP

Configuring VTP Domain and Parameters

Before You Begin

You should configure the VTP domain before configuring other VTP parameters.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**vtp domain** *domain-name*	Configures the VTP administrative-domain name. The name can be 1 to 32 characters. All switches operating in VTP server or client mode under the same administrative responsibility must be configured with the same domain name. This command is optional for modes other than server mode. VTP server mode requires a domain name. If the switch has a trunk connection to a VTP domain, the switch learns the domain name from the VTP server in the domain.
3.	**vtp mode** {**client | server** | **transparent** | **off**} {**vlan** | **mst** | **unknown**}	Configures the switch for VTP mode (client, server, transparent, or off). (Optional) Database parameters: ■ **vlan**–The VLAN database is the default if none are configured. ■ **mst**–The multiple spanning tree (MST) database. ■ **unknown**–An unknown database type.

	Command	Purpose
4.	**vtp password** *password*	(Optional) Sets the password for the VTP domain. The password can be 8 to 64 characters. If you configure a VTP password, the VTP domain does not function properly if you do not assign the same password to each switch in the domain. See Configuring a VTP Version 3 Password, page 322 for options available with VTP version 3.
1.	**vtp primary-server** [**vlan** | **mst**] [**force**]	(Optional) Changes the operational state of a switch from a secondary server (the default) to a primary server and advertise the configuration to the domain. If the switch password is configured as **hidden**, you are prompted to reenter the password. ■ **vlan**–Selects the VLAN database as the takeover feature. This is the default. ■ **mst**–Selects the multiple spanning tree (MST) database as the takeover feature. ■ **force**–Overwrites the configuration of any conflicting servers. If you do not enter **force**, you are prompted for confirmation before the takeover.
2.	**end**	Returns to privileged EXEC mode.
3.	**show vtp status**	Verifies your entries in the *VTP Operating Mode* and the *VTP Domain Name* fields of the display.
4.	**copy running-config startup-config**	(Optional) Saves the configuration in the startup configuration file. **Note:** Only VTP mode and domain name are saved in the switch running configuration and can be copied to the startup configuration file.

Configuring a VTP Version 3 Password

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**vtp password** *password* [**hidden** | **secret**]	(Optional) Sets the password for the VTP domain. The password can be 8 to 64 characters. ■ (Optional) **hidden**–Ensures that the secret key generated from the password string is saved in the nvam:vlan.dat file. If you configure a takeover by configuring a VTP primary server, you are prompted to reenter the password. ■ (Optional) **secret**–Directly configures the password. The secret password must contain 32 hexadecimal characters.
3.	**end**	Returns to privileged EXEC mode.
4.	**show vtp password**	Verifies your entries.

Enabling the VTP Version

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**vtp version** {**1** | **2** | **3**}	Enables the VTP version on the switch. The default is VTP version 1.
3.	**end**	Returns to privileged EXEC mode.
4.	**show vtp status**	Verifies that the configured VTP version is enabled.
5.	**copy running-config startup-config**	(Optional) Saves the configuration in the startup configuration file.

Enabling VTP Pruning

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**vtp pruning**	Enables pruning in the VTP administrative domain. By default, pruning is disabled. You need to enable pruning on only one switch in VTP server mode.
3.	**end**	Returns to privileged EXEC mode.
4.	**show vtp status**	Verifies your entries in the *VTP Pruning Mode* field of the display.

Configuring VTP on a Per-Port Basis

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Identifies an interface, and enters interface configuration mode.
3.	**vtp**	Enables VTP on the specified port.
4.	**end**	Returns to privileged EXEC mode.
5.	**show running-config interface** *interface-id*	Verifies the change to the port.
6.	**show vtp status**	Verifies the configuration.

Adding a VTP Client Switch to a VTP Domain

Before You Begin

Before adding a VTP client to a VTP domain, always verify that its VTP configuration revision number is *lower* than the configuration revision number of the other switches in the VTP domain. Switches in a VTP domain always use the VLAN configuration of the switch with the highest VTP configuration revision number. With VTP versions 1 and 2, adding a switch that has a revision number higher than the revision number in the VTP domain can erase all VLAN information from the VTP server and VTP domain. With VTP version 3, the VLAN information is not erased.

	Command	Purpose
1.	**show vtp status**	Checks the VTP configuration revision number. If the number is 0, add the switch to the VTP domain. If the number is greater than 0, follow these steps: **a.** Write down the domain name. **b.** Write down the configuration revision number. **c.** Continue with the next steps to reset the switch configuration revision number.
2.	**configure terminal**	Enters global configuration mode.
3.	**vtp domain** *domain-name*	Changes the domain name from the original one displayed in Step 1 to a new name.
4.	**end**	Updates VLAN information on the switch and resets configuration revision number to 0.
5.	**show vtp status**	Verifies that the configuration revision number has been reset to 0.
6.	**configure terminal**	Enters global configuration mode.
7.	**vtp domain** *domain-name*	Enters the original domain name on the switch.
8.	**end**	Returns to privileged EXEC mode.
9.	**show vtp status**	(Optional) Verifies that the domain name is the same as in Step 1 and that the configuration revision number is 0.
10.	After resetting the configuration revision number, add the switch to the VTP domain.	

Monitoring and Maintaining VTP

Command	Purpose
show vtp counters	Displays counters about VTP messages that have been sent and received.
show vtp devices [**conflict**]	Displays information about all VTP version 3 devices in the domain. Conflicts are VTP version 3 devices with conflicting primary servers. The **show vtp devices** command does not display information when the switch is in transparent or off mode.
show vtp interface [*interface-id*]	Displays VTP status and configuration for all interfaces or the specified interface.
show vtp password	Displays the VTP password. The form of the password displayed depends on whether or not the **hidden** keyword was entered and if encryption is enabled on the switch.
show vtp status	Displays the VTP switch configuration information.

Configuration Examples for Configuring VTP

Configuring a VTP Server: Example

This example shows how to configure the switch as a VTP server with the domain name *eng_group* and the password *mypassword*:

```
Switch(config)# vtp domain eng_group
Setting VTP domain name to eng_group.
Switch(config)# vtp mode server
Setting device to VTP Server mode for VLANS.
Switch(config)# vtp password mypassword
Setting device VLAN database password to mypassword.
Switch(config)# end
```

Configuring a Hidden VTP Password: Example

This example shows how to configure a hidden password and how it appears:

```
Switch(config)# vtp password mypassword hidden
Generating the secret associated to the password.
Switch(config)# end
Switch# show vtp password
VTP password: 89914640C8D90868B6A0D8103847A733
```

Configuring a VTP Version 3 Primary Server: Example

This example shows how to configure a switch as the primary server for the VLAN database (the default) when a hidden or secret password was configured:

```
Switch# vtp primary vlan
Enter VTP password: mypassword
This switch is becoming Primary server for vlan feature in the VTP  domain

VTP Database Conf Switch ID      Primary Server Revision System Name
------------ ---- -------------- -------------- --------- --------------------
VLANDB       Yes  00d0.00b8.1400=00d0.00b8.1400 1         stp7

Do you want to continue (y/n) [n]? y
```

Additional References for Configuring VTP

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*
VLAN configuration	Configuring VLANs, page 289

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Configuring Voice VLAN

Information About Configuring Voice VLAN

Voice VLAN

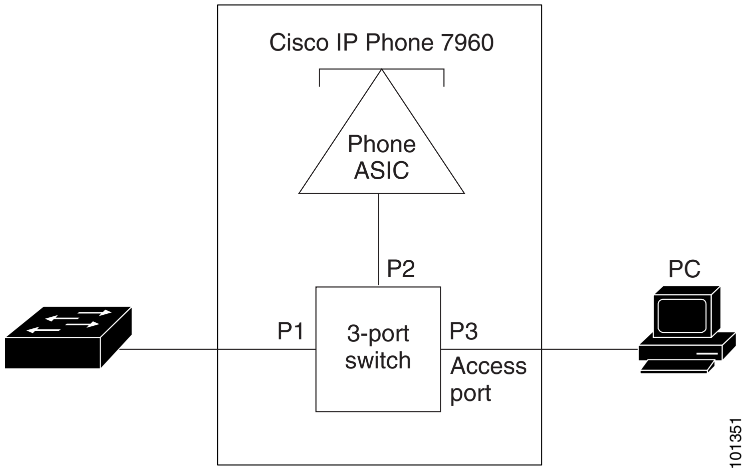
The voice VLAN feature enables access ports to carry IP voice traffic from an IP phone. When the switch is connected to a Cisco 7960 IP Phone, the phone sends voice traffic with Layer 3 IP precedence and Layer 2 class of service (CoS) values, which are both set to 5 by default. Because the sound quality of a Cisco IP phone call can deteriorate if the data is unevenly sent, the switch supports quality of service (QoS) based on IEEE 802.1p CoS. QoS uses classification and scheduling to send network traffic from the switch in a predictable manner. Voice VLAN is referred to as an *auxiliary VLAN* in some switch documentation.

The Cisco 7960 IP Phone is a configurable device, and you can configure it to forward traffic with an IEEE 802.1p priority. You can configure the switch to trust or override the traffic priority assigned by a Cisco IP phone.

The Cisco IP phone contains an integrated three-port 10/100 switch as shown in Figure 35 on page 328. The ports provide dedicated connections to these devices:

- Port 1 connects to the switch or other voice-over-IP (VoIP) device.
- Port 2 is an internal 10/100 interface that carries the IP phone traffic.
- Port 3 (access port) connects to a PC or other device.

Figure 35 Cisco 7960 IP Phone Connected to a Switch



Cisco IP Phone Voice Traffic

You can configure an access port with an attached Cisco IP phone to use one VLAN for voice traffic and another VLAN for data traffic from a device attached to the phone. You can configure access ports on the switch to send Cisco Discovery Protocol (CDP) packets that instruct an attached phone to send voice traffic to the switch in any of these ways:

- In the voice VLAN tagged with a Layer 2 CoS priority value
- In the access VLAN tagged with a Layer 2 CoS priority value
- In the access VLAN, untagged (no Layer 2 CoS priority value)

Note: In all configurations, the voice traffic carries a Layer 3 IP precedence value (the default is 5 for voice traffic and 3 for voice control traffic).

You can configure a port connected to the Cisco IP phone to send CDP packets to the phone to configure the way in which the phone sends voice traffic. The phone can carry voice traffic in IEEE 802.1Q frames for a specified voice VLAN with a Layer 2 CoS value. It can use IEEE 802.1p priority tagging to give voice traffic a higher priority and forward all voice traffic through the native (access) VLAN. The Cisco IP phone can also send untagged voice traffic or use its own configuration to send voice traffic in the access VLAN. In all configurations, the voice traffic carries a Layer 3 IP precedence value (the default is 5).

Cisco IP Phone Data Traffic

The switch can also process tagged data traffic (traffic in IEEE 802.1Q or IEEE 802.1p frame types) from the device attached to the access port on the Cisco IP phone (see Figure 35 on page 328). You can configure Layer 2 access ports on the switch to send CDP packets that instruct the attached phone to configure the phone access port in one of these modes:

- In trusted mode, all traffic received through the access port on the Cisco IP phone passes through the phone unchanged.
- In untrusted mode, all traffic in IEEE 802.1Q or IEEE 802.1p frames received through the access port on the Cisco IP phone receive a configured Layer 2 CoS value. The default Layer 2 CoS value is 0. Untrusted mode is the default.

Note: Untagged traffic from the device attached to the Cisco IP phone passes through the phone unchanged, regardless of the trust state of the access port on the phone.

Default Voice VLAN Configuration

The voice VLAN feature is disabled by default.

When the voice VLAN feature is enabled, all untagged traffic is sent according to the default CoS priority of the port.

The CoS value is not trusted for IEEE 802.1p or IEEE 802.1Q tagged traffic.

Voice VLAN Configuration Guidelines

- Voice VLAN configuration is only supported on switch access ports; voice VLAN configuration is not supported on trunk ports.

  Note: Trunk ports can carry any number of voice VLANs, similar to regular VLANs. The configuration of voice VLANs is not required on trunk ports.

- The voice VLAN should be present and active on the switch for the IP phone to correctly communicate on the voice VLAN. Use the **show vlan** privileged EXEC command to see if the VLAN is present (listed in the display).
- Before you enable voice VLAN, we recommend that you enable QoS on the switch. If you use the auto-QoS feature, these settings are automatically configured. For more information, see Configuring QoS, page 613
- You must enable CDP on the switch port connected to the Cisco IP phone to send the configuration to the phone. (CDP is globally enabled by default on all switch interfaces.)
- The Port Fast feature is automatically enabled when voice VLAN is configured. When you disable voice VLAN, the Port Fast feature is not automatically disabled.
- If the Cisco IP phone and a device attached to the phone are in the same VLAN, they must be in the same IP subnet. These conditions indicate that they are in the same VLAN:
  - They both use IEEE 802.1p or untagged frames.
  - The Cisco IP phone uses IEEE 802.1p frames, and the device uses untagged frames.
  - The Cisco IP phone uses untagged frames, and the device uses IEEE 802.1p frames.
  - The Cisco IP phone uses IEEE 802.1Q frames, and the voice VLAN is the same as the access VLAN.
- The Cisco IP phone and a device attached to the phone cannot communicate if they are in the same VLAN and subnet but use different frame types because traffic in the same subnet is not routed (routing would eliminate the frame type difference).
- You cannot configure static secure MAC addresses in the voice VLAN.
- Voice VLAN ports can also be these port types:
  - Dynamic access port.
  - IEEE 802.1x authenticated port. See Configuring IEEE 802.1x Port-Based Authentication, page 189 for more information.

    If you enable IEEE 802.1x on an access port on which a voice VLAN is configured and to which a Cisco IP phone is connected, the phone loses connectivity to the switch for up to 30 seconds.

  - Protected port.
  - A source or destination port for a SPAN or RSPAN session.
  - Secure port.

When you enable port security on an interface that is also configured with a voice VLAN, you must set the maximum allowed secure addresses on the port to two plus the maximum number of secure addresses allowed on the access VLAN. When the port is connected to a Cisco IP phone, the phone requires up to two MAC addresses. The phone address is learned on the voice VLAN and might also be learned on the access VLAN. Connecting a PC to the phone requires additional MAC addresses.

Port Connection to a Cisco 7960 IP Phone

Because a Cisco 7960 IP Phone also supports a connection to a PC or other device, a port connecting the switch to a Cisco IP phone can carry mixed traffic. You can configure a port to decide how the Cisco IP phone carries voice traffic and data traffic.

Priority of Incoming Data Frames

You can connect a PC or other data device to a Cisco IP phone port. To process tagged data traffic (in IEEE 802.1Q or IEEE 802.1p frames), you can configure the switch to send CDP packets to instruct the phone how to send data packets from the device attached to the access port on the Cisco IP phone. The PC can generate packets with an assigned CoS value. You can configure the phone to not change (trust) or to override (not trust) the priority of frames arriving on the phone port from connected devices.

How to Configure Voice VLAN

Configuring the Priority of Incoming Data Frames

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the interface connected to the Cisco IP phone, and enters interface configuration mode.
3.	**switchport priority extend** {**cos** *value* | **trust**}	Sets the priority of data traffic received from the Cisco IP phone access port: ■ **cos** *value*—Configures the phone to override the priority received from the PC or the attached device with the specified CoS value. The value is a number from 0 to 7, with 7 as the highest priority. The default priority is **cos** 0. ■ **trust**—Configures the phone access port to trust the priority received from the PC or the attached device.
4.	**end**	Returns to privileged EXEC mode.

Monitoring and Maintaining Voice VLAN

Command	Purpose
show interfaces *interface-id* **switchport**	Verifies your entries.
copy running-config startup-config	Saves your entries in the configuration file.

Configuration Examples for Configuring Voice VLAN

Configuring the Cisco IP Phone Priority of Incoming Data Frames: Example

This example shows how to configure a port connected to a Cisco IP phone to not change the priority of frames received from the PC or the attached device:

```
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# switchport priority extend trust
Switch(config-if)# end
```

Additional References for Configuring Voice VLAN

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*
QoS configuration	*Configuring QoS, page 613*
VLAN configuration	*Configuring VLANs, page 289*
IEEE 802.1x authenticated port configuration	Configuring IEEE 802.1x Port-Based Authentication, page 189
Protected port configuration	“Configuring Protected Ports”
Secure port configuration	“Configuring Port Security”

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Configuring STP

Prerequisites for Configuring STP

When you configure VTP, you must configure a trunk port so that the switch can send and receive VTP advertisements to and from other switches in the domain.

For more information, see Configuring an Ethernet Interface as a Trunk Port, page 304.

Restrictions for Configuring STP

- If you are configuring VTP on a cluster member switch to a VLAN, use the **rcommand** privileged EXEC command to log in to the member switch.
- In VTP versions 1 and 2, when you configure extended-range VLANs on the switch, the switch must be in VTP transparent mode. VTP version 3 also supports creating extended-range VLANs in client or server mode.

Information About Configuring STP

This chapter describes how to configure the Spanning Tree Protocol (STP) on port-based VLANs on the switch. The switch can use either the per-VLAN spanning-tree plus (PVST+) protocol based on the IEEE 802.1D standard and Cisco proprietary extensions, or the rapid per-VLAN spanning-tree plus (rapid-PVST+) protocol based on the IEEE 802.1w standard.

STP

STP is a Layer 2 link management protocol that provides path redundancy while preventing loops in the network. For a Layer 2 Ethernet network to function properly, only one active path can exist between any two stations. Multiple active paths among end stations cause loops in the network. If a loop exists in the network, end stations might receive duplicate messages. Switches might also learn end-station MAC addresses on multiple Layer 2 interfaces. These conditions result in an unstable network. Spanning-tree operation is transparent to end stations, which cannot detect whether they are connected to a single LAN segment or a switched LAN of multiple segments.

The STP uses a spanning-tree algorithm to select one switch of a redundantly connected network as the root of the spanning tree. The algorithm calculates the best loop-free path through a switched Layer 2 network by assigning a role to each port based on the role of the port in the active topology:

- Root—A forwarding port elected for the spanning-tree topology
- Designated—A forwarding port elected for every switched LAN segment
- Alternate—A blocked port providing an alternate path to the root bridge in the spanning tree
- Backup—A blocked port in a loopback configuration

The switch that has *all* of its ports as the designated role or as the backup role is the root switch. The switch that has at least *one* of its ports in the designated role is called the designated switch.

Spanning tree forces redundant data paths into a standby (blocked) state. If a network segment in the spanning tree fails and a redundant path exists, the spanning-tree algorithm recalculates the spanning-tree topology and activates the standby path. Switches send and receive spanning-tree frames, called bridge protocol data units (BPDUs), at regular intervals. The switches do not forward these frames but use them to construct a loop-free path. BPDUs contain information about the sending switch and its ports, including switch and MAC addresses, switch priority, port priority, and path cost. Spanning tree uses this information to elect the root switch and root port for the switched network and the root port and designated port for each switched segment.

When two ports on a switch are part of a loop, the spanning-tree port priority and path cost settings control which port is put in the forwarding state and which is put in the blocking state. The spanning-tree port priority value represents the location of a port in the network topology and how well it is located to pass traffic. The path cost value represents the media speed.

Note: The default is for the switch to send keepalive messages (to ensure the connection is up) only on interfaces that do not have small form-factor pluggable (SFP) modules. You can use the [**no**] **keepalive** interface configuration command to change the default for an interface.

Spanning-Tree Topology and BPDUs

The stable, active spanning-tree topology of a switched network is controlled by these elements:

- The unique bridge ID (switch priority and MAC address) associated with each VLAN on each switch.
- The spanning-tree path cost to the root switch.
- The port identifier (port priority and MAC address) associated with each Layer 2 interface.

When the switches in a network are powered up, each functions as the root switch. Each switch sends a configuration BPDU through all of its ports. The BPDUs communicate and compute the spanning-tree topology. Each configuration BPDU contains this information:

- The unique bridge ID of the switch that the sending switch identifies as the root switch
- The spanning-tree path cost to the root
- The bridge ID of the sending switch
- Message age
- The identifier of the sending interface
- Values for the hello, forward delay, and max-age protocol timers

When a switch receives a configuration BPDU that contains *superior* information (lower bridge ID, lower path cost, and so forth), it stores the information for that port. If this BPDU is received on the root port of the switch, the switch also forwards it with an updated message to all attached LANs for which it is the designated switch.

If a switch receives a configuration BPDU that contains *inferior* information to that currently stored for that port, it discards the BPDU. If the switch is a designated switch for the LAN from which the inferior BPDU was received, it sends that LAN a BPDU containing the up-to-date information stored for that port. In this way, inferior information is discarded, and superior information is propagated on the network.

A BPDU exchange results in these actions:

- One switch in the network is elected as the root switch (the logical center of the spanning-tree topology in a switched network).

  For each VLAN, the switch with the highest switch priority (the lowest numerical priority value) is elected as the root switch. If all switches are configured with the default priority (32768), the switch with the lowest MAC address in the VLAN becomes the root switch. The switch priority value occupies the most significant bits of the bridge ID, as shown in Table 37 on page 335.

- A root port is selected for each switch (except the root switch). This port provides the best path (lowest cost) when the switch forwards packets to the root switch.
- The shortest distance to the root switch is calculated for each switch based on the path cost.
- A designated switch for each LAN segment is selected. The designated switch incurs the lowest path cost when forwarding packets from that LAN to the root switch. The port through which the designated switch is attached to the LAN is called the designated port.

All paths that are not needed to reach the root switch from anywhere in the switched network are placed in the spanning-tree blocking mode.

Bridge ID, Switch Priority, and Extended System ID

The IEEE 802.1D standard requires that each switch has an unique bridge identifier (bridge ID), which controls the selection of the root switch. Because each VLAN is considered as a different *logical bridge* with PVST+ and rapid PVST+, the same switch must have a different bridge IDs for each configured VLAN. Each VLAN on the switch has a unique 8-byte bridge ID. The 2 most-significant bytes are used for the switch priority, and the remaining 6 bytes are derived from the switch MAC address.

The switch supports the IEEE 802.1t spanning-tree extensions, and some of the bits previously used for the switch priority are now used as the VLAN identifier. The result is that fewer MAC addresses are reserved for the switch, and a larger range of VLAN IDs can be supported, all while maintaining the uniqueness of the bridge ID. As shown in Table 37 on page 335, the 2 bytes previously used for the switch priority are reallocated into a 4-bit priority value and a 12-bit extended system ID value equal to the VLAN ID.

Table 37 Switch Priority Value and Extended System ID

Switch Priority Value				Extended System ID (Set Equal to the VLAN ID)											
Bit 16	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1
32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1

Spanning tree uses the extended system ID, the switch priority, and the allocated spanning-tree MAC address to make the bridge ID unique for each VLAN.

Support for the extended system ID affects how you manually configure the root switch, the secondary root switch, and the switch priority of a VLAN. For example, when you change the switch priority value, you change the probability that the switch will be elected as the root switch. Configuring a higher value decreases the probability; a lower value increases the probability. For more information, see Configuring the Root Switch, page 346, the Configuring a Secondary Root Switch, page 346, and the Configuring Optional STP Parameters, page 347.

Spanning-Tree Interface States

Propagation delays can occur when protocol information passes through a switched LAN. As a result, topology changes can take place at different times and at different places in a switched network. When an interface transitions directly from nonparticipation in the spanning-tree topology to the forwarding state, it can create temporary data loops. Interfaces must wait for new topology information to propagate through the switched LAN before starting to forward frames. They must allow the frame lifetime to expire for forwarded frames that have used the old topology.

Each Layer 2 interface on a switch using spanning tree exists in one of these states:

- Blocking—The interface does not participate in frame forwarding.
- Listening—The first transitional state after the blocking state when the spanning tree decides that the interface should participate in frame forwarding.

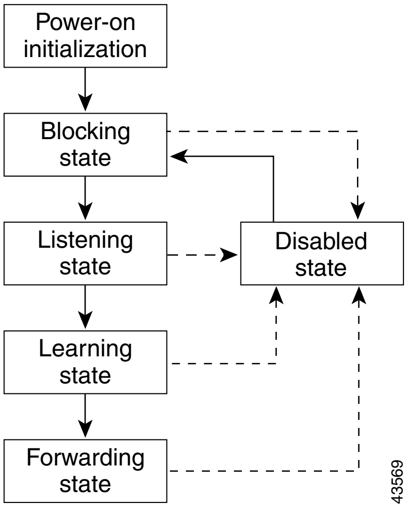
- Learning—The interface prepares to participate in frame forwarding.
- Forwarding—The interface forwards frames.
- Disabled—The interface is not participating in spanning tree because of a shutdown port, no link on the port, or no spanning-tree instance running on the port.

An interface moves through these states:

- 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 36 on page 336 illustrates how an interface moves through the states.

Figure 36 Spanning-Tree Interface States



When you power up the switch, spanning tree is enabled by default, and every interface in the switch, VLAN, or network goes through the blocking state and the transitory states of listening and learning. Spanning tree stabilizes each interface at the forwarding or blocking state.

When the spanning-tree algorithm places a Layer 2 interface in the forwarding state, this process occurs:

1. The interface is in the listening state while spanning tree waits for protocol information to move the interface to the blocking state.
2. While spanning tree waits the forward-delay timer to expire, it moves the interface to the learning state and resets the forward-delay timer.
3. In the learning state, the interface continues to block frame forwarding as the switch learns end-station location information for the forwarding database.

4. When the forward-delay timer expires, spanning tree moves the interface to the forwarding state, where both learning and frame forwarding are enabled.

Blocking State

A Layer 2 interface in the blocking state does not participate in frame forwarding. After initialization, a BPDU is sent to each switch interface. A switch initially functions as the root until it exchanges BPDUs with other switches. This exchange establishes which switch in the network is the root or root switch. If there is only one switch in the network, no exchange occurs, the forward-delay timer expires, and the interface moves to the listening state. An interface always enters the blocking state after switch initialization.

An interface in the blocking state performs these functions:

- Discards frames received on the interface
- Discards frames switched from another interface for forwarding
- Does not learn addresses
- Receives BPDUs

Listening State

The listening state is the first state a Layer 2 interface enters after the blocking state. The interface enters this state when the spanning tree decides that the interface should participate in frame forwarding.

An interface in the listening state performs these functions:

- Discards frames received on the interface
- Discards frames switched from another interface for forwarding
- Does not learn addresses
- Receives BPDUs

Learning State

A Layer 2 interface in the learning state prepares to participate in frame forwarding. The interface enters the learning state from the listening state.

An interface in the learning state performs these functions:

- Discards frames received on the interface
- Discards frames switched from another interface for forwarding
- Learns addresses
- Receives BPDUs

Forwarding State

A Layer 2 interface in the forwarding state forwards frames. The interface enters the forwarding state from the learning state.

An interface in the forwarding state performs these functions:

- Receives and forwards frames received on the interface

- Forwards frames switched from another interface
- Learns addresses
- Receives BPDUs

Disabled State

A Layer 2 interface in the disabled state does not participate in frame forwarding or in the spanning tree. An interface in the disabled state is nonoperational.

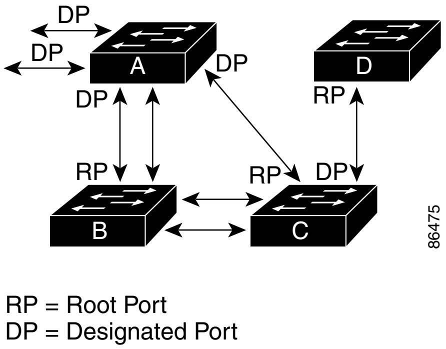
A disabled interface performs these functions:

- Discards frames received on the interface
- Discards frames switched from another interface for forwarding
- Does not learn addresses
- Does not receive BPDUs

How a Switch or Port Becomes the Root Switch or Root Port

If all switches in a network are enabled with default spanning-tree settings, the switch with the lowest MAC address becomes the root switch. In Figure 37 on page 338, Switch A is elected as the root switch because the switch priority of all the switches is set to the default (32768) and Switch A has the lowest MAC address. However, because of traffic patterns, number of forwarding interfaces, or link types, Switch A might not be the ideal root switch. By increasing the priority (lowering the numerical value) of the ideal switch so that it becomes the root switch, you force a spanning-tree recalculation to form a new topology with the ideal switch as the root.

Figure 37 Spanning-Tree Topology



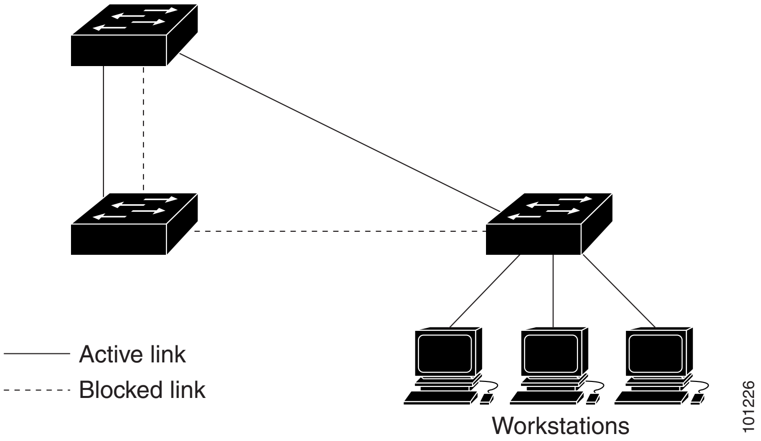
When the spanning-tree topology is calculated based on default parameters, the path between source and destination end stations in a switched network might not be ideal. For instance, connecting higher-speed links to an interface that has a higher number than the root port can cause a root-port change. The goal is to make the fastest link the root port.

For example, assume that one port on Switch B is a Gigabit Ethernet link and that another port on Switch B (a 10/100 link) is the root port. Network traffic might be more efficient over the Gigabit Ethernet link. By changing the spanning-tree port priority on the Gigabit Ethernet port to a higher priority (lower numerical value) than the root port, the Gigabit Ethernet port becomes the new root port.

Spanning Tree and Redundant Connectivity

You can create a redundant backbone with spanning tree by connecting two switch interfaces to another device or to two different devices, as shown in Figure 38 on page 339. Spanning tree automatically disables one interface but enables it if the other one fails. If one link is high-speed and the other is low-speed, the low-speed link is always disabled. If the speeds are the same, the port priority and port ID are added together, and spanning tree disables the link with the lowest value.

Figure 38 Spanning Tree and Redundant Connectivity



You can also create redundant links between switches by using EtherChannel groups. For more information, see Configuring EtherChannels, page 1069

Spanning-Tree Address Management

IEEE 802.1D specifies 17 multicast addresses, ranging from 0x00180C2000000 to 0x0180C2000010, to be used by different bridge protocols. These addresses are static addresses that cannot be removed.

Regardless of the spanning-tree state, each switch receives but does not forward packets destined for addresses between 0x0180C2000000 and 0x0180C200000F.

If spanning tree is enabled, the CPU on the switch receives packets destined for 0x0180C2000000 and 0x0180C2000010. If spanning tree is disabled, the switch forwards those packets as unknown multicast addresses.

Accelerated Aging to Retain Connectivity

The default for aging dynamic addresses is 5 minutes, the default setting of the **mac address-table aging-time** global configuration command. However, a spanning-tree reconfiguration can cause many station locations to change. Because these stations could be unreachable for 5 minutes or more during a reconfiguration, the address-aging time is accelerated so that station addresses can be dropped from the address table and then relearned. The accelerated aging is the same as the forward-delay parameter value (**spanning-tree vlan** *vlan-id* **forward-time** *seconds* global configuration command) when the spanning tree reconfigures.

Because each VLAN is a separate spanning-tree instance, the switch accelerates aging on a per-VLAN basis. A spanning-tree reconfiguration on one VLAN can cause the dynamic addresses learned on that VLAN to be subject to accelerated aging. Dynamic addresses on other VLANs can be unaffected and remain subject to the aging interval entered for the switch.

Spanning-Tree Modes and Protocols

The switch supports these spanning-tree modes and protocols:

- PVST+—This spanning-tree mode is based on the IEEE 802.1D standard and Cisco proprietary extensions. It is the default spanning-tree mode used on all Ethernet port-based VLANs. The PVST+ runs on each VLAN on the switch up to the maximum supported, ensuring that each has a loop-free path through the network.

  The PVST+ provides Layer 2 load balancing for the VLAN on which it runs. You can create different logical topologies by using the VLANs on your network to ensure that all of your links are used but that no one link is oversubscribed. Each instance of PVST+ on a VLAN has a single root switch. This root switch propagates the spanning-tree information associated with that VLAN to all other switches in the network. Because each switch has the same information about the network, this process ensures that the network topology is maintained.

- Rapid PVST+—This spanning-tree mode is the same as PVST+ except that is uses a rapid convergence based on the IEEE 802.1w standard. To provide rapid convergence, the rapid PVST+ immediately deletes dynamically learned MAC address entries on a per-port basis upon receiving a topology change. By contrast, PVST+ uses a short aging time for dynamically learned MAC address entries.

  The rapid PVST+ uses the same configuration as PVST+ (except where noted), and the switch needs only minimal extra configuration. The benefit of rapid PVST+ is that you can migrate a large PVST+ install base to rapid PVST+ without having to learn the complexities of the MSTP configuration and without having to reprovision your network. In rapid-PVST+ mode, each VLAN runs its own spanning-tree instance up to the maximum supported.

- MSTP—This spanning-tree mode is based on the IEEE 802.1s standard. You can map multiple VLANs to the same spanning-tree instance, which reduces the number of spanning-tree instances required to support a large number of VLANs. The MSTP runs on top of the RSTP (based on IEEE 802.1w), which provides for rapid convergence of the spanning tree by eliminating the forward delay and by quickly transitioning root ports and designated ports to the forwarding state. You cannot run MSTP without RSTP.

  The most common initial deployment of MSTP is in the backbone and distribution layers of a Layer 2 switched network. For more information, see Configuring MSTP, page 351

For information about the number of supported spanning-tree instances, see the next section.

Supported Spanning-Tree Instances

In PVST+ or rapid-PVST+ mode, the switch supports up to 128 spanning-tree instances.

In MSTP mode, the switch supports up to 65 MST instances. The number of VLANs that can be mapped to a particular MST instance is unlimited.

For information about how spanning tree interoperates with the VLAN Trunking Protocol (VTP), see Changing the Spanning-Tree Mode, page 345.

Spanning-Tree Interoperability and Backward Compatibility

Table 38Spanning-Tree Interoperability and Backward Compatibility, page 341 lists the interoperability and compatibility among the supported spanning-tree modes in a network.

Table 38 Spanning-Tree Interoperability and Backward Compatibility

	PVST+	MSTP	Rapid PVST+
PVST+	Yes	Yes (with restrictions)	Yes (reverts to PVST+)
MSTP	Yes (with restrictions)	Yes	Yes (reverts to PVST+)
Rapid PVST+	Yes (reverts to PVST+)	Yes (reverts to PVST+)	Yes

In a mixed MSTP and PVST+ network, the common spanning-tree (CST) root must be inside the MST backbone, and a PVST+ switch cannot connect to multiple MST regions.

When a network contains switches running rapid PVST+ and switches running PVST+, we recommend that the rapid-PVST+ switches and PVST+ switches be configured for different spanning-tree instances. In the rapid-PVST+ spanning-tree instances, the root switch must be a rapid-PVST+ switch. In the PVST+ instances, the root switch must be a PVST+ switch. The PVST+ switches should be at the edge of the network.

STP and IEEE 802.1Q Trunks

The IEEE 802.1Q standard for VLAN trunks imposes some limitations on the spanning-tree strategy for a network. The standard requires only one spanning-tree instance for *all* VLANs allowed on the trunks. However, in a network of Cisco switches connected through IEEE 802.1Q trunks, the switches maintain one spanning-tree instance for *each* VLAN allowed on the trunks.

When you connect a Cisco switch to a non-Cisco device through an IEEE 802.1Q trunk, the Cisco switch uses PVST+ to provide spanning-tree interoperability. If rapid PVST+ is enabled, the switch uses it instead of PVST+. The switch combines the spanning-tree instance of the IEEE 802.1Q VLAN of the trunk with the spanning-tree instance of the non-Cisco IEEE 802.1Q switch.

However, all PVST+ or rapid-PVST+ information is maintained by Cisco switches separated by a cloud of non-Cisco IEEE 802.1Q switches. The non-Cisco IEEE 802.1Q cloud separating the Cisco switches is treated as a single trunk link between the switches.

PVST+ is automatically enabled on IEEE 802.1Q trunks, and no user configuration is required. The external spanning-tree behavior on access ports is not affected by PVST+.

VLAN-Bridge Spanning Tree

Cisco VLAN-bridge spanning tree is used with the fallback bridging feature (bridge groups), which forwards non-IP protocols such as DECnet between two or more VLAN bridge domains or routed ports. The VLAN-bridge spanning tree allows the bridge groups to form a spanning tree on top of the individual VLAN spanning trees to prevent loops from forming if there are multiple connections among VLANs. It also prevents the individual spanning trees from the VLANs being bridged from collapsing into a single spanning tree.

To support VLAN-bridge spanning tree, some of the spanning-tree timers are increased.

Default Spanning-Tree Settings

Table 39 Default Spanning-Tree Settings

Feature	Default Setting
Enable state	Enabled on VLAN 1.
Spanning-tree mode	PVST+. (Rapid PVST+ and MSTP are disabled.)
Switch priority	32768.
Spanning-tree port priority (configurable on a per-interface basis)	128.
Spanning-tree port cost (configurable on a per-interface basis)	1000 Mb/s: 4. 100 Mb/s: 19. 10 Mb/s: 100.
Spanning-tree VLAN port priority (configurable on a per-VLAN basis)	128.
Spanning-tree VLAN port cost (configurable on a per-VLAN basis)	1000 Mb/s: 4. 100 Mb/s: 19. 10 Mb/s: 100.
Spanning-tree timers	Hello time: 2 seconds. Forward-delay time: 15 seconds. Maximum-aging time: 20 seconds. Transmit hold count: 6 BPDUs

Disabling Spanning Tree

Spanning tree is enabled by default on VLAN 1 and on all newly created VLANs up to the spanning-tree limit specified in the Supported Spanning-Tree Instances, page 340. Disable spanning tree only if you are sure there are no loops in the network topology.

Caution: When spanning tree is disabled and loops are present in the topology, excessive traffic and indefinite packet duplication can drastically reduce network performance.

Root Switch

The switch maintains a separate spanning-tree instance for each active VLAN configured on it. A bridge ID, consisting of the switch priority and the switch MAC address, is associated with each instance. For each VLAN, the switch with the lowest bridge ID becomes the root switch for that VLAN.

To configure a switch to become the root for the specified VLAN, use the **spanning-tree vlan** *vlan-id* **root** global configuration command to modify the switch priority from the default value (32768) to a significantly lower value. When you enter this command, the software checks the switch priority of the root switches for each VLAN. Because of the extended system ID support, the switch sets its own priority for the specified VLAN to 24576 if this value will cause this switch to become the root for the specified VLAN.

If any root switch for the specified VLAN has a switch priority lower than 24576, the switch sets its own priority for the specified VLAN to 4096 less than the lowest switch priority. (4096 is the value of the least-significant bit of a 4-bit switch priority value as shown in Table 37 on page 335.)

Note: The **spanning-tree vlan** *vlan-id* **root** global configuration command fails if the value necessary to be the root switch is less than 1.

Note: If your network consists of switches that both do and do not support the extended system ID, it is unlikely that the switch with the extended system ID support will become the root switch. The extended system ID increases the switch priority value every time the VLAN number is greater than the priority of the connected switches running older software.

Note: The root switch for each spanning-tree instance should be a backbone or distribution switch. Do not configure an access switch as the spanning-tree primary root.

Use the **diameter** keyword to specify the Layer 2 network diameter (that is, the maximum number of switch hops between any two end stations in the Layer 2 network). When you specify the network diameter, the switch automatically sets an optimal hello time, forward-delay time, and maximum-age time for a network of that diameter, which can significantly reduce the convergence time. You can use the **hello** keyword to override the automatically calculated hello time.

Note: After configuring the switch as the root switch, we recommend that you avoid manually configuring the hello time, forward-delay time, and maximum-age time through the **spanning-tree vlan** *vlan-id* **hello-time**, **spanning-tree vlan** *vlan-id* **forward-time**, and the **spanning-tree vlan** *vlan-id* **max-age** global configuration commands.

Secondary Root Switch

When you configure a switch as the secondary root, the switch priority is modified from the default value (32768) to 28672. The switch is then likely to become the root switch for the specified VLAN if the primary root switch fails. This is assuming that the other network switches use the default switch priority of 32768 and therefore are unlikely to become the root switch.

You can execute this command on more than one switch to configure multiple backup root switches. Use the same network diameter and hello-time values that you used when you configured the primary root switch with the **spanning-tree vlan** *vlan-id* **root primary** global configuration command.

Port Priority

If a loop occurs, spanning tree uses the port priority when selecting an interface to put into the forwarding state. You can assign higher priority values (lower numerical values) to interfaces that you want selected first and lower priority values (higher numerical values) that you want selected last. If all interfaces have the same priority value, spanning tree puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.

Path Cost

The spanning-tree path cost default value is derived from the media speed of an interface. If a loop occurs, spanning tree uses cost when selecting an interface to put in the forwarding state. You can assign lower cost values to interfaces that you want selected first and higher cost values that you want selected last. If all interfaces have the same cost value, spanning tree puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.

Spanning-Tree Timers

Table 40 Spanning-Tree Timers

Variable	Description
Hello timer	Controls how often the switch broadcasts hello messages to other switches.
Forward-delay timer	Controls how long each of the listening and learning states last before the interface begins forwarding.
Maximum-age timer	Controls the amount of time the switch stores protocol information received on an interface.
Transmit hold count	Controls the number of BPDUs that can be sent before pausing for 1 second.

Spanning-Tree Configuration Guidelines

If more VLANs are defined in the VTP than there are spanning-tree instances, you can enable PVST+ or rapid PVST+ on only 128 VLANs on the switch. The remaining VLANs operate with spanning tree disabled. However, you can map multiple VLANs to the same spanning-tree instances by using MSTP. For more information, see Configuring MSTP, page 351

If 128 instances of spanning tree are already in use, you can disable spanning tree on one of the VLANs and then enable it on the VLAN where you want it to run. Use the **no spanning-tree vlan** *vlan-id* global configuration command to disable spanning tree on a specific VLAN, and use the **spanning-tree vlan** *vlan-id* global configuration command to enable spanning tree on the desired VLAN.

Caution: Switches that are not running spanning tree still forward BPDUs that they receive so that the other switches on the VLAN that have a running spanning-tree instance can break loops. Therefore, spanning tree must be running on enough switches to break all the loops in the network; for example, at least one switch on each loop in the VLAN must be running spanning tree. It is not absolutely necessary to run spanning tree on all switches in the VLAN. However, if you are running spanning tree only on a minimal set of switches, an incautious change to the network that introduces another loop into the VLAN can result in a broadcast storm.

Note: If you have already used all available spanning-tree instances on your switch, adding another VLAN anywhere in the VTP domain creates a VLAN that is not running spanning tree on that switch. If you have the default allowed list on the trunk ports of that switch, the new VLAN is carried on all trunk ports. Depending on the topology of the network, this could create a loop in the new VLAN that will not be broken, particularly if there are several adjacent switches that have all run out of spanning-tree instances. You can prevent this possibility by setting up allowed lists on the trunk ports of switches that have used up their allocation of spanning-tree instances. Setting up allowed lists is not necessary in many cases and can make it more labor-intensive to add another VLAN to the network.

Spanning-tree commands control the configuration of VLAN spanning-tree instances. You create a spanning-tree instance when you assign an interface to a VLAN. The spanning-tree instance is removed when the last interface is moved to another VLAN. You can configure switch and port parameters before a spanning-tree instance is created; these parameters are applied when the spanning-tree instance is created.

The switch supports PVST+, rapid PVST+, and MSTP, but only one version can be active at any time. (For example, all VLANs run PVST+, all VLANs run rapid PVST+, or all VLANs run MSTP.) For information about the different spanning-tree modes and how they interoperate, see Spanning-Tree Interoperability and Backward Compatibility, page 340.

For configuration information about UplinkFast and BackboneFast, see Information About Configuring the Optional Spanning-Tree Features, page 371.

Caution: Loop guard works only on point-to-point links. We recommend that each end of the link has a directly connected device that is running STP.

How to Configure STP

Changing the Spanning-Tree Mode

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**spanning-tree mode** {**pvst** | **mst** | **rapid-pvst**}	Configures a spanning-tree mode. ■ **pvst**–Enables PVST+ (the default setting). ■ **mst**–Enables MSTP (and RSTP). For more configuration steps, see Configuring MSTP, page 351 ■ **rapid-pvst**–Enables rapid PVST+.
3.	**interface** *interface-id*	(Recommended for rapid-PVST+ mode only) Specifies an interface to configure, and enters interface configuration mode. Valid interfaces include physical ports, VLANs, and port channels.
4.	**spanning-tree link-type point-to-point**	(Recommended for rapid-PVST+ mode only) Specifies that the link type for this port is point-to-point. If you connect this port (local port) to a remote port through a point-to-point link and the local port becomes a designated port, the switch negotiates with the remote port and rapidly changes the local port to the forwarding state.
5.	**end**	Returns to privileged EXEC mode.
6.	**clear spanning-tree detected-protocols**	(Recommended for rapid-PVST+ mode only) Restarts the protocol migration process on the entire switch if any port on the switch is connected to a port on a legacy IEEE 802.1D switch, This step is optional if the designated switch detects that this switch is running rapid PVST+.

Configuring the Root Switch

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**spanning-tree vlan** *vlan-id* **root primary** [**diameter** *net-diameter* [**hello-time** *seconds*]]	Configures a switch to become the root for the specified VLAN. ■ *vlan-id*–Specifies a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. ■ (Optional) **diameter** *net-diameter*–Specifies the maximum number of switches between any two end stations. ■ (Optional) **hello-time** *seconds*–Specifies the interval in seconds between the generation of configuration messages by the root switch.
3.	**end**	Returns to privileged EXEC mode.

Configuring a Secondary Root Switch

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**spanning-tree vlan** *vlan-id* **root secondary** [**diameter** *net-diameter* [**hello-time** *seconds*]]	Configures a switch to become the secondary root for the specified VLAN. ■ *vlan-id*–Specifies a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4096. ■ (Optional) **diameter** *net-diameter*–Specifies the maximum number of switches between any two end stations. The range is 2 to 7. ■ (Optional) **hello-time** *seconds*–Specifies the interval in seconds between the generation of configuration messages by the root switch. The range is 1 to 10; the default is 2. Use the same network diameter and hello-time values that you used when configuring the primary root switch. See Configuring the Root Switch, page 346.
3.	**end**	Returns to privileged EXEC mode.

Configuring Port Priority

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies an interface to configure, and enters interface configuration mode. Valid interfaces include physical ports and port-channel logical interfaces (**port-channel** *port-channel-number*).
3.	**spanning-tree port-priority** *priority*	Configures the port priority for an interface.
4.	**spanning-tree vlan** *vlan-id* **port-priority** *priority*	Configures the port priority for a VLAN.
5.	**end**	Returns to privileged EXEC mode.

Configuring Path Cost

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies an interface to configure, and enters interface configuration mode. Valid interfaces include physical ports and port-channel logical interfaces (**port-channel** *port-channel-number*).
3.	**spanning-tree cost** *cost*	Configures the cost for an interface.
4.	**spanning-tree vlan** *vlan-id* **cost** *cost*	Configures the cost for a VLAN. If a loop occurs, spanning tree uses the path cost when selecting an interface to place into the forwarding state. A lower path cost represents higher-speed transmission.
5.	**end**	Returns to privileged EXEC mode.

Configuring Optional STP Parameters

Before You Begin

Exercise care when configuring the priority, and hello time for STP.

For most situations, we recommend that you use the **spanning-tree vlan** *vlan-id* **root primary** and the **spanning-tree vlan** *vlan-id* **root secondary** global configuration commands to modify the switch priority.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**spanning-tree vlan** *vlan-id* **priority** *priority*	Configures the switch priority of a VLAN.
3.	**spanning-tree vlan** *vlan-id* **hello-time** *seconds*	Configures the hello time of a VLAN.
4.	**spanning-tree vlan** *vlan-id* **max-age** *seconds*	Configures the maximum-aging time of a VLAN.
5.	**spanning-tree vlan** *vlan-id* **forward-time** *seconds*	Configures the forward time of a VLAN.

	Command	Purpose
6.	**spanning-tree vlan** *vlan-id* **max-age** *seconds*	Configures the maximum-aging time of a VLAN.
7.	**spanning-tree transmit hold-count** *value*	Configures the number of BPDUs that can be sent before pausing for 1 second. **Note:** Changing this parameter to a higher value can have a significant impact on CPU utilization, especially in Rapid-PVST mode. Lowering this value can slow down convergence in certain scenarios. We recommend that you maintain the default setting.
8.	**end**	Returns to privileged EXEC mode.

Monitoring and Maintaining STP

Command	Purpose
show spanning-tree active	Displays spanning-tree information on active interfaces only.
show spanning-tree detail	Displays a detailed summary of interface information.
show spanning-tree interface *interface-id*	Displays spanning-tree information for the specified interface.
show spanning-tree summary	Displays a summary of interface states.
show spanning-tree vlan *vlan-id*	Displays spanning-tree VLAN entries.
copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*
VLAN configuration	*Configuring VLANs, page 289*
Multiple Spanning Tree Protocol configuration	*Configuring MSTP, page 351*
Optional Spanning-Tree configuration	*Configuring Optional Spanning-Tree Features, page 371*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Configuring MSTP

Information About Configuring MSTP

This chapter describes how to configure the Cisco implementation of the IEEE 802.1s Multiple STP (MSTP) on the switch.

Note: The multiple spanning-tree (MST) implementation is based on the IEEE 802.1s standard.

The MSTP enables multiple VLANs to be mapped to the same spanning-tree instance, reducing the number of spanning-tree instances needed to support a large number of VLANs. The MSTP provides for multiple forwarding paths for data traffic and enables load balancing. It improves the fault tolerance of the network because a failure in one instance (forwarding path) does not affect other instances (forwarding paths). The most common initial deployment of MSTP is in the backbone and distribution layers of a Layer 2 switched network. This deployment provides the highly available network required in a service-provider environment.

When the switch is in the MST mode, the Rapid Spanning Tree Protocol (RSTP), which is based on IEEE 802.1w, is automatically enabled. The RSTP provides rapid convergence of the spanning tree through explicit handshaking that eliminates the IEEE 802.1D forwarding delay and quickly transitions root ports and designated ports to the forwarding state.

Both MSTP and RSTP improve the spanning-tree operation and maintain backward compatibility with equipment that is based on the (original) IEEE 802.1D spanning tree, with existing Cisco-proprietary Multiple Instance STP (MISTP), and with existing Cisco per-VLAN spanning-tree plus (PVST+) and rapid per-VLAN spanning-tree plus (rapid PVST+).

MSTP

MSTP, which uses RSTP for rapid convergence, enables VLANs to be grouped into a spanning-tree instance, with each instance having a spanning-tree topology independent of other spanning-tree instances. This architecture provides multiple forwarding paths for data traffic, enables load balancing, and reduces the number of spanning-tree instances required to support a large number of VLANs.

Multiple Spanning-Tree Regions

For switches to participate in multiple spanning-tree (MST) instances, you must consistently configure the switches with the same MST configuration information. A collection of interconnected switches that have the same MST configuration comprises an MST region as shown in Figure 1 on page 3.

The MST configuration controls to which MST region each switch belongs. The configuration includes the name of the region, the revision number, and the MST VLAN-to-instance assignment map. You configure the switch for a region by using the **spanning-tree mst configuration** global configuration command, after which the switch enters the MST configuration mode. From this mode, you can map VLANs to an MST instance by using the **instance** MST configuration command, specify the region name by using the **name** MST configuration command, and set the revision number by using the **revision** MST configuration command.

A region can have one or multiple members with the same MST configuration. Each member must be capable of processing RSTP bridge protocol data units (BPDUs). There is no limit to the number of MST regions in a network, but each region can support up to 65 spanning-tree instances. Instances can be identified by any number in the range from 0 to 4096. You can assign a VLAN to only one spanning-tree instance at a time.

IST, CIST, and CST

Unlike PVST+ and rapid PVST+ in which all the spanning-tree instances are independent, the MSTP establishes and maintains two types of spanning trees:

- An internal spanning tree (IST), which is the spanning tree that runs in an MST region.

  Within each MST region, the MSTP maintains multiple spanning-tree instances. Instance 0 is a special instance for a region, known as the internal spanning tree (IST). All other MST instances are numbered from 1 to 4096.

  The IST is the only spanning-tree instance that sends and receives BPDUs. All of the other spanning-tree instance information is contained in M-records, which are encapsulated within MSTP BPDUs. Because the MSTP BPDU carries information for all instances, the number of BPDUs that need to be processed to support multiple spanning-tree instances is significantly reduced.

  All MST instances within the same region share the same protocol timers, but each MST instance has its own topology parameters, such as root switch ID, root path cost, and so forth. By default, all VLANs are assigned to the IST.

  An MST instance is local to the region; for example, MST instance 1 in region A is independent of MST instance 1 in region B, even if regions A and B are interconnected.

- A common and internal spanning tree (CIST), which is a collection of the ISTs in each MST region, and the common spanning tree (CST) that interconnects the MST regions and single spanning trees.

  The spanning tree computed in a region appears as a subtree in the CST that encompasses the entire switched domain. The CIST is formed by the spanning-tree algorithm running among switches that support the IEEE 802.1w, IEEE 802.1s, and IEEE 802.1D standards. The CIST inside an MST region is the same as the CST outside a region.

For more information, see Operations Within an MST Region, page 2 and the Operations Between MST Regions, page 3.

Note: The implementation of the IEEE 802.1s standard, changes some of the terminology associated with MST implementations.

Operations Within an MST Region

The IST connects all the MSTP switches in a region. When the IST converges, the root of the IST becomes the CIST regional root (called the *IST master* before the implementation of the IEEE 802.1s standard) as shown in Figure 1 on page 3. It is the switch within the region with the lowest switch ID and path cost to the CIST root. The CIST regional root is also the CIST root if there is only one region in the network. If the CIST root is outside the region, one of the MSTP switches at the boundary of the region is selected as the CIST regional root.

When an MSTP switch initializes, it sends BPDUs claiming itself as the root of the CIST and the CIST regional root, with both of the path costs to the CIST root and to the CIST regional root set to zero. The switch also initializes all of its MST instances and claims to be the root for all of them. If the switch receives superior MST root information (lower switch ID, lower path cost, and so forth) than currently stored for the port, it relinquishes its claim as the CIST regional root.

During initialization, a region might have many subregions, each with its own CIST regional root. As switches receive superior IST information, they leave their old subregions and join the new subregion that contains the true CIST regional root. All subregions shrink, except for the one that contains the true CIST regional root.

For correct operation, all switches in the MST region must agree on the same CIST regional root. Therefore, any two switches in the region only synchronize their port roles for an MST instance if they converge to a common CIST regional root.

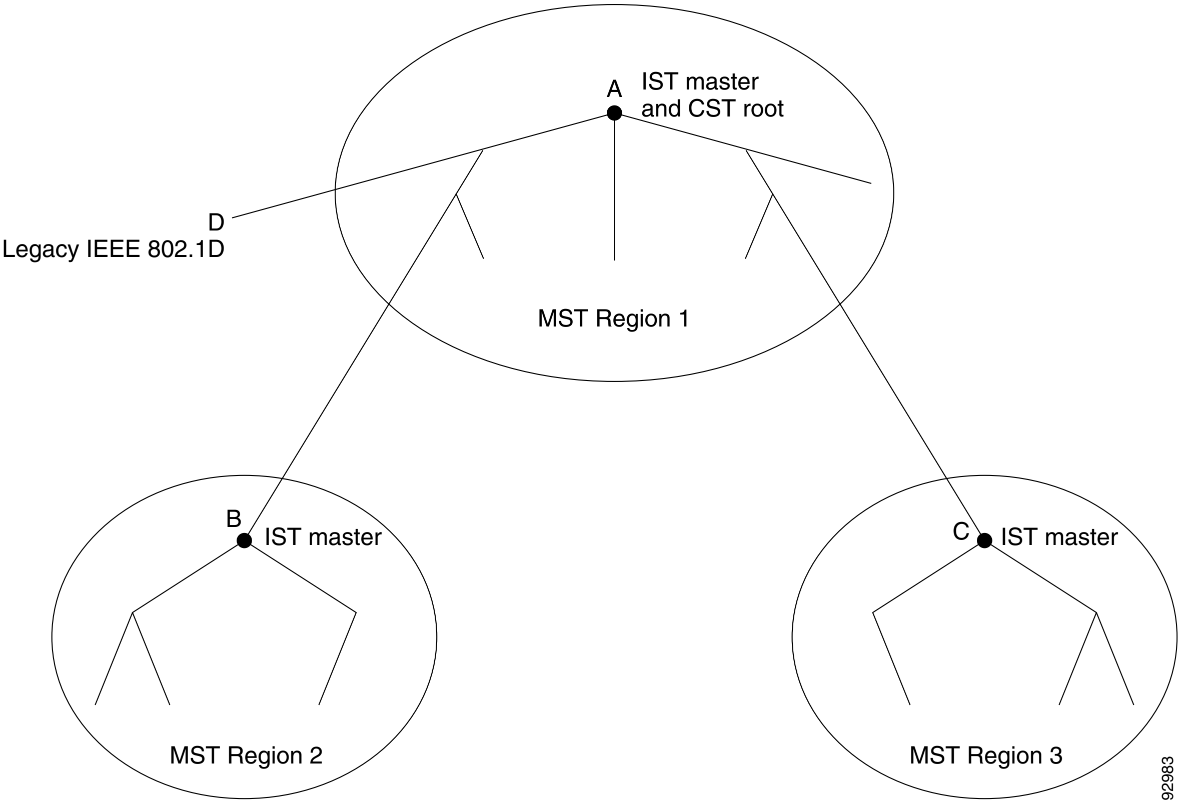
Operations Between MST Regions

If there are multiple regions or legacy IEEE 802.1D switches within the network, MSTP establishes and maintains the CST, which includes all MST regions and all legacy STP switches in the network. The MST instances combine with the IST at the boundary of the region to become the CST.

The IST connects all the MSTP switches in the region and appears as a subtree in the CIST that encompasses the entire switched domain. The root of the subtree is the CIST regional root. The MST region appears as a virtual switch to adjacent STP switches and MST regions.

Figure 1 on page 3 shows a network with three MST regions and a legacy IEEE 802.1D switch (D). The CIST regional root for region 1 (A) is also the CIST root. The CIST regional root for region 2 (B) and the CIST regional root for region 3 (C) are the roots for their respective subtrees within the CIST. The RSTP runs in all regions.

Figure 39 MST Regions, CIST Masters, and CST Root



Only the CST instance sends and receives BPDUs, and MST instances add their spanning-tree information into the BPDUs to interact with neighboring switches and compute the final spanning-tree topology. Because of this, the spanning-tree parameters related to BPDU transmission (for example, hello time, forward time, max-age, and max-hops) are configured only on the CST instance but affect all MST instances. Parameters related to the spanning-tree topology (for example, switch priority, port VLAN cost, and port VLAN priority) can be configured on both the CST instance and the MST instance.

MSTP switches use Version 3 RSTP BPDUs or IEEE 802.1D STP BPDUs to communicate with legacy IEEE 802.1D switches. MSTP switches use MSTP BPDUs to communicate with MSTP switches.

IEEE 802.1s Terminology

Some MST naming conventions used in Cisco's prestandard implementation have been changed to identify some *internal* or *regional* parameters. These parameters are significant only within an MST region, as opposed to external parameters that are relevant to the whole network. Because the CIST is the only spanning-tree instance that spans the whole network, only the CIST parameters require the external rather than the internal or regional qualifiers.

- The CIST root is the root switch for the unique instance that spans the whole network, the CIST.

- The CIST external root path cost is the cost to the CIST root. This cost is left unchanged within an MST region. Remember that an MST region looks like a single switch for the CIST. The CIST external root path cost is the root path cost calculated between these virtual switches and switches that do not belong to any region.
- The CIST regional root was called the IST master in the prestandard implementation. If the CIST root is in the region, the CIST regional root is the CIST root. Otherwise, the CIST regional root is the closest switch to the CIST root in the region. The CIST regional root acts as a root switch for the IST.
- The CIST internal root path cost is the cost to the CIST regional root in a region. This cost is only relevant to the IST, instance 0.

Table 41 on page 354 compares the IEEE standard and the Cisco prestandard terminology.

Table 41 IEEE standard and the Cisco Prestandard Terminology

IEEE Standard	Cisco Prestandard	Cisco Standard
CIST regional root	IST master	CIST regional root
CIST internal root path cost	IST master path cost	CIST internal path cost
CIST external root path cost	Root path cost	Root path cost
MSTI regional root	Instance root	Instance root
MSTI internal root path cost	Root path cost	Root path cost

Hop Count

The IST and MST instances do not use the message-age and maximum-age information in the configuration BPDU to compute the spanning-tree topology. Instead, they use the path cost to the root and a hop-count mechanism similar to the IP time-to-live (TTL) mechanism.

By using the **spanning-tree mst max-hops** global configuration command, you can configure the maximum hops inside the region and apply it to the IST and all MST instances in that region. The hop count achieves the same result as the message-age information (triggers a reconfiguration). The root switch of the instance always sends a BPDU (or M-record) with a cost of 0 and the hop count set to the maximum value. When a switch receives this BPDU, it decrements the received remaining hop count by one and propagates this value as the remaining hop count in the BPDUs it generates. When the count reaches zero, the switch discards the BPDU and ages the information held for the port.

The message-age and maximum-age information in the RSTP portion of the BPDU remain the same throughout the region, and the same values are propagated by the region designated ports at the boundary.

Boundary Ports

In the Cisco prestandard implementation, a boundary port connects an MST region to a single spanning-tree region running RSTP, to a single spanning-tree region running PVST+ or rapid PVST+, or to another MST region with a different MST configuration. A boundary port also connects to a LAN, the designated switch of which is either a single spanning-tree switch or a switch with a different MST configuration.

There is no definition of a boundary port in the IEEE 802.1s standard. The IEEE 802.1Q-2002 standard identifies two kinds of messages that a port can receive: internal (coming from the same region) and external. When a message is external, it is received only by the CIST. If the CIST role is root or alternate, or if the external BPDU is a topology change, it could have an impact on the MST instances. When a message is internal, the CIST part is received by the CIST, and each MST instance receives its respective M-record. The Cisco prestandard implementation treats a port that receives an external message as a boundary port. This means a port cannot receive a mix of internal and external messages.

An MST region includes both switches and LANs. A segment belongs to the region of its designated port. Therefore, a port in a different region than the designated port for a segment is a boundary port. This definition allows two ports internal to a region to share a segment with a port belonging to a different region, creating the possibility of receiving both internal and external messages on a port.

The primary change from the Cisco prestandard implementation is that a designated port is not defined as boundary, unless it is running in an STP-compatible mode.

Note: If there is a legacy STP switch on the segment, messages are always considered external.

The other change from the prestandard implementation is that the CIST regional root switch ID field is now inserted where an RSTP or legacy IEEE 802.1Q switch has the sender switch ID. The whole region performs like a single virtual switch by sending a consistent sender switch ID to neighboring switches. In this example, switch C would receive a BPDU with the same consistent sender switch ID of root, whether or not A or B is designated for the segment.

IEEE 802.1s Implementation

The Cisco implementation of the IEEE MST standard includes features required to meet the standard, as well as some of the desirable prestandard functionality that is not yet incorporated into the published standard.

Port Role Naming Change

The boundary role is no longer in the final MST standard, but this boundary concept is maintained in Cisco's implementation. However, an MST instance port at a boundary of the region might not follow the state of the corresponding CIST port. Two cases exist now:

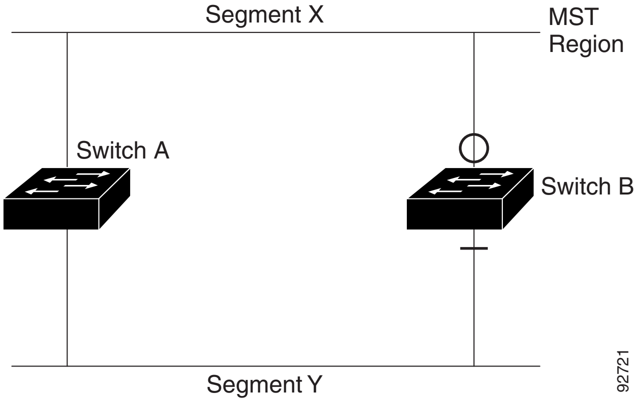
- The boundary port is the root port of the CIST regional root—When the CIST instance port is proposed and is in sync, it can send back an agreement and move to the forwarding state only after all the corresponding MSTI ports are in sync (and forwarding). The MSTI ports now have a special *master* role.
- The boundary port is not the root port of the CIST regional root—The MSTI ports follow the state and role of the CIST port. The standard provides less information, and it might be difficult to understand why an MSTI port can be alternately blocking when it receives no BPDUs (MRecords). In this case, although the boundary role no longer exists, the **show** commands identify a port as boundary in the *type* column of the output.

Interoperation Between Legacy and Standard Switches

Because automatic detection of prestandard switches can fail, you can use an interface configuration command to identify prestandard ports. A region cannot be formed between a standard and a prestandard switch, but they can interoperate by using the CIST. Only the capability of load balancing over different instances is lost in that particular case. The CLI displays different flags depending on the port configuration when a port receives prestandard BPDUs. A syslog message also appears the first time a switch receives a prestandard BPDU on a port that has not been configured for prestandard BPDU transmission.

Figure 2 on page 6 illustrates this scenario. Assume that A is a standard switch and B a prestandard switch, both configured to be in the same region. A is the root switch for the CIST, and B has a root port (BX) on segment X and an alternate port (BY) on segment Y. If segment Y flaps, and the port on BY becomes the alternate before sending out a single prestandard BPDU, AY cannot detect that a prestandard switch is connected to Y and continues to send standard BPDUs. The port BY is fixed in a boundary, and no load balancing is possible between A and B. The same problem exists on segment X, but B might transmit topology changes.

Figure 40 Standard and Prestandard Switch Interoperation



Note: We recommend that you minimize the interaction between standard and prestandard MST implementations.

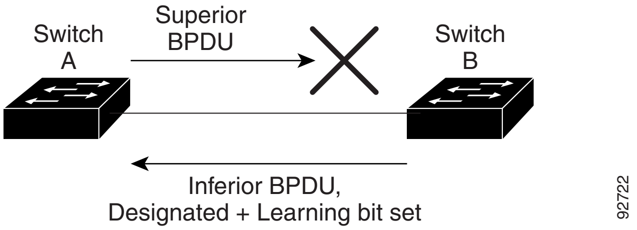
Detecting Unidirectional Link Failure

This feature is not yet present in the IEEE MST standard, but it is included in this Cisco IOS release. The software checks the consistency of the port role and state in the received BPDUs to detect unidirectional link failures that could cause bridging loops.

When a designated port detects a conflict, it keeps its role, but reverts to discarding state because disrupting connectivity in case of inconsistency is preferable to opening a bridging loop.

Figure 3 on page 6 illustrates a unidirectional link failure that typically creates a bridging loop. Switch A is the root switch, and its BPDUs are lost on the link leading to switch B. RSTP and MST BPDUs include the role and state of the sending port. With this information, switch A can detect that switch B does not react to the superior BPDUs it sends and that switch B is the designated, not root switch. As a result, switch A blocks (or keeps blocking) its port, preventing the bridging loop.

Figure 41 Detecting Unidirectional Link Failure



Interoperability with IEEE 802.1D STP

A switch running MSTP supports a built-in protocol migration mechanism that enables it to interoperate with legacy IEEE 802.1D switches. If this switch receives a legacy IEEE 802.1D configuration BPDU (a BPDU with the protocol version set to 0), it sends only IEEE 802.1D BPDUs on that port. An MSTP switch also can detect that a port is at the boundary of a region when it receives a legacy BPDU, an MSTP BPDU (Version 3) associated with a different region, or an RSTP BPDU (Version 2).

However, the switch does not automatically revert to the MSTP mode if it no longer receives IEEE 802.1D BPDUs because it cannot detect whether the legacy switch has been removed from the link unless the legacy switch is the designated switch. A switch might also continue to assign a boundary role to a port when the switch to which this switch is connected has joined the region. To restart the protocol migration process (force the renegotiation with neighboring switches), use the **clear spanning-tree detected-protocols** privileged EXEC command.

If all the legacy switches on the link are RSTP switches, they can process MSTP BPDUs as if they are RSTP BPDUs. Therefore, MSTP switches send either a Version 0 configuration and TCN BPDUs or Version 3 MSTP BPDUs on a boundary port. A boundary port connects to a LAN, the designated switch of which is either a single spanning-tree switch or a switch with a different MST configuration.

RSTP

The RSTP takes advantage of point-to-point wiring and provides rapid convergence of the spanning tree. Reconfiguration of the spanning tree can occur in less than 1 second (in contrast to 50 seconds with the default settings in the IEEE 802.1D spanning tree).

Port Roles and the Active Topology

The RSTP provides rapid convergence of the spanning tree by assigning port roles and by learning the active topology. The RSTP builds upon the IEEE 802.1D STP to select the switch with the highest switch priority (lowest numerical priority value) as the root switch as described in the Configuring STP, page 1. Then the RSTP assigns one of these port roles to individual ports:

- Root port—Provides the best path (lowest cost) when the switch forwards packets to the root switch.
- Designated port—Connects to the designated switch, which incurs the lowest path cost when forwarding packets from that LAN to the root switch. The port through which the designated switch is attached to the LAN is called the designated port.
- Alternate port—Offers an alternate path toward the root switch to that provided by the current root port.
- Backup port—Acts as a backup for the path provided by a designated port toward the leaves of the spanning tree. A backup port can exist only when two ports are connected in a loopback by a point-to-point link or when a switch has two or more connections to a shared LAN segment.
- Disabled port—Has no role within the operation of the spanning tree.

A port with the root or a designated port role is included in the active topology. A port with the alternate or backup port role is excluded from the active topology.

In a stable topology with consistent port roles throughout the network, the RSTP ensures that every root port and designated port immediately transition to the forwarding state while all alternate and backup ports are always in the discarding state (equivalent to blocking in IEEE 802.1D). The port state controls the operation of the forwarding and learning processes. Table 42 on page 357 provides a comparison of IEEE 802.1D and RSTP port states.

Table 42 EEE 802.1D and RSTP Port States

Operational Status	STP Port State (IEEE 802.1D)	RSTP Port State	Is Port Included in the Active Topology?
Enabled	Blocking	Discarding	No
Enabled	Listening	Discarding	No
Enabled	Learning	Learning	Yes
Enabled	Forwarding	Forwarding	Yes
Disabled	Disabled	Discarding	No

To be consistent with Cisco STP implementations, this guide defines the port state as *blocking* instead of *discarding*. Designated ports start in the listening state.

Rapid Convergence

The RSTP provides for rapid recovery of connectivity following the failure of a switch, a switch port, or a LAN. It provides rapid convergence for edge ports, new root ports, and ports connected through point-to-point links as follows:

- Edge ports—If you configure a port as an edge port on an RSTP switch by using the **spanning-tree portfast** interface configuration command, the edge port immediately transitions to the forwarding state. An edge port is the same as a Port Fast-enabled port, and you should enable it only on ports that connect to a single end station.

- Root ports—If the RSTP selects a new root port, it blocks the old root port and immediately transitions the new root port to the forwarding state.

- Point-to-point links—If you connect a port to another port through a point-to-point link and the local port becomes a designated port, it negotiates a rapid transition with the other port by using the proposal-agreement handshake to ensure a loop-free topology.

  As shown in Figure 4 on page 9, Switch A is connected to Switch B through a point-to-point link, and all of the ports are in the blocking state. Assume that the priority of Switch A is a smaller numerical value than the priority of Switch B. Switch A sends a proposal message (a configuration BPDU with the proposal flag set) to Switch B, proposing itself as the designated switch.

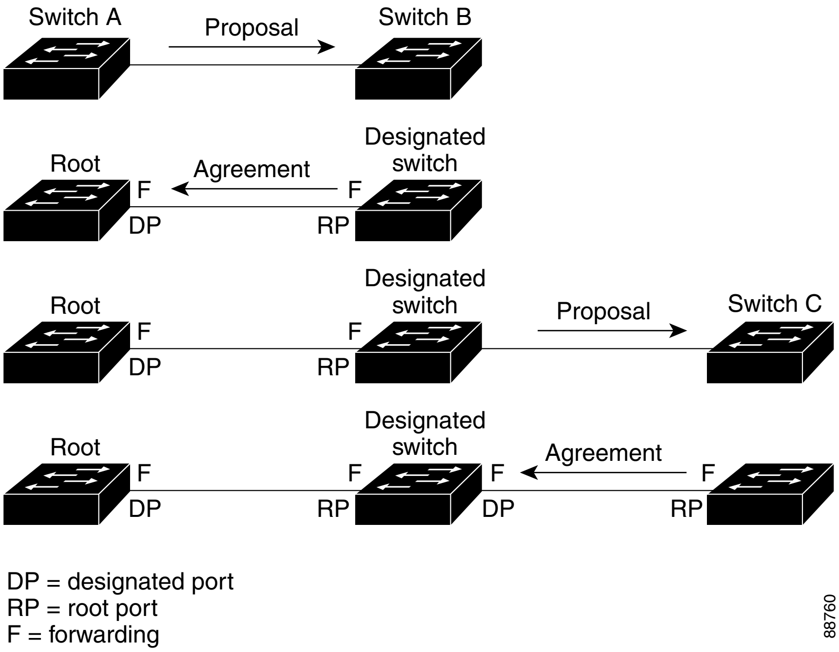
  After receiving the proposal message, Switch B selects as its new root port the port from which the proposal message was received, forces all nonedge ports to the blocking state, and sends an agreement message (a BPDU with the agreement flag set) through its new root port.

  After receiving Switch B’s agreement message, Switch A also immediately transitions its designated port to the forwarding state. No loops in the network are formed because Switch B blocked all of its nonedge ports and because there is a point-to-point link between Switches A and B.

  When Switch C is connected to Switch B, a similar set of handshaking messages are exchanged. Switch C selects the port connected to Switch B as its root port, and both ends immediately transition to the forwarding state. With each iteration of this handshaking process, one more switch joins the active topology. As the network converges, this proposal-agreement handshaking progresses from the root toward the leaves of the spanning tree.

  The switch learns the link type from the port duplex mode: a full-duplex port is considered to have a point-to-point connection; a half-duplex port is considered to have a shared connection. You can override the default setting that is controlled by the duplex setting by using the **spanning-tree link-type** interface configuration command.

Figure 42 Proposal and Agreement Handshaking for Rapid Convergence



Synchronization of Port Roles

When the switch receives a proposal message on one of its ports and that port is selected as the new root port, the RSTP forces all other ports to synchronize with the new root information.

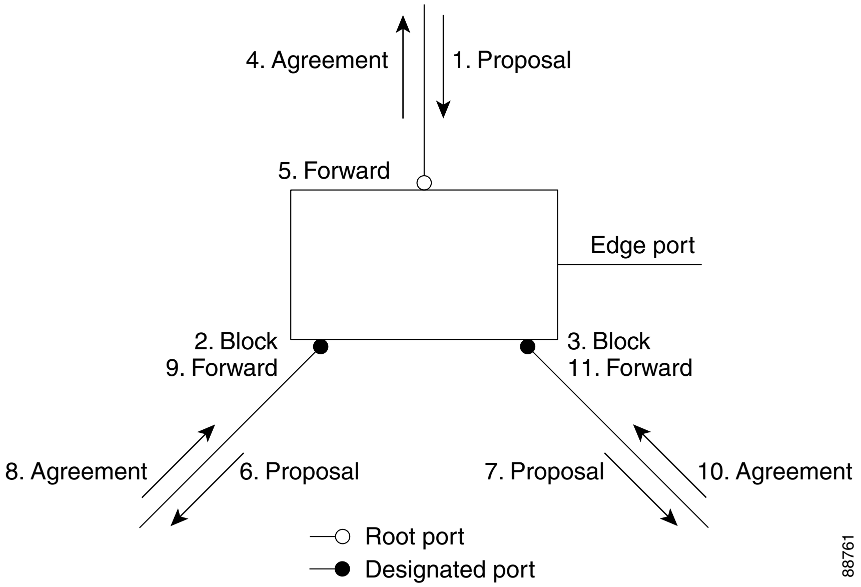
The switch is synchronized with superior root information received on the root port if all other ports are synchronized. An individual port on the switch is synchronized if

- That port is in the blocking state.
- It is an edge port (a port configured to be at the edge of the network).

If a designated port is in the forwarding state and is not configured as an edge port, it transitions to the blocking state when the RSTP forces it to synchronize with new root information. In general, when the RSTP forces a port to synchronize with root information and the port does not satisfy any of the above conditions, its port state is set to blocking.

After ensuring that all of the ports are synchronized, the switch sends an agreement message to the designated switch corresponding to its root port. When the switches connected by a point-to-point link are in agreement about their port roles, the RSTP immediately transitions the port states to forwarding. The sequence of events is shown in Figure 5 on page 10.

Figure 43 Sequence of Events During Rapid Convergence



Bridge Protocol Data Unit Format and Processing

The RSTP BPDU format is the same as the IEEE 802.1D BPDU format except that the protocol version is set to 2. A new 1-byte Version 1 Length field is set to zero, which means that no version 1 protocol information is present. Table 3 shows the RSTP flag fields.

Table 43 RSTP Flag Fields

Bit	Function
0	Topology change (TC)
1	Proposal
2–3: 00 01 10 11	Port role: Unknown Alternate port Root port Designated port
4	Learning
5	Forwarding
6	Agreement
7	Topology change acknowledgement (TCA)

The sending switch sets the proposal flag in the RSTP BPDU to propose itself as the designated switch on that LAN. The port role in the proposal message is always set to the designated port.

The sending switch sets the agreement flag in the RSTP BPDU to accept the previous proposal. The port role in the agreement message is always set to the root port.

The RSTP does not have a separate topology change notification (TCN) BPDU. It uses the topology change (TC) flag to show the topology changes. However, for interoperability with IEEE 802.1D switches, the RSTP switch processes and generates TCN BPDUs.

The learning and forwarding flags are set according to the state of the sending port.

Processing Superior BPDU Information

If a port receives superior root information (lower switch ID, lower path cost, and so forth) than currently stored for the port, the RSTP triggers a reconfiguration. If the port is proposed and is selected as the new root port, RSTP forces all the other ports to synchronize.

If the BPDU received is an RSTP BPDU with the proposal flag set, the switch sends an agreement message after all of the other ports are synchronized. If the BPDU is an IEEE 802.1D BPDU, the switch does not set the proposal flag and starts the forward-delay timer for the port. The new root port requires twice the forward-delay time to transition to the forwarding state.

If the superior information received on the port causes the port to become a backup or alternate port, RSTP sets the port to the blocking state but does not send the agreement message. The designated port continues sending BPDUs with the proposal flag set until the forward-delay timer expires, at which time the port transitions to the forwarding state.

Processing Inferior BPDU Information

If a designated port receives an inferior BPDU (higher switch ID, higher path cost, and so forth than currently stored for the port) with a designated port role, it immediately replies with its own information.

Topology Changes

This section describes the differences between the RSTP and the IEEE 802.1D in handling spanning-tree topology changes.

- Detection—Unlike IEEE 802.1D in which *any* transition between the blocking and the forwarding state causes a topology change, *only* transitions from the blocking to the forwarding state cause a topology change with RSTP (only an increase in connectivity is considered a topology change). State changes on an edge port do not cause a topology change. When an RSTP switch detects a topology change, it deletes the learned information on all of its nonedge ports except on those from which it received the TC notification.

- Notification—Unlike IEEE 802.1D, which uses TCN BPDUs, the RSTP does not use them. However, for IEEE 802.1D interoperability, an RSTP switch processes and generates TCN BPDUs.

- Acknowledgement—When an RSTP switch receives a TCN message on a designated port from an IEEE 802.1D switch, it replies with an IEEE 802.1D configuration BPDU with the TCA bit set. However, if the TC-while timer (the same as the topology-change timer in IEEE 802.1D) is active on a root port connected to an IEEE 802.1D switch and a configuration BPDU with the TCA bit set is received, the TC-while timer is reset.

  This behavior is only required to support IEEE 802.1D switches. The RSTP BPDUs never have the TCA bit set.

- Propagation—When an RSTP switch receives a TC message from another switch through a designated or root port, it propagates the change to all of its nonedge, designated ports and to the root port (excluding the port on which it is received). The switch starts the TC-while timer for all such ports and flushes the information learned on them.

- Protocol migration—For backward compatibility with IEEE 802.1D switches, RSTP selectively sends IEEE 802.1D configuration BPDUs and TCN BPDUs on a per-port basis.

  When a port is initialized, the migrate-delay timer is started (specifies the minimum time during which RSTP BPDUs are sent), and RSTP BPDUs are sent. While this timer is active, the switch processes all BPDUs received on that port and ignores the protocol type.

If the switch receives an IEEE 802.1D BPDU after the port migration-delay timer has expired, it assumes that it is connected to an IEEE 802.1D switch and starts using only IEEE 802.1D BPDUs. However, if the RSTP switch is using IEEE 802.1D BPDUs on a port and receives an RSTP BPDU after the timer has expired, it restarts the timer and starts using RSTP BPDUs on that port.

Default MSTP Settings

Table 44 Default MSTP Settings

Feature	Default Setting
Spanning-tree mode	PVST+ (Rapid PVST+ and MSTP are disabled)
Switch priority (configurable on a per-CIST port basis)	32768
Spanning-tree port priority (configurable on a per-CIST port basis)	128
Spanning-tree port cost (configurable on a per-CIST port basis)	1000 Mbps: 4 100 Mbps: 19 10 Mbps: 100
Hello time	2 seconds
Forward-delay time	15 seconds
Maximum-aging time	20 seconds
Maximum hop count	20 hops

MSTP Configuration Guidelines

These are the configuration guidelines for MSTP:

- When you enable MST by using the **spanning-tree mode mst** global configuration command, RSTP is automatically enabled.
- For two or more switches to be in the same MST region, they must have the same VLAN-to-instance map, the same configuration revision number, and the same name.
- The switch supports up to 65 MST instances. The number of VLANs that can be mapped to a particular MST instance is unlimited.
- PVST+, rapid PVST+, and MSTP are supported, but only one version can be active at any time. (For example, all VLANs run PVST+, all VLANs run rapid PVST+, or all VLANs run MSTP.) For more information, see "Spanning-Tree Interoperability and Backward Compatibility" section on page 10.
- VTP propagation of the MST configuration is not supported. However, you can manually configure the MST configuration (region name, revision number, and VLAN-to-instance mapping) on each switch within the MST region by using the command-line interface (CLI) or through the SNMP support.
- For load balancing across redundant paths in the network to work, all VLAN-to-instance mapping assignments must match; otherwise, all traffic flows on a single link.
- All MST boundary ports must be forwarding for load balancing between a PVST+ and an MST cloud or between a rapid-PVST+ and an MST cloud. For this to occur, the IST master of the MST cloud should also be the root of the CST. If the MST cloud consists of multiple MST regions, one of the MST regions must contain the CST root, and all of the other MST regions must have a better path to the root contained within the MST cloud than a path through the PVST+ or rapid-PVST+ cloud. You might have to manually configure the switches in the clouds.

- Partitioning the network into a large number of regions is not recommended. However, if this situation is unavoidable, we recommend that you partition the switched LAN into smaller LANs interconnected by routers or non-Layer 2 devices.
- For configuration information about UplinkFast and BackboneFast, see "Information About Configuring the Optional Spanning-Tree Features" section on page 1.

MST Region Configuration and Enabling MSTP

For two or more switches to be in the same MST region, they must have the same VLAN-to-instance mapping, the same configuration revision number, and the same name.

A region can have one member or multiple members with the same MST configuration; each member must be capable of processing RSTP BPDUs. There is no limit to the number of MST regions in a network, but each region can only support up to 65 spanning-tree instances. You can assign a VLAN to only one spanning-tree instance at a time.

Root Switch

The switch maintains a spanning-tree instance for the group of VLANs mapped to it. A switch ID, consisting of the switch priority and the switch MAC address, is associated with each instance. For a group of VLANs, the switch with the lowest switch ID becomes the root switch.

To configure a switch to become the root, use the **spanning-tree mst** *instance-id* **root** global configuration command to modify the switch priority from the default value (32768) to a significantly lower value so that the switch becomes the root switch for the specified spanning-tree instance. When you enter this command, the switch checks the switch priorities of the root switches. Because of the extended system ID support, the switch sets its own priority for the specified instance to 24576 if this value will cause this switch to become the root for the specified spanning-tree instance.

If any root switch for the specified instance has a switch priority lower than 24576, the switch sets its own priority to 4096 less than the lowest switch priority. (4096 is the value of the least-significant bit of a 4-bit switch priority value as shown in Table 1 on page 4.)

If your network consists of switches that both do and do not support the extended system ID, it is unlikely that the switch with the extended system ID support will become the root switch. The extended system ID increases the switch priority value every time the VLAN number is greater than the priority of the connected switches running older software.

The root switch for each spanning-tree instance should be a backbone or distribution switch. Do not configure an access switch as the spanning-tree primary root.

Use the **diameter** keyword, which is available only for MST instance 0, to specify the Layer 2 network diameter (that is, the maximum number of switch hops between any two end stations in the Layer 2 network). When you specify the network diameter, the switch automatically sets an optimal hello time, forward-delay time, and maximum-age time for a network of that diameter, which can significantly reduce the convergence time. You can use the **hello** keyword to override the automatically calculated hello time.

Secondary Root Switch

When you configure a switch with the extended system ID support as the secondary root, the switch priority is modified from the default value (32768) to 28672. The switch is then likely to become the root switch for the specified instance if the primary root switch fails. This is assuming that the other network switches use the default switch priority of 32768 and therefore are unlikely to become the root switch.

You can execute this command on more than one switch to configure multiple backup root switches. Use the same network diameter and hello-time values that you used when you configured the primary root switch with the **spanning-tree mst** *instance-id* **root primary** global configuration command.

Port Priority

If a loop occurs, the MSTP uses the port priority when selecting an interface to put into the forwarding state. You can assign higher priority values (lower numerical values) to interfaces that you want selected first and lower priority values (higher numerical values) that you want selected last. If all interfaces have the same priority value, the MSTP puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.

Path Cost

The MSTP path cost default value is derived from the media speed of an interface. If a loop occurs, the MSTP uses cost when selecting an interface to put in the forwarding state. You can assign lower cost values to interfaces that you want selected first and higher cost values that you want selected last. If all interfaces have the same cost value, the MSTP puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.

Link Type to Ensure Rapid Transitions

If you connect a port to another port through a point-to-point link and the local port becomes a designated port, the RSTP negotiates a rapid transition with the other port by using the proposal-agreement handshake to ensure a loop-free topology as described in the Rapid Convergence, page 8.

By default, the link type is controlled from the duplex mode of the interface: a full-duplex port is considered to have a point-to-point connection; a half-duplex port is considered to have a shared connection. If you have a half-duplex link physically connected point-to-point to a single port on a remote switch running MSTP, you can override the default setting of the link type and enable rapid transitions to the forwarding state.

Neighbor Type

A topology could contain both prestandard and IEEE 802.1s standard compliant devices. By default, ports can automatically detect prestandard devices, but they can still receive both standard and prestandard BPDUs. When there is a mismatch between a device and its neighbor, only the CIST runs on the interface.

You can choose to set a port to send only prestandard BPDUs. The prestandard flag appears in all the show commands, even if the port is in STP compatibility mode.

Restarting the Protocol Migration Process

A switch running MSTP supports a built-in protocol migration mechanism that enables it to interoperate with legacy IEEE 802.1D switches. If this switch receives a legacy IEEE 802.1D configuration BPDU (a BPDU with the protocol version set to 0), it sends only IEEE 802.1D BPDUs on that port. An MSTP switch also can detect that a port is at the boundary of a region when it receives a legacy BPDU, an MST BPDU (Version 3) associated with a different region, or an RST BPDU (Version 2).

However, the switch does not automatically revert to the MSTP mode if it no longer receives IEEE 802.1D BPDUs because it cannot detect whether the legacy switch has been removed from the link unless the legacy switch is the designated switch. A switch also might continue to assign a boundary role to a port when the switch to which it is connected has joined the region.

How to Configure MSTP

Specifying the MST Region Configuration and Enabling MSTP

This task is required.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**spanning-tree mst configuration**	Enters MST configuration mode.
3.	**instance** *instance-id* **vlan** *vlan-range*	Maps VLANs to an MST instance. ■ *instance-id*–range is 0 to 4096. ■ **vlan** *vlan-range*–range is 1 to 4096. When you map VLANs to an MST instance, the mapping is incremental, and the VLANs specified in the command are added to or removed from the VLANs that were previously mapped. To specify a VLAN range, use a hyphen; for example, **instance 1 vlan 1-63** maps VLANs 1 through 63 to MST instance 1. To specify a VLAN series, use a comma; for example, **instance 1 vlan 10, 20, 30** maps VLANs 10, 20, and 30 to MST instance 1.
4.	**name** *name*	Specifies the configuration name. The *name* string has a maximum length of 32 characters and is case sensitive.
5.	**revision** *version*	Specifies the configuration revision number. The range is 0 to 65535.
6.	**show pending**	Verifies your configuration by displaying the pending configuration.
7.	**exit**	Applies all changes, and returns to global configuration mode.
8.	**spanning-tree mode mst**	Enables MSTP. RSTP is also enabled. **Caution: Changing spanning-tree modes can disrupt traffic because all spanning-tree instances are stopped for the previous mode and restarted in the new mode.** You cannot run both MSTP and PVST+ or both MSTP and rapid PVST+ at the same time.
9.	**end**	Returns to privileged EXEC mode.

Configuring the Root Switch

Before You Begin

After configuring the switch as the root switch, we recommend that you avoid manually configuring the hello time, forward-delay time, and maximum-age time through the **spanning-tree mst hello-time**, **spanning-tree mst forward-time**, and the **spanning-tree mst max-age** global configuration commands.

This task is optional.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**spanning-tree mst** *instance-id* **root primary** [**diameter** *net-diameter* [**hello-time** *seconds*]]	Configures a switch as the root switch. ■ *instance-id*–Specifies a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 0 to 4096. ■ (Optional) **diameter** *net-diameter*–Specifies the maximum number of switches between any two end stations. The range is 2 to 7. This keyword is available only for MST instance 0. ■ (Optional) **hello-time** *seconds*–Specifies the interval in seconds between the generation of configuration messages by the root switch. The range is 1 to 10 seconds; the default is 2 seconds.
3.	**spanning-tree mst** *instance-id* **root secondary** [**diameter** *net-diameter* [**hello-time** *seconds*]]	Configures a switch as the secondary root switch. ■ *instance-id*–Specifies a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 0 to 4096. ■ (Optional) **diameter** *net-diameter*–Specifies the maximum number of switches between any two end stations. The range is 2 to 7. This keyword is available only for MST instance 0. ■ (Optional) **hello-time** *seconds*–Specifies the interval in seconds between the generation of configuration messages by the root switch. The range is 1 to 10 seconds; the default is 2 seconds. Use the same network diameter and hello-time values that you used when configuring the primary root switch.
4.	**end**	Returns to privileged EXEC mode.

Configuring the Optional MSTP Parameters

Before You Begin

Exercise care when configuring the switch priority. For most situations, we recommend that you use the **spanning-tree mst** *instance-id* **root primary** and the **spanning-tree mst** *instance-id* **root secondary** global configuration commands to modify the switch priority.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**spanning-tree mst** *instance-id* **priority** *priority*	Configures the switch priority. ■ *instance-id*–Specifies a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 0 to 4096. ■ *priority*–The range is 0 to 61440 in increments of 4096; the default is 32768. The lower the number, the more likely the switch will be chosen as the root switch. Priority values are 0, 4096, 8192, 12288, 16384, 20480, 24576, 28672, 32768, 36864, 40960, 45056, 49152, 53248, 57344, and 61440. All other values are rejected.
3.	**spanning-tree mst hello-time** *seconds*	Configures the hello time for all MST instances. The hello time is the interval between the generation of configuration messages by the root switch. These messages mean that the switch is alive. *seconds*–The range is 1 to 10; the default is 2.
4.	**spanning-tree mst forward-time** *seconds*	Configures the forward time for all MST instances. The forward delay is the number of seconds a port waits before changing from its spanning-tree learning and listening states to the forwarding state. *seconds*–The range is 4 to 30; the default is 15.
5.	**spanning-tree mst max-age** *seconds*	Configures the maximum-aging time for all MST instances. The maximum-aging time is the number of seconds a switch waits without receiving spanning-tree configuration messages before attempting a reconfiguration. *seconds*–The range is 6 to 40; the default is 20.
6.	**spanning-tree mst max-hops** *hop-count*	Specifies the number of hops in a region before the BPDU is discarded, and the information held for a port is aged. *hop-count*–The range is 1 to 255; the default is 20.
7.	**interface** *interface-id*	Specifies an interface to configure, and enters interface configuration mode. Valid interfaces include physical ports and port-channel logical interfaces.

	Command	Purpose
8.	**spanning-tree mst** *instance-id* **port-priority** *priority*	Configures the port priority. ■ *instance-id*–Specifies a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 0 to 4096. ■ *priority*–The range is 0 to 240 in increments of 16. The default is 128. The lower the number, the higher the priority. The priority values are 0, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 176, 192, 208, 224, and 240. All other values are rejected.
9.	**spanning-tree mst** *instance-id* **cost** *cost*	Configures the cost. If a loop occurs, the MSTP uses the path cost when selecting an interface to place into the forwarding state. A lower path cost represents higher-speed transmission. ■ *instance-id*–Specifies a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 0 to 4096. ■ *cost*–The range is 1 to 200000000; the default value is derived from the media speed of the interface.
10.	**spanning-tree link-type point-to-point**	Specifies that the link type of a port is point-to-point.
11.	**spanning-tree mst pre-standard**	Specifies that the port can send only prestandard BPDUs.
12.	**end**	Returns to privileged EXEC mode.

Monitoring and Maintaining MSTP

Command	Purpose
show spanning-tree mst configuration	Displays the MST region configuration.
show spanning-tree mst configuration digest	Displays the MD5 digest included in the current MSTCI.
show spanning-tree mst *instance-id*	Displays MST information for the specified instance.
show spanning-tree mst interface *interface-id*	Displays MST information for the specified interface.
clear spanning-tree detected-protocols	Restarts the protocol migration process (forces the renegotiation with neighboring switches) on the switch,
clear spanning-tree detected-protocols interface *interface-id*	Restarts the protocol migration process on a specific interface.
show running-config	Verifies your entries.
copy running-config startup-config	Saves your entries in the configuration file.

Configuration Examples for Configuring MSTP

Configuring the MST Region: Example

This example shows how to enter MST configuration mode, map VLANs 10 to 20 to MST instance 1, name the region *region1*, set the configuration revision to 1, display the pending configuration, apply the changes, and return to global configuration mode:

```
Switch(config)# spanning-tree mst configuration
Switch(config-mst)# instance 1 vlan 10-20
Switch(config-mst)# name region1
Switch(config-mst)# revision 1
Switch(config-mst)# show pending
Pending MST configuration
Name      [region1]
Revision  1
Instance  Vlans Mapped
--------  ---------------------
0         1-9,21-4096
1         10-20
-------------------------------

Switch(config-mst)# exit
Switch(config)#
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*
PVST+ and rapid PVST+ configuration	Chapter 19, “Configuring VLANs”
Optional Spanning-Tree configuration	Chapter 24, “Configuring Optional Spanning-Tree Features”
Supported number of spanning-tree instances	Chapter 22, “Supported Spanning-Tree Instances”

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Configuring Optional Spanning-Tree Features

Prerequisites for the Optional Spanning-Tree Features

You can configure all of these features when your switch is running the per-VLAN spanning-tree plus (PVST+). You can configure only the noted features when your switch is running the Multiple Spanning Tree Protocol (MSTP) or the rapid per-VLAN spanning-tree plus (rapid-PVST+) protocol.

Restrictions for the Optional Spanning-Tree Features

You can configure the UplinkFast or the BackboneFast feature for rapid PVST+ or for the MSTP, but the feature remains disabled (inactive) until you change the spanning-tree mode to PVST+.

Information About Configuring the Optional Spanning-Tree Features

PortFast

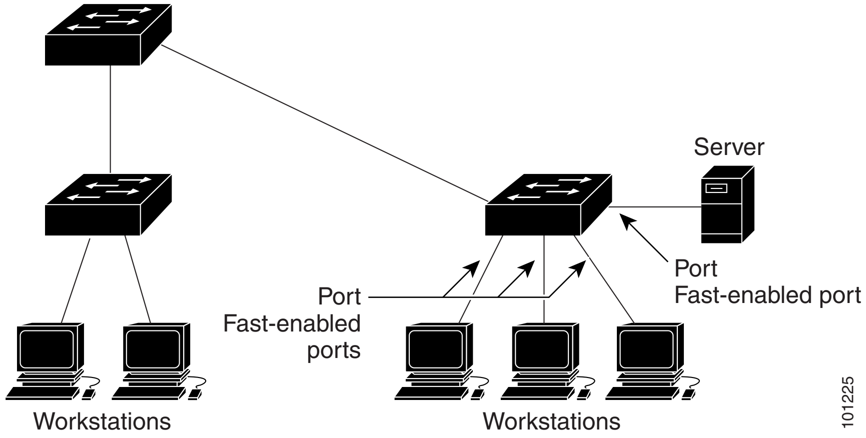
PortFast immediately brings an interface configured as an access or trunk port to the forwarding state from a blocking state, bypassing the listening and learning states. You can use PortFast on interfaces connected to a single workstation or server, as shown in Figure 44 on page 372, to allow those devices to immediately connect to the network, rather than waiting for the spanning tree to converge.

Interfaces connected to a single workstation or server should not receive bridge protocol data units (BPDUs). An interface with PortFast enabled goes through the normal cycle of spanning-tree status changes when the switch is restarted.

Note: Because the purpose of PortFast is to minimize the time interfaces must wait for spanning-tree to converge, it is effective only when used on interfaces connected to end stations. If you enable PortFast on an interface connecting to another switch, you risk creating a spanning-tree loop.

You can enable this feature by using the **spanning-tree portfast** interface configuration or the **spanning-tree portfast default** global configuration command.

Figure 44 PortFast-Enabled Interfaces



BPDU Guard

The BPDU guard feature can be globally enabled on the switch or can be enabled per port, but the feature operates with some differences.

At the global level, you enable BPDU guard on PortFast-enabled ports by using the **spanning-tree portfast bpduguard default** global configuration command. Spanning tree shuts down ports that are in a PortFast-operational state if any BPDU is received on them. In a valid configuration, PortFast-enabled ports do not receive BPDUs. Receiving a BPDU on a PortFast-enabled port means an invalid configuration, such as the connection of an unauthorized device, and the BPDU guard feature puts the port in the error-disabled state. When this happens, the switch shuts down the entire port on which the violation occurred.

To prevent the port from shutting down, you can use the **errdisable detect cause bpduguard shutdown vlan** global configuration command to shut down just the offending VLAN on the port where the violation occurred.

At the interface level, you enable BPDU guard on any port by using the **spanning-tree bpduguard enable** interface configuration command without also enabling the PortFast feature. When the port receives a BPDU, it is put in the error-disabled state.

The BPDU guard feature provides a secure response to invalid configurations because you must manually put the interface back in service. Use the BPDU guard feature in a service-provider network to prevent an access port from participating in the spanning tree.

Enabling BPDU Guard

When you globally enable BPDU guard on ports that are Port Fast-enabled (the ports are in a Port Fast-operational state), spanning tree continues to run on the ports. They remain up unless they receive a BPDU.

In a valid configuration, Port Fast-enabled ports do not receive BPDUs. Receiving a BPDU on a Port Fast-enabled port means an invalid configuration, such as the connection of an unauthorized device, and the BPDU guard feature puts the port in the error-disabled state. When this happens, the switch shuts down the entire port on which the violation occurred.

To prevent the port from shutting down, you can use the **errdisable detect cause bpduguard shutdown vlan** global configuration command to shut down just the offending VLAN on the port where the violation occurred.

The BPDU guard feature provides a secure response to invalid configurations because you must manually put the port back in service. Use the BPDU guard feature in a service-provider network to prevent an access port from participating in the spanning tree.

Caution: Configure Port Fast only on ports that connect to end stations; otherwise, an accidental topology loop could cause a data packet loop and disrupt switch and network operation.

You also can use the **spanning-tree bpduguard enable** interface configuration command to enable BPDU guard on any port without also enabling the Port Fast feature. When the port receives a BPDU, it is put it in the error-disabled state.

BPDU Filtering

The BPDU filtering feature can be globally enabled on the switch or can be enabled per interface, but the feature operates with some differences.

At the global level, you can enable BPDU filtering on PortFast-enabled interfaces by using the **spanning-tree portfast bpdufilter default** global configuration command. This command prevents interfaces that are in a PortFast-operational state from sending or receiving BPDUs. The interfaces still send a few BPDUs at link-up before the switch begins to filter outbound BPDUs. You should globally enable BPDU filtering on a switch so that hosts connected to these interfaces do not receive BPDUs. If a BPDU is received on a PortFast-enabled interface, the interface loses its PortFast-operational status, and BPDU filtering is disabled.

At the interface level, you can enable BPDU filtering on any interface by using the **spanning-tree bpdufilter enable** interface configuration command without also enabling the PortFast feature. This command prevents the interface from sending or receiving BPDUs.

Caution: Enabling BPDU filtering on an interface is the same as disabling spanning tree on it and can result in spanning-tree loops.

You can enable the BPDU filtering feature for the entire switch or for an interface.

Enabling BPDU Filtering

When you globally enable BPDU filtering on Port Fast-enabled interfaces, it prevents interfaces that are in a Port Fast-operational state from sending or receiving BPDUs. The interfaces still send a few BPDUs at link-up before the switch begins to filter outbound BPDUs. You should globally enable BPDU filtering on a switch so that hosts connected to these interfaces do not receive BPDUs. If a BPDU is received on a Port Fast-enabled interface, the interface loses its Port Fast-operational status, and BPDU filtering is disabled.

Caution: Configure Port Fast only on interfaces that connect to end stations; otherwise, an accidental topology loop could cause a data packet loop and disrupt switch and network operation.

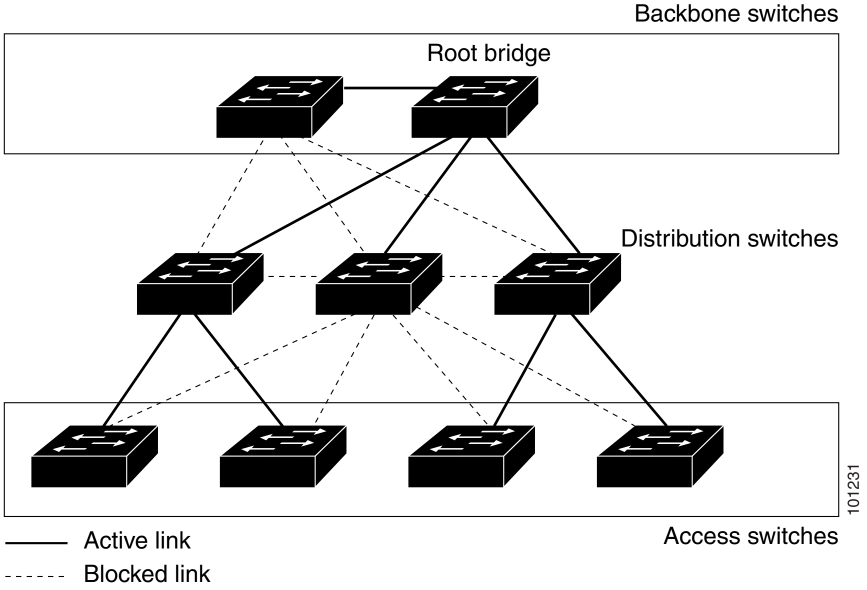
You can also use the **spanning-tree bpdufilter enable** interface configuration command to enable BPDU filtering on any interface without also enabling the Port Fast feature. This command prevents the interface from sending or receiving BPDUs.

Caution: Enabling BPDU filtering on an interface is the same as disabling spanning tree on it and can result in spanning-tree loops.

UplinkFast

Switches in hierarchical networks can be grouped into backbone switches, distribution switches, and access switches. Figure 45 on page 374 shows a complex network where distribution switches and access switches each have at least one redundant link that spanning tree blocks to prevent loops.

Figure 45 Switches in a Hierarchical Network



If a switch loses connectivity, it begins using the alternate paths as soon as the spanning tree selects a new root port. By enabling UplinkFast with the **spanning-tree uplinkfast** global configuration command, you can accelerate the choice of a new root port when a link or switch fails or when the spanning tree reconfigures itself. The root port transitions to the forwarding state immediately without going through the listening and learning states, as it would with the normal spanning-tree procedures.

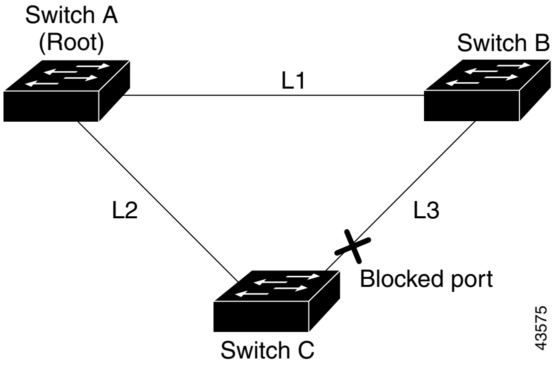
When the spanning tree reconfigures the new root port, other interfaces flood the network with multicast packets, one for each address that was learned on the interface. You can limit these bursts of multicast traffic by reducing the max-update-rate parameter (the default for this parameter is 150 packets per second). However, if you enter zero, station-learning frames are not generated, so the spanning-tree topology converges more slowly after a loss of connectivity.

Note: UplinkFast is most useful in wiring-closet switches at the access or edge of the network. It is not appropriate for backbone devices. This feature might not be useful for other types of applications.

UplinkFast provides fast convergence after a direct link failure and achieves load balancing between redundant Layer 2 links using uplink groups. An uplink group is a set of Layer 2 interfaces (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, except for self-looping ports. The uplink group provides an alternate path in case the currently forwarding link fails.

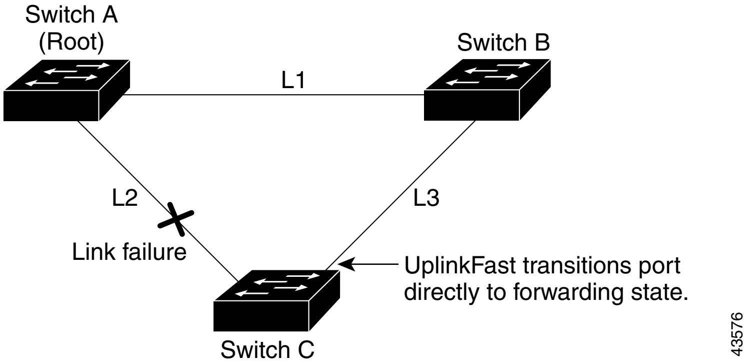
Figure 46 on page 375 shows an example topology with no link failures. Switch A, the root switch, is connected directly to Switch B over link L1 and to Switch C over link L2. The Layer 2 interface on Switch C that is connected directly to Switch B is in a blocking state.

Figure 46 UplinkFast Example Before Direct Link Failure



If Switch C detects a link failure on the currently active link L2 on the root port (a *direct* link failure), UplinkFast unblocks the blocked interface on Switch C and transitions it to the forwarding state without going through the listening and learning states, as shown in Figure 47 on page 375. This change takes approximately 1 to 5 seconds.

Figure 47 UplinkFast Example After Direct Link Failure



Enabling UplinkFast for Use with Redundant Links

UplinkFast cannot be enabled on VLANs that have been configured with a switch priority. To enable UplinkFast on a VLAN with switch priority configured, first restore the switch priority on the VLAN to the default value by using the **no spanning-tree vlan** *vlan-id* **priority** global configuration command.

Note: When you enable UplinkFast, it affects all VLANs on the switch. You cannot configure UplinkFast on an individual VLAN.

You can configure the UplinkFast feature for rapid PVST+ or for the MSTP, but the feature remains disabled (inactive) until you change the spanning-tree mode to PVST+.

When UplinkFast is enabled, the switch priority of all VLANs is set to 49152. If you change the path cost to a value less than 3000 and you enable UplinkFast or UplinkFast is already enabled, the path cost of all interfaces and VLAN trunks is increased by 3000 (if you change the path cost to 3000 or above, the path cost is not altered). The changes to the switch priority and the path cost reduce the chance that a switch will become the root switch.

When UplinkFast is disabled, the switch priorities of all VLANs and path costs of all interfaces are set to default values if you did not modify them from their defaults.

BackboneFast

BackboneFast detects indirect failures in the core of the backbone. BackboneFast is a complementary technology to the UplinkFast feature, which responds to failures on links directly connected to access switches. BackboneFast optimizes the maximum-age timer, which controls the amount of time the switch stores protocol information received on an interface. When a switch receives an inferior BPDU from the designated port of another switch, the BPDU is a signal that the other switch might have lost its path to the root, and BackboneFast tries to find an alternate path to the root.

BackboneFast, which is enabled by using the **spanning-tree backbonefast** global configuration command, starts when a root port or blocked interface on a switch receives inferior BPDUs from its designated switch. An inferior BPDU identifies a switch that declares itself as both the root bridge and the designated switch. When a switch receives an inferior BPDU, it means that a link to which the switch is not directly connected (an *indirect* link) has failed (that is, the designated switch has lost its connection to the root switch). Under spanning-tree rules, the switch ignores inferior BPDUs for the configured maximum aging time specified by the **spanning-tree vlan** *vlan-id max-age* global configuration command.

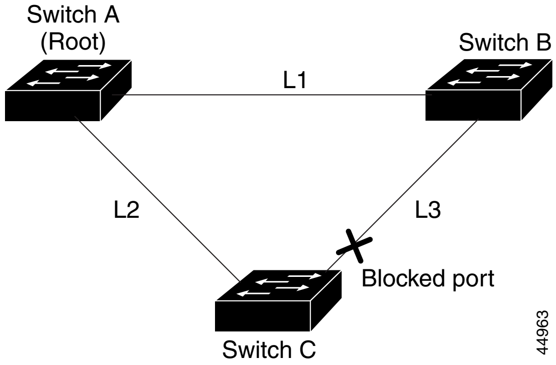
The switch tries to find if it has an alternate path to the root switch. If the inferior BPDU arrives on a blocked interface, the root port and other blocked interfaces on the switch become alternate paths to the root switch. (Self-looped ports are not considered alternate paths to the root switch.) If the inferior BPDU arrives on the root port, all blocked interfaces become alternate paths to the root switch. If the inferior BPDU arrives on the root port and there are no blocked interfaces, the switch assumes that it has lost connectivity to the root switch, causes the maximum aging time on the root port to expire, and becomes the root switch according to normal spanning-tree rules.

If the switch has alternate paths to the root switch, it uses these alternate paths to send a root link query (RLQ) request. The switch sends the RLQ request on all alternate paths and waits for an RLQ reply from other switches in the network.

If the switch discovers that it still has an alternate path to the root, it expires the maximum aging time on the interface that received the inferior BPDU. If all the alternate paths to the root switch indicate that the switch has lost connectivity to the root switch, the switch expires the maximum aging time on the interface that received the RLQ reply. If one or more alternate paths can still connect to the root switch, the switch makes all interfaces on which it received an inferior BPDU its designated ports and moves them from the blocking state (if they were in the blocking state), through the listening and learning states, and into the forwarding state.

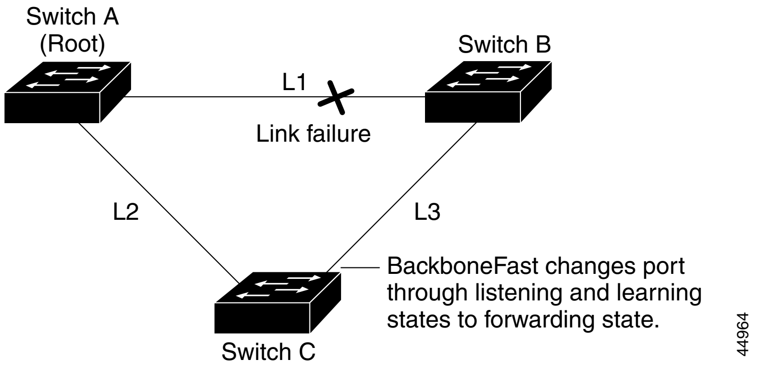
Figure 48 on page 376 shows an example topology with no link failures. Switch A, the root switch, connects directly to Switch B over link L1 and to Switch C over link L2. The Layer 2 interface on Switch C that connects directly to Switch B is in the blocking state.

Figure 48 BackboneFast Example Before Indirect Link Failure



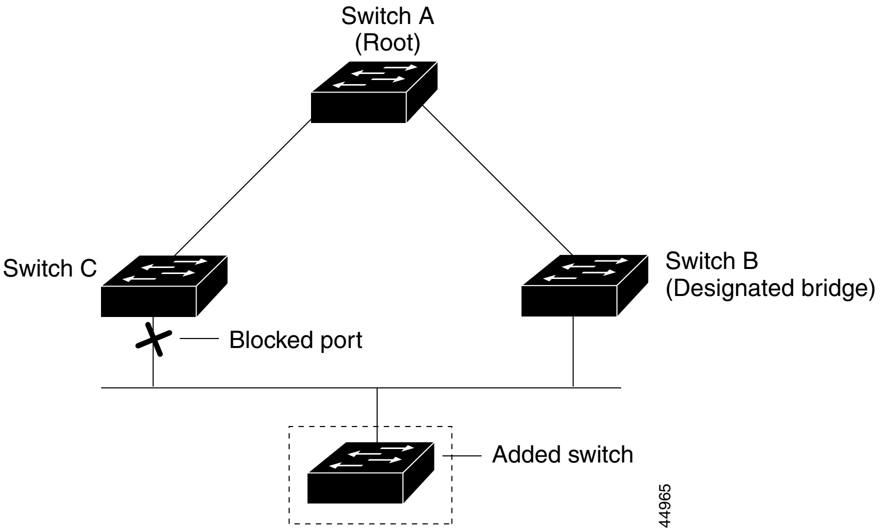
If link L1 fails as shown in Figure 49 on page 377, Switch C cannot detect this failure because it is not connected directly to link L1. However, because Switch B is directly connected to the root switch over L1, it detects the failure, elects itself the root, and begins sending BPDUs to Switch C, identifying itself as the root. When Switch C receives the inferior BPDUs from Switch B, Switch C assumes that an indirect failure has occurred. At that point, BackboneFast allows the blocked interface on Switch C to move immediately to the listening state without waiting for the maximum aging time for the interface to expire. BackboneFast then transitions the Layer 2 interface on Switch C to the forwarding state, providing a path from Switch B to Switch A. The root-switch election takes approximately 30 seconds, twice the Forward Delay time if the default Forward Delay time of 15 seconds is set. Figure 49 on page 377 shows how BackboneFast reconfigures the topology to account for the failure of link L1.

Figure 49 BackboneFast Example After Indirect Link Failure


Switch A
(Root)
Switch B
L1
Link failure
L2
L3
BackboneFast changes port through listening and learning states to forwarding state.
Switch C
44964

If a new switch is introduced into a shared-medium topology as shown in Figure 50 on page 377, BackboneFast is not activated because the inferior BPDUs did not come from the recognized designated switch (Switch B). The new switch begins sending inferior BPDUs that indicate it is the root switch. However, the other switches ignore these inferior BPDUs, and the new switch learns that Switch B is the designated switch to Switch A, the root switch.

Figure 50 Adding a Switch in a Shared-Medium Topology



Enabling BackboneFast

You can enable BackboneFast to detect indirect link failures and to start the spanning-tree reconfiguration sooner.

Note: If you use BackboneFast, 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.

EtherChannel Guard

You can use EtherChannel guard to detect an EtherChannel misconfiguration between the switch and a connected device. A misconfiguration can occur if the switch interfaces are configured in an EtherChannel, but the interfaces on the other device are not. A misconfiguration can also occur if the channel parameters are not the same at both ends of the EtherChannel.

If the switch detects a misconfiguration on the other device, EtherChannel guard places the switch interfaces in the error-disabled state, and displays an error message.

You can enable this feature by using the **spanning-tree etherchannel guard misconfig** global configuration command.

Root Guard

The Layer 2 network of a service provider (SP) can include many connections to switches that are not owned by the SP. In such a topology, the spanning tree can reconfigure itself and select a *customer switch* as the root switch, as shown in Figure 51 on page 379. You can avoid this situation by enabling root guard on SP switch interfaces that connect to switches in your customer's network. If spanning-tree calculations cause an interface in the customer network to be selected as the root port, root guard then places the interface in the root-inconsistent (blocked) state to prevent the customer's switch from becoming the root switch or being in the path to the root.

If a switch outside the SP network becomes the root switch, the interface is blocked (root-inconsistent state), and spanning tree selects a new root switch. The customer's switch does not become the root switch and is not in the path to the root.

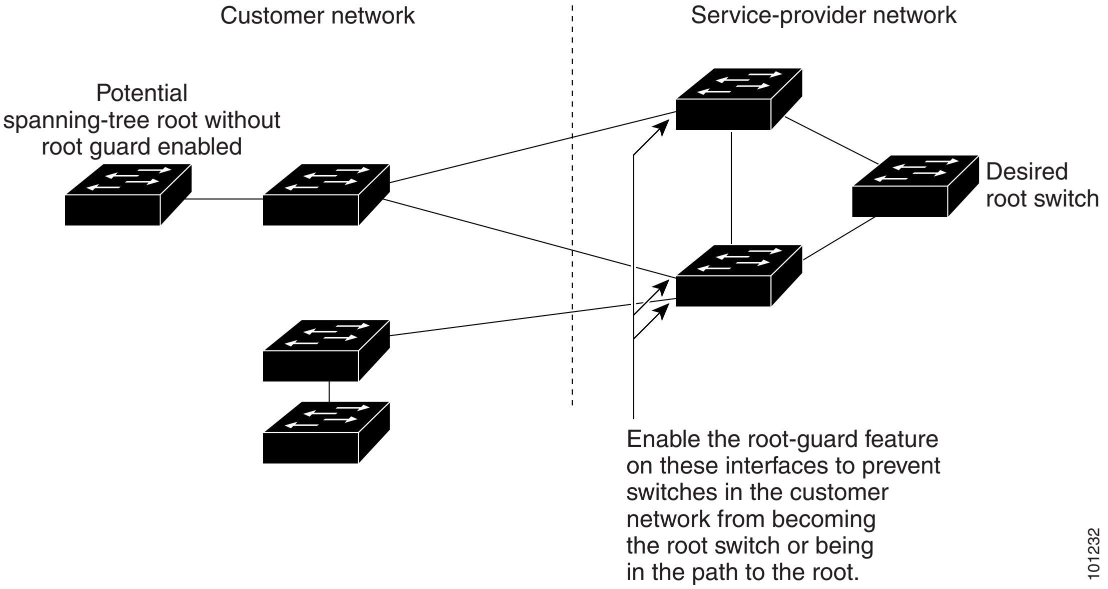
If the switch is operating in multiple spanning-tree (MST) mode, root guard forces the interface to be a designated port. If a boundary port is blocked in an internal spanning-tree (IST) instance because of root guard, the interface also is blocked in all MST instances. A boundary port is an interface that connects to a LAN, the designated switch of which is either an IEEE 802.1D switch or a switch with a different MST region configuration.

Root guard enabled on an interface applies to all the VLANs to which the interface belongs. VLANs can be grouped and mapped to an MST instance.

You can enable this feature by using the **spanning-tree guard root** interface configuration command.

Caution: Misuse of the root guard feature can cause a loss of connectivity.

Figure 51 Root Guard in a Service-Provider Network


Customer network
Service-provider network
Potential spanning-tree root without root guard enabled
Desired root switch
Enable the root-guard feature on these interfaces to prevent switches in the customer network from becoming the root switch or being in the path to the root.
101232

Enabling Root Guard

Root guard enabled on an interface applies to all the VLANs to which the interface belongs. Do not enable the root guard on interfaces to be used by the UplinkFast feature. With UplinkFast, the backup interfaces (in the blocked state) replace the root port in the case of a failure. However, if root guard is also enabled, all the backup interfaces used by the UplinkFast feature are placed in the root-inconsistent state (blocked) and are prevented from reaching the forwarding state.

Note: You cannot enable both root guard and loop guard at the same time.

Loop Guard

You can use loop guard to prevent alternate or root ports from becoming designated ports because of a failure that leads to a unidirectional link. This feature is most effective when it is enabled on the entire switched network. Loop guard prevents alternate and root ports from becoming designated ports, and spanning tree does not send BPDUs on root or alternate ports.

You can enable this feature by using the **spanning-tree loopguard default** global configuration command.

When the switch is operating in PVST+ or rapid-PVST+ mode, loop guard prevents alternate and root ports from becoming designated ports, and spanning tree does not send BPDUs on root or alternate ports.

When the switch is operating in MST mode, BPDUs are not sent on nonboundary ports only if the interface is blocked by loop guard in all MST instances. On a boundary port, loop guard blocks the interface in all MST instances.

Enabling Loop Guard

You can use loop guard to prevent alternate or root ports from becoming designated ports because of a failure that leads to a unidirectional link. This feature is most effective when it is configured on the entire switched network. Loop guard operates only on interfaces that are considered point-to-point by the spanning tree.

Note: You cannot enable both loop guard and root guard at the same time.

Default Optional Spanning-Tree Settings

Table 45 Default Optional Spanning-Tree Settings

Feature	Default Setting
PortFast, BPDU filtering, BPDU guard	Globally disabled (unless they are individually configured per interface).
UplinkFast	Globally disabled.
BackboneFast	Globally disabled.
EtherChannel guard	Globally enabled.
Root guard	Disabled on all interfaces.
Loop guard	Disabled on all interfaces.

How to Configure the Optional Spanning-Tree Features

Enabling Optional SPT Features

Before You Begin

- Make sure that there are no loops in the network between the trunk port and the workstation or server before you enable PortFast on a trunk port.
- Use PortFast *only* when connecting a single end station to an access or trunk port. Enabling this feature on an interface connected to a switch or hub could prevent spanning tree from detecting and disabling loops in your network, which could cause broadcast storms and address-learning problems.
- An interface with the PortFast feature enabled is moved directly to the spanning-tree forwarding state without waiting for the standard forward-time delay.
- You cannot enable both loop guard and root guard at the same time.
- When you enable UplinkFast, it affects all VLANs on the switch. You cannot configure UplinkFast on an individual VLAN.
- If you enable the voice VLAN feature, the PortFast feature is automatically enabled. When you disable voice VLAN, the PortFast feature is not automatically disabled.

	Command	Purpose
1.	**show spanning-tree active** or **show spanning-tree mst**	Verifies which interfaces are alternate or root ports.
2.	**configure terminal**	Enters global configuration mode.
3.	**spanning-tree loopguard default**	Enables loop guard. By default, loop guard is disabled.
4.	**spanning-tree portfast bpduguard default**	Enables BPDU guard. By default, BPDU guard is disabled.

	Command	Purpose
5.	**spanning-tree portfast bpdufilter default**	Enables BPDU filtering. By default, BPDU filtering is disabled.
6.	**spanning-tree uplinkfast** [**max-update-rate** *pkts-per-second*]	Enables UplinkFast. (Optional) *pkts-per-second*–The range is 0 to 32000 packets per second; the default is 150. If you set the rate to 0, station-learning frames are not generated, and the spanning-tree topology converges more slowly after a loss of connectivity.
7.	**spanning-tree backbonefast**	Enables BackboneFast.
8.	**spanning-tree etherchannel guard misconfig**	Enables EtherChannel guard.
9.	**interface** *interface-id*	Specifies an interface to configure, and enters interface configuration mode.
10.	**spanning-tree portfast** [**trunk**]	Enables PortFast on an access port connected to a single workstation or server. By specifying the **trunk** keyword, you can enable PortFast on a trunk port. **Note:** To enable PortFast on trunk ports, you must use the **spanning-tree portfast trunk** interface configuration command. The **spanning-tree portfast** command will not work on trunk ports. By default, PortFast is disabled on all interfaces.
11.	**spanning-tree guard root**	Enables root guard on the interface. By default, root guard is disabled on all interfaces.
12.	**end**	Returns to privileged EXEC mode.

Maintaining and Monitoring Optional Spanning-Tree Features

Command	Purpose
show spanning-tree active	Displays spanning-tree information on active interfaces only.
show spanning-tree detail	Displays a detailed summary of interface information.
show spanning-tree interface *interface-id*	Displays spanning-tree information for the specified interface.
show spanning-tree mst interface *interface-id*	Displays MST information for the specified interface.
show spanning-tree summary [**totals**]	Displays a summary of interface states or displays the total lines of the spanning-tree state section.
show interfaces status err-disabled	Displays which switch ports are disabled because of an EtherChannel misconfiguration.
show etherchannel summary	Displays the EtherChannel configuration. Useful to use on the remote device after switch ports are disabled.
[**no**] **shutdown**	Disables the interface. The **no** option reenables the interface.

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*
VLAN configuration	*Configuring VLANs, page 289*
Voice VLAN configuration	*Configuring Voice VLAN, page 327*
PVST+ and rapid PVST+ configuratio	*Configuring STP, page 333*
Multiple Spanning Tree Protocol configuration	*Configuring MSTP, page 351*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	–

MIBs

MIBs	MIBs Link
–	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	–

Configuring Resilient Ethernet Protocol

Information About Configuring REP

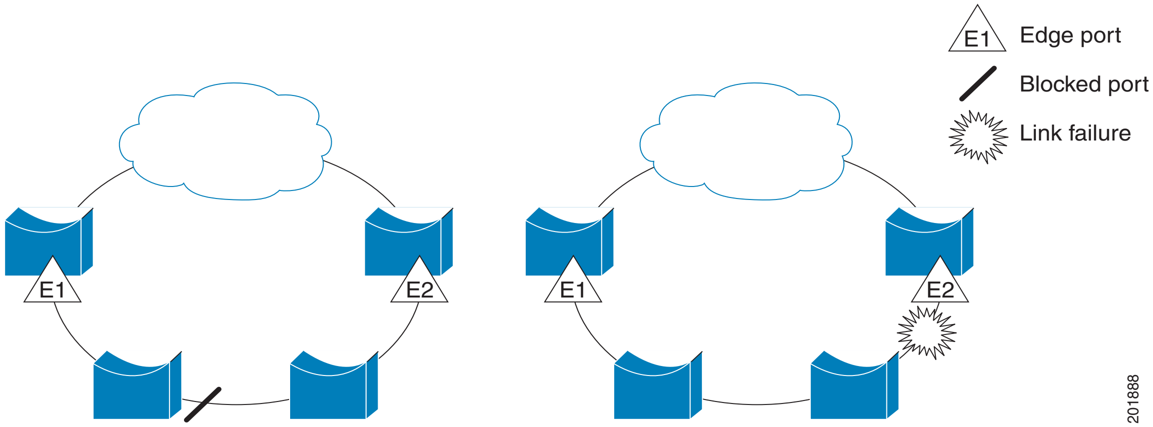
REP

Resilient Ethernet Protocol (REP) is a Cisco proprietary protocol that provides an alternative to Spanning Tree Protocol (STP) to control network loops, handle link failures, and improve convergence time. REP controls a group of ports connected in a segment, ensures that the segment does not create any bridging loops, and responds to link failures within the segment. REP provides a basis for constructing more complex networks and supports VLAN load balancing.

One REP segment is a chain of ports connected to each other and configured with a segment ID. Each segment consists of standard (non-edge) segment ports and two user-configured edge ports. A switch can have no more than two ports that belong to the same segment, and each segment port can have only one external neighbor. A segment can go through a shared medium, but on any link only two ports can belong to the same segment. REP is supported only on Layer 2 trunk interfaces.

Figure 52 on page 383 shows an example of a segment consisting of six ports spread across four switches. Ports E1 and E2 are configured as edge ports. When all ports are operational (as in the segment on the left), a single port is blocked, shown by the diagonal line. When there is a failure in the network, as shown in the diagram on the right, the blocked port returns to the forwarding state to minimize network disruption.

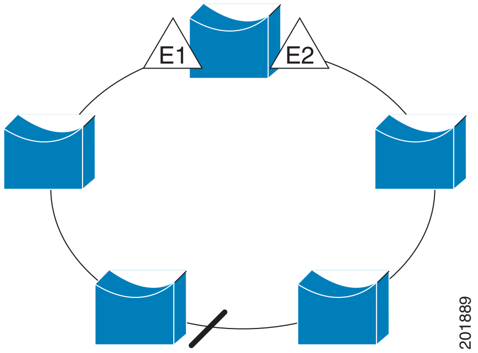
Figure 52 REP Open Segments



The segment shown in Figure 52 on page 383 is an open segment; there is no connectivity between the two edge ports. The REP segment cannot cause a bridging loop and it is safe to connect the segment edges to any network. All hosts connected to switches inside the segment have two possible connections to the rest of the network through the edge ports, but only one connection is accessible at any time. If a failure causes a host to be unable to access its usual gateway, REP unblocks all ports to ensure that connectivity is available through the other gateway.

The segment shown in Figure 53 on page 384, with both edge ports located on the same switch, is a ring segment. In this configuration, there is connectivity between the edge ports through the segment. With this configuration, you can create a redundant connection between any two switches in the segment.

Figure 53 REP Ring Segment



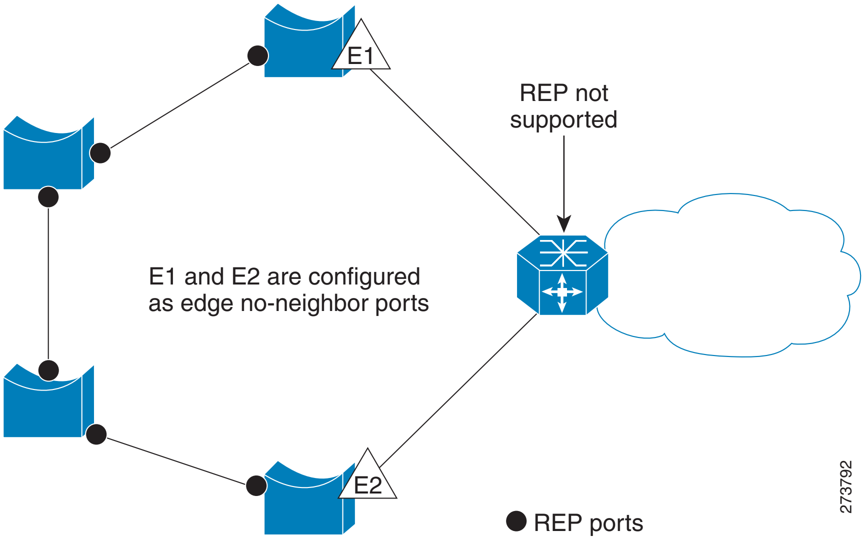
REP segments have these characteristics:

- If all ports in the segment are operational, one port (referred to as the *alternate* port) is in the blocked state for each VLAN. If VLAN load balancing is configured, two ports in the segment control the blocked state of VLANs.
- If one or more ports in a segment is not operational, causing a link failure, all ports forward traffic on all VLANs to ensure connectivity.
- In case of a link failure, the alternate ports are unblocked as quickly as possible. When the failed link comes back up, a logically blocked port per VLAN is selected with minimal disruption to the network.

You can construct almost any type of network based on REP segments. REP also supports VLAN load-balancing, controlled by the primary edge port but occurring at any port in the segment.

In access ring topologies, the neighboring switch might not support REP, as shown in Figure 54 on page 384. In this case, you can configure the non-REP facing ports (E1 and E2) as edge no-neighbor ports. These ports inherit all properties of edge ports, and you can configure them the same as any edge port, including configuring them to send STP or REP topology change notices to the aggregation switch. In this case the STP topology change notice (TCN) that is sent is a multiple spanning-tree (MST) STP message.

Figure 54 Edge No-Neighbor Ports



REP has these limitations:

- You must configure each segment port; an incorrect configuration can cause forwarding loops in the networks.

- REP can manage only a single failed port within the segment; multiple port failures within the REP segment cause loss of network connectivity.
- You should configure REP only in networks with redundancy. Configuring REP in a network without redundancy causes loss of connectivity.

Link Integrity

REP does not use an end-to-end polling mechanism between edge ports to verify link integrity. It implements local link failure detection. The REP Link Status Layer (LSL) detects its REP-aware neighbor and establishes connectivity within the segment. All VLANs are blocked on an interface until it detects the neighbor. After the neighbor is identified, REP determines which neighbor port should become the alternate port and which ports should forward traffic.

Each port in a segment has a unique port ID. The port ID format is similar to that used by the spanning tree algorithm: a port number (unique on the bridge), associated to a MAC address (unique in the network). When a segment port is coming up, its LSL starts sending packets that include the segment ID and the port ID. The port is declared as operational after it performs a three-way handshake with a neighbor in the same segment.

A segment port does not become operational if:

- No neighbor has the same segment ID.
- More than one neighbor has the same segment ID.
- The neighbor does not acknowledge the local port as a peer.

Each port creates an adjacency with its immediate neighbor. Once the neighbor adjacencies are created, the ports negotiate to determine one blocked port for the segment, the alternate port. All other ports become unblocked. By default, REP packets are sent to a BPDU class MAC address. The packets can also be sent to the Cisco multicast address, which is used only to send blocked port advertisement (BPA) messages when there is a failure in the segment. The packets are dropped by devices not running REP.

REP Negotiated

Spanning Tree Protocol (STP) is enabled by default on Cisco switches. If a Cisco switch is inserted in an already running REP ring (for example, to add a new node or replace an existing node), the new switch running STP will cause a break in the REP ring and cannot communicate over the REP ring until it is configured to be part of the ring.

After a new switch is inserted in the ring, it is running STP, but the rest of the ring is running REP. Neither of these protocols can recognize a loop in the ring and keep both ends of the ring in the forwarding state, causing an endless loop. To address this problem, **rep bpduleak** should be configured on the new switch so that REP BPDUs are transparently forwarded between two ring ports on the switch when REP is not configured. This function, called BPDU leaking, causes the REP ring to converge but new devices will not be part of the ring nor be seen in or show the REP topology.

When the switch interfaces are configured with REP Negotiated, REP status is negotiated with the peers. If the peer supports REP, it is migrated to REP. If the peer does not support REP, it is migrated to STP. The peer is migrated to REP or STP using an Embedded Event Manager (EEM) macro.

Note: REP Negotiated works only on uplink ports.

See Configuring REP Negotiated, page 392 for information about configuring REP Negotiated.

Fast Convergence

Because REP runs on a physical link basis and not a per-VLAN basis, only one hello message is required for all VLANs, reducing the load on the protocol. We recommend that you create VLANs consistently on all switches in a given segment and configure the same allowed VLANs on the REP trunk ports. To avoid the delay introduced by relaying messages in software, REP also allows some packets to be flooded to a regular multicast address. These messages operate at the hardware flood layer (HFL) and are flooded to the whole network, not just the REP segment. Switches that do not belong to the segment treat them as data traffic. You can control flooding of these messages by configuring a dedicated administrative VLAN for the whole domain.

The estimated convergence recovery time on fiber interfaces is 50-200 ms for the local segment with 200 VLANs configured. When REP is configured on RJ45 Gigabit copper interfaces, the convergence time is 500-750 ms. Convergence for VLAN load balancing is 300 ms or less.

VLAN Load Balancing

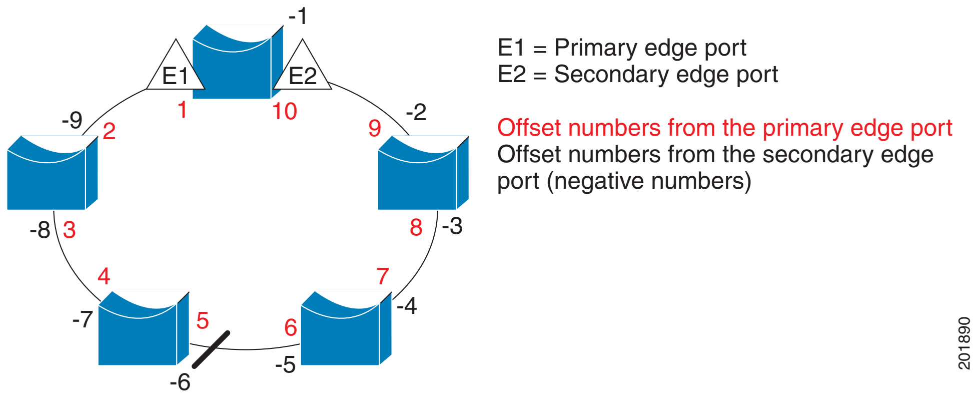
One edge port in the REP segment acts as the primary edge port; the other as the secondary edge port. It is the primary edge port that always participates in VLAN load balancing in the segment. REP VLAN balancing is achieved by blocking some VLANs at a configured alternate port and all other VLANs at the primary edge port. When you configure VLAN load balancing, you can specify the alternate port in one of three ways:

- By entering the port ID of the interface. To identify the port ID of a port in the segment, enter the **show interface rep detail** interface configuration command for the port.
- By entering the neighbor offset number of a port in the segment, which identifies the downstream neighbor port of an edge port. The neighbor offset number range is –256 to +256; a value of 0 is invalid. The primary edge port has an offset number of 1; positive numbers above 1 identify downstream neighbors of the primary edge port. Negative numbers indicate the secondary edge port (offset number -1) and its downstream neighbors.

  Note: You configure offset numbers on the primary edge port by identifying a port's downstream position from the primary (or secondary) edge port. You would never enter an offset value of 1 because that is the offset number of the primary edge port itself.

  Figure 55 on page 386 shows neighbor offset numbers for a segment where E1 is the primary edge port and E2 is the secondary edge port. The red numbers inside the ring are numbers offset from the primary edge port; the black numbers outside of the ring show the offset numbers from the secondary edge port. Note that you can identify all ports (except the primary edge port) by either a positive offset number (downstream position from the primary edge port) or a negative offset number (downstream position from the secondary edge port). If E2 became the primary edge port, its offset number would then be 1 and E1 would be -1.
- By entering the **preferred** keyword to select the port that you previously configured as the preferred alternate port with the **rep segment** *segment-id* **preferred** interface configuration command.

Figure 55 Neighbor Offset Numbers in a Segment


-1
E1
E2
1
10
-9
2
9
-2
-8
3
8
-3
4
7
-7
5
6
-4
-6
-5
E1 = Primary edge port
E2 = Secondary edge port
Offset numbers from the primary edge port
Offset numbers from the secondary edge port (negative numbers)
201890

When the REP segment is complete, all VLANs are blocked. When you configure VLAN load balancing, you must also configure triggers in one of two ways:

- Manually trigger VLAN load balancing at any time by entering the **rep preempt segment** *segment-id* privileged EXEC command on the switch that has the primary edge port.
- Configure a preempt delay time by entering the **rep preempt delay** *seconds* interface configuration command. After a link failure and recovery, VLAN load balancing begins after the configured preemption time period elapses. Note that the delay timer restarts if another port fails before the time has elapsed.

Note: When VLAN load balancing is configured, it does not start working until triggered by either manual intervention or a link failure and recovery.

When VLAN load balancing is triggered, the primary edge port sends out a message to alert all interfaces in the segment about the preemption. When the secondary port receives the message, it is reflected into the network to notify the alternate port to block the set of VLANs specified in the message and to notify the primary edge port to block the remaining VLANs.

You can also configure a particular port in the segment to block all VLANs. Only the primary edge port initiates VLAN load balancing, which is not possible if the segment is not terminated by an edge port on each end. The primary edge port determines the local VLAN load balancing configuration.

Reconfigure the primary edge port to reconfigure load balancing. When you change the load balancing configuration, the primary edge port again waits for the **rep preempt segment** command or for the configured preempt delay period after a port failure and recovery before executing the new configuration. If you change an edge port to a regular segment port, the existing VLAN load balancing status does not change. Configuring a new edge port might cause a new topology configuration.

Spanning Tree Interaction

REP does not interact with STP or with the FlexLink feature, but can coexist with both. A port that belongs to a segment is removed from spanning tree control and STP BPDUs are not accepted or sent from segment ports. Therefore, STP cannot run on a segment.

To migrate from an STP ring configuration to REP segment configuration, begin by configuring a single port in the ring as part of the segment and continue by configuring contiguous ports to minimize the number of segments. Each segment always contains a blocked port, so multiple segments means multiple blocked ports and a potential loss of connectivity. When the segment has been configured in both directions up to the location of the edge ports, you then configure the edge ports.

REP Ports

Ports in REP segments are Failed, Open, or Alternate.

- A port configured as a regular segment port starts as a failed port.
- After the neighbor adjacencies are determined, the port transitions to alternate port state, blocking all VLANs on the interface. Blocked port negotiations occur and when the segment settles, one blocked port remains in the alternate role and all other ports become open ports.
- When a failure occurs in a link, all ports move to the failed state. When the alternate port receives the failure notification, it changes to the open state, forwarding all VLANs.

A regular segment port converted to an edge port, or an edge port converted to a regular segment port, does not always result in a topology change. If you convert an edge port into a regular segment port, VLAN load balancing is not implemented unless it has been configured. For VLAN load balancing, you must configure two edge ports in the segment.

A segment port that is reconfigured as a spanning tree port restarts according the spanning tree configuration. By default, this is a designated blocking port. If PortFast is configured or if STP is disabled, the port goes into the forwarding state.

REP Segments

A segment is a collection of ports connected one to the other in a chain and configured with a segment ID. To configure REP segments, you configure the REP administrative VLAN (or use the default VLAN 1) and then add the ports to the segment using interface configuration mode. You should configure two edge ports in the segment, with one of them the primary edge port and the other by default the secondary edge port. A segment has only one primary edge port. If you configure two ports in a segment as the primary edge port, for example, ports on different switches, the REP selects one of them to serve as the segment primary edge port. You can also optionally configure where to send segment topology change notices (STCNs) and VLAN load balancing.

Default REP Configuration

REP is disabled on all interfaces. When enabled, the interface is a regular segment port unless it is configured as an edge port.

When REP is enabled, the sending of segment topology change notices (STCNs) is disabled, all VLANs are blocked, and the administrative VLAN is VLAN 1.

When VLAN load balancing is enabled, the default is manual preemption with the delay timer disabled. If VLAN load balancing is not configured, the default after manual preemption is to block all VLANs at the primary edge port.

REP Configuration Guidelines

Follow these guidelines when configuring REP:

- We recommend that you begin by configuring one port and then configure the contiguous ports to minimize the number of segments and the number of blocked ports.
- If more than two ports in a segment fail when no external neighbors are configured, one port changes to a forwarding state for the data path to help maintain connectivity during configuration. In the show rep interface privileged EXEC command output, the Port Role for this port shows as *Fail Logical Open*; the Port Role for the other failed port shows as *Fail No Ext Neighbor*. When the external neighbors for the failed ports are configured, the ports go through the alternate port state transitions and eventually go to an open state or remain as the alternate port, based on the alternate port election mechanism.
- REP ports must be Layer 2 trunk ports.
- Be careful when configuring REP through a Telnet connection. Because REP blocks all VLANs until another REP interface sends a message to unblock it, you might lose connectivity to the switch if you enable REP in a Telnet session that accesses the switch through the same interface.
- You cannot run REP and STP or REP and Flex Links on the same segment or interface.
- If you connect an STP network to the REP segment, be sure that the connection is at the segment edge. An STP connection that is not at the edge could cause a bridging loop because STP does not run on REP segments. All STP BPDUs are dropped at REP interfaces.
- You must configure all trunk ports in the segment with the same set of allowed VLANs, or a misconfiguration occurs.
- REP ports follow these rules:
  - There is no limit to the number of REP ports on a switch; however, only two ports on a switch can belong to the same REP segment.
  - If only one port on a switch is configured in a segment, the port should be an edge port.
  - If two ports on a switch belong to the same segment, they must be both edge ports, both regular segment ports, or one regular port and one edge no-neighbor port. An edge port and regular segment port on a switch cannot belong to the same segment.

  - If two ports on a switch belong to the same segment and one is configured as an edge port and one as a regular segment port (a misconfiguration), the edge port is treated as a regular segment port.

- REP interfaces come up in a blocked state and remains in a blocked state until notified that it is safe to unblock. You need to be aware of this to avoid sudden connection losses.

- REP sends all LSL PDUs in untagged frames on the native VLAN. The BPA message sent to the Cisco multicast address is sent on the administration VLAN, which is VLAN 1 by default.

- You can configure how long a REP interface remains up without receiving a hello from a neighbor. You can use the **rep lsl-age-timer** *value* interface configuration command to set the time from 120 ms to 10000 ms. The LSL hello timer is then set to the age-timer value divided by 3. In normal operation, three LSL hellos are sent before the age timer on the peer switch expires and checks for hello messages.

  - In Cisco IOS Release 12.2(52)SE, the LSL age-timer range changed from 3000 to 10000 ms in 500-ms increments to 120 to 10000 ms in 40-ms increments. If the REP neighbor device is not running Cisco IOS release 12.2(52)SE or later, do not configure a timer value less than 3000 ms. Configuring a value less than 3000 ms causes the port to shut down because the neighbor switch does not respond within the requested time period.

  - EtherChannel port channel interfaces do not support LSL age-timer values less than 1000 ms. If you try to configure a value less than 1000 ms on a port channel, you receive an error message and the command is rejected.

- When configuring the REP LSL age timer, make sure that both ends of the link have the same time value configured. Configuring different values on ports at each end of the link results in a REP link flap.

- REP ports cannot be configured as one of these port types:

  - SPAN destination port

  - Tunnel port

  - Access port

- REP is supported on EtherChannels, but not on an individual port that belongs to an EtherChannel.

- There is a maximum of 64 REP segments per switch.

- REP Negotiated can be configured only on the first two uplink ports of the switch:

  - IE3000 and IE4000—GigabitEthernet 1/1 and GigabitEthernet 1/2

  - IE4010 and IE5000 (without 10G ports)—GigabitEthernet 1/25 and GigabitEthernet 1/26

  - IE5000 with 10G ports—TenGigabitEthernet 1/25 and TenGigabitEthernet 1/26

REP Administrative VLAN

To avoid the delay introduced by relaying messages in software for link-failure or VLAN-blocking notification during load balancing, REP floods packets at the hardware flood layer (HFL) to a regular multicast address. These messages are flooded to the whole network, not just the REP segment. You can control flooding of these messages by configuring an administrative VLAN for the whole domain.

Follow these guidelines when configuring the REP administrative VLAN:

- If you do not configure an administrative VLAN, the default is VLAN 1.

- There can be only one administrative VLAN on a switch and on a segment. However, this is not enforced by software.

- The administrative VLAN cannot be the RSPAN VLAN.

How to Configure REP

Configuring the REP Administrative VLAN

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**rep admin vlan** *vlan-id*	Specifies the administrative VLAN. The range is 2 to 4096. The default is VLAN 1. To set the admin VLAN to 1, enter the **no rep admin vlan** global configuration command.
3.	**end**	Returns to privileged EXEC mode.

Configuring REP Interfaces

Before You Begin

For REP operation, you need to enable it on each segment interface and identify the segment ID. This step is required and must be done before other REP configuration. You must also configure a primary and secondary edge port on each segment. All other steps are optional.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the interface, and enters interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface). The port-channel range is 1 to 10.
3.	**switchport mode trunk**	Configures the interface as a Layer 2 trunk port.

	Command	Purpose
4.	**rep segment** *segment-id* [**edge** [**no-neighbor**] [**primary**]] [**preferred**]	Enables REP on the interface, and identifies a segment number. The segment ID range is from 1 to 1024. These optional keywords are available: **Note:** You must configure two edge ports, including one primary edge port for each segment. ■ **edge**–Configures the port as an edge port. Entering **edge** without the **primary** keyword configures the port as the secondary edge port. Each segment has only two edge ports. ■ (Optional) **primary**– Configures the port as the primary edge port, the port on which you can configure VLAN load balancing. ■ (Optional) **no-neighbor–**Configures a port with no external REP neighbors as an edge port. The port inherits all properties of edge ports, and you can configure them the same as any edge port. **Note:** Although each segment can have only one primary edge port, if you configure edge ports on two different switches and enter the **primary** keyword on both switches, the configuration is allowed. However, REP selects only one of these ports as the segment primary edge port. You can identify the primary edge port for a segment by entering the **show rep topology** privileged EXEC command. ■ (Optional) **preferred**–Indicates that the port is the preferred alternate port or the preferred port for VLAN load balancing. **Note:** Configuring a port as preferred does not guarantee that it becomes the alternate port; it merely gives it a slight edge among equal contenders. The alternate port is usually a previously failed port.
5.	**rep stcn** {**interface** *interface-id* | **segment** *id-list* | **stp**}	(Optional) Configures the edge port to send segment topology change notices (STCNs). ■ **interface** *interface-id*–Designates a physical interface or port channel to receive STCNs. ■ **segment** *id-list*–Identifies one or more segments to receive STCNs. The range is 1 to 1024. ■ **stp–**Sends STCNs to STP networks.

	Command	Purpose
6.	**rep block port** {**id** *port-id* | *neighbor_offset* | **preferred**} **vlan** {*vlan-list* | **all**}	(Optional) Configures VLAN load balancing on the primary edge port, identify the REP alternate port in one of three ways, and configure the VLANs to be blocked on the alternate port. ■ **id** *port-id*–Identifies the alternate port by port ID. The port ID is automatically generated for each port in the segment. You can view interface port IDs by entering the **show interface** *interface-id* **rep** [**detail**] privileged EXEC command. ■ *neighbor_offset* number–Identifies the alternate port as a downstream neighbor from an edge port. The range is from -256 to 256, with negative numbers indicating the downstream neighbor from the secondary edge port. A value of **0** is invalid. Enters **-1** to identify the secondary edge port as the alternate port. See Figure 55 on page 386 for an example of neighbor offset numbering. **Note:** Because you enter this command at the primary edge port (offset number 1), you would never enter an offset value of 1 to identify an alternate port. ■ **preferred**–Selects the regular segment port previously identified as the preferred alternate port for VLAN load balancing. ■ **vlan** *vlan-list*–Blocks one VLAN or a range of VLANs. ■ **vlan all**–Blocks all VLANs. **Note:** Enter this command only on the REP primary edge port.
7.	**rep preempt delay** *seconds*	(Optional) You must enter this command and configure a preempt time delay if you want VLAN load balancing to automatically trigger after a link failure and recovery. The time delay range is 15 to 300 seconds. The default is manual preemption with no time delay. **Note:** Enter this command only on the REP primary edge port.
8.	**rep lsl-age-timer** *value*	(Optional) Configures a time (in milliseconds) for which the REP interface remains up without receiving a hello from a neighbor. The range is from 120 to 10000 ms in 40-ms increments. The default is 5000 ms (5 seconds). **Note:** If the neighbor device is not running Cisco IOS Release 12.2(52)SE or later, it only accepts values from 3000 to 10000 ms in 500-ms intervals. EtherChannel port channel interfaces do not support LSL age-timer values less than 1000 ms.
9.	**end**	Returns to privileged EXEC mode.

Configuring REP Negotiated

Use the following procedure to configure REP Negotiated in a REP network where a new switch is being inserted into the existing REP ring topology. The adjacent switches to this newly inserted switch are referred to as peer switches.

1. On the new switch, configure **rep bpduleak** in global configuration mode:

```
rep bpduleak
!
```

2. Configure the EEM macros on the new switch as shown in the following example. This example assumes that the peer switches are configured with REP Segment 777 and the newly inserted switch has uplink ports GigabitEthernet 1/1 and GigabitEthernet 1/2.

   ```
   macro auto execute CISCO_REP_NEG_EVENT {
    config terminal
    no rep bpduleak
    interface GigabitEthernet 1/1
    switchport mode trunk
    no rep negotiated
    rep segment 777
    interface GigabitEthernet 1/2
    switchport mode trunk
    no rep negotiated
    rep segment 777
    exit
   }

   macro auto execute CISCO_REP_NONNEG_EVENT {
   config terminal
   no rep bpduleak
   interface GigabitEthernet 1/1
   no rep negotiated
   interface GigabitEthernet 1/2
   no rep negotiated
   }
   ```

3. Insert the new switch into the existing REP Ring topology.

   Note: The newly inserted switch still does not have any REP Segment configurations.

4. Check the output of **show rep topology** on the peer switches.

   The output should show that **rep bpduleak** is in effect. The REP segment remains intact, but the newly inserted switch is not reflected in the topology. This indicates that the newly inserted switch is transparently forwarding the REP traffic between its uplink ports.

5. Configure **rep negotiated** on both the uplink interfaces of the newly inserted switch.

   Example:

   ```
   interface range GigabitEthernet 1/1-2
   rep negotiated
   !
   ```

6. Use the **show rep negotiated** command on the newly inserted switch to verify the status.

   Example:

   ```
   Switch2 #show rep negotiated
   REP negotiation status : Fail

   Interface1: GigabitEthernet1/1
   Status : enabled
   Rx State : fail, Segment-ID: 0

   Interface2: GigabitEthernet1/2
   Status : enabled
   Rx State : fail, Segment-ID: 0
   ```

7. Configure **rep negotiated** on the connected uplink interfaces of both the peer switches and wait for the REP Negotiation to complete. The following console log message on the newly inserted switch indicates that an EEM event has been triggered by REP Negotiation.

```
May 22 22:54:41.087:  REP negotiated event generated, executed CISCO_REP_NEG for Segment 777
```

8. Use the **show rep negotiated** command on the newly inserted switch to verify the status.

```
Switch2#show rep negotiated
REP negotiation status : Success

Interface1: GigabitEthernet1/1
Status : disabled

Interface2: GigabitEthernet1/2
Status : disabled
```

   REP is configured automatically in the newly inserted switch and it also appears in the **show rep topology** output on all the switches in the REP Ring.

REP Segment ID Validation

When inserting or replacing a REP node in a ring, you need to know the REP Segment-ID in advance. When operating multiple rings, this information can be difficult to obtain.

The REP negotiated feature exchanges REP information over a CDP TLV. The same CDP TLV is extended to learn the REP Segment-ID of the peer REP node. The REP node learns the Segment-ID and maintains it on interfaces until the link goes down. The ring must be initially configured with a static REP Segment-ID from the edge, but the rest of the REP ring can implement REP Segment-ID auto-discovery to facilitate deployment and automation.

To configure REP Segment ID Validation:

1. On the new switch, configure **rep bpduleak** in global configuration mode:

```
rep bpduleak
!
```

2. Configure a macro on the new switch, as shown in the following example. In this example, the uplink ports are GigabitEthernet 1/1/ and GigabitEthernet 1/2.

```
macro auto execute CISCO_REP_NEG_EVENT {
 config terminal
 no rep bpduleak
 interface GigabitEthernet 1/1
 switchport mode trunk
 no rep negotiated
 rep segment $LIMIT
 interface GigabitEthernet 1/2
 switchport mode trunk
 no rep negotiated
 rep segment $LIMIT
 exit
}
```

3. Insert the new switch into the existing REP Ring topology.

   Note: The newly inserted switch still does not have any REP Segment configurations.

4. Check the output of **show rep topology** on the peer switches.

The output should show that **rep bpduleak** is in effect. The REP segment remains intact, but the newly inserted switch is not reflected in the topology. This indicates that the newly inserted switch is transparently forwarding the REP traffic between its uplink ports.

5. Configure **rep negotiated** on both the uplink interfaces of the newly inserted switch.

   Example:

   ```
   interface range GigabitEthernet 1/1-2
   rep negotiated
   !
   ```

6. Use the **show rep negotiated** command on the newly inserted switch to verify the status.

   Example:

   ```
   Switch2 #show rep negotiated
   REP negotiation status : Fail

   Interface1: GigabitEthernet1/1
   Status : enabled
   Rx State : fail, Segment-ID: 0

   Interface2: GigabitEthernet1/2
   Status : enabled
   Rx State : fail, Segment-ID: 0
   ```

7. Configure **rep negotiated** on the connected uplink interfaces of both the peer switches and wait for the REP Negotiation to complete. The following console log message on the newly inserted switch indicates that an EEM event has been triggered by REP Negotiation.

   ```
   May 22 22:54:41.087:  REP negotiated event generated, executed CISCO_REP_NEG for Segment 777
   ```

8. Use the **show rep negotiated** command on the newly inserted switch to verify the status.

   ```
   Switch2#show rep negotiated
   REP negotiation status : Success

   Interface1: GigabitEthernet1/1
   Status : disabled

   Interface2: GigabitEthernet1/2
   Status : disabled
   ```

   REP is configured automatically in the newly inserted switch and it also appears in the **show rep topology** output on all the switches in the REP Ring.

Setting Manual Preemption for VLAN Load Balancing

Before You Begin

If you do not enter the **rep preempt delay** *seconds* interface configuration command on the primary edge port to configure a preemption time delay, the default is to manually trigger VLAN load balancing on the segment. Be sure that all other segment configuration has been completed before manually preempting VLAN load balancing. When you enter the **rep preempt segment** *segment-id* command, a confirmation message appears before the command is executed because preemption can cause network disruption.

	Command	Purpose
1.	**rep preempt segment** *segment-id*	Manually triggers VLAN load balancing on the segment. You will need to confirm the command before it is executed.
2.	**show rep topology**	Displays REP topology information.

Configuring SNMP Traps for REP

You can configure the switch to send REP-specific traps to notify the SNMP server of link operational status changes and port role changes.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**snmp mib rep trap-rate** *value*	Enables the switch to send REP traps, and sets the number of traps sent per second. The range is from 0 to 1000. The default is 0 (no limit imposed; a trap is sent at every occurrence).
3.	**end**	Returns to privileged EXEC mode.

Monitoring and Maintaining REP

Command	Purpose
show interface [*interface-id*] **rep** [**detail**]	Displays REP configuration and status for an interface or for all interfaces.
show rep topology [**segment** *segment_id*] [**archive**] [**detail**]	Displays REP topology information for a segment or for all segments, including the primary and secondary edge ports in the segment.
copy running-config startup config	Saves your entries in the switch startup configuration file.

Configuration Examples for Configuring REP

Configuring the Administrative VLAN: Example

This example shows how to configure the administrative VLAN as VLAN 100 and verify the configuration by entering the **show interface rep detail** command on one of the REP interfaces:

```
Switch# configure terminal
Switch (conf)# rep admin vlan 100
Switch (conf-if)# end
Switch# show interface GigabitEthernet1/17 rep detail
GigabitEthernet1/17 REP enabled
Segment-id: 2 (Edge)
PortID: 00010019E7144680
Preferred flag: No
Operational Link Status: TWO_WAY
Current Key: 0002001121A2D5800E4D
Port Role: Open
Blocked Vlan: <empty>
```

```
Admin-vlan: 100
Preempt Delay Timer: disabled
LSL Ageout Timer: 5000 ms
Configured Load-balancing Block Port: none
Configured Load-balancing Block VLAN: none
STCN Propagate to: none
LSL PDU rx: 3322, tx: 1722
HFL PDU rx: 32, tx: 5
BPA TLV rx: 16849, tx: 508
BPA (STCN, LSL) TLV rx: 0, tx: 0
BPA (STCN, HFL) TLV rx: 0, tx: 0
EPA-ELECTION TLV rx: 118, tx: 118
EPA-COMMAND TLV rx: 0, tx: 0
EPA-INFO TLV rx: 4214, tx: 4190
```

Configuring a Primary Edge Port: Examples

This example shows how to configure an interface as the primary edge port for segment 1, to send STCNs to segments 2 through 5, and to configure the alternate port as the port with port ID 0009001818D68700 to block all VLANs after a preemption delay of 60 seconds after a segment port failure and recovery. The interface is configured to remain up for 6000 milliseconds without receiving a hello from a neighbor.

```
Switch# configure terminal
Switch (conf)# interface GigabitEthernet1/17
Switch (conf-if)# rep segment 1 edge primary
Switch (conf-if)# rep stcn segment 2-5
Switch (conf-if)# rep block port 0009001818D68700 vlan all
Switch (conf-if)# rep preempt delay 60
Switch (conf-if)# rep lsl-age-timer 6000
Switch (conf-if)# end
```

This example shows how to configure an interface as the primary edge port when the interface has no external REP neighbor:

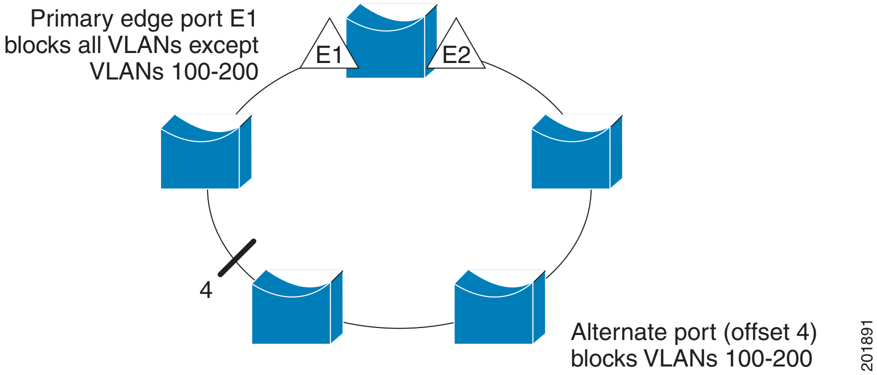
```
Switch# configure terminal
Switch (conf)# interface GigabitEthernet1/17
Switch (conf-if)# rep segment 1 edge no-neighbor primary
Switch (conf-if)# rep stcn segment 2-5
Switch (conf-if)# rep block port 0009001818D68700 vlan all
Switch (conf-if)# rep preempt delay 60
Switch (conf-if)# rep lsl-age-timer 6000
```

Configuring VLAN Blocking: Example

This example shows how to configure the VLAN blocking configuration shown in Figure 56 on page 398. The alternate port is the neighbor with neighbor offset number 4. After manual preemption, VLANs 100 to 200 are blocked at this port, and all other VLANs are blocked at the primary edge port E1 (Gigabit Ethernet port 1/0/1).

```
Switch# configure terminal
Switch (conf)# interface GigabitEthernet1/17
Switch (conf-if)# rep segment 1 edge primary
Switch (conf-if)# rep block port 4 vlan 100-200
Switch (conf-if)# end
```

Figure 56 Example of VLAN Blocking



Feature History

Table 46 Feature History

Feature	Release	Feature Information
REP Negotiated	15.2(8)E2	When the switch interfaces are configured with REP Negotiated (see REP Negotiated, page 385), REP status is negotiated with the peers. If the peer supports REP, it is migrated to REP. If the peer does not support REP, it is migrated to STP. The peer is migrated to REP or STP using an Embedded Event Manager (EEM) macro.

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Configuring FlexLinks and the MAC Address-Table Move Update

Restrictions for the FlexLinks and the MAC Address-Table Move Update

- To use this feature, the switch must be running the LAN Base image.

Information About Configuring the FlexLinks and the MAC Address-Table Move Update

FlexLinks

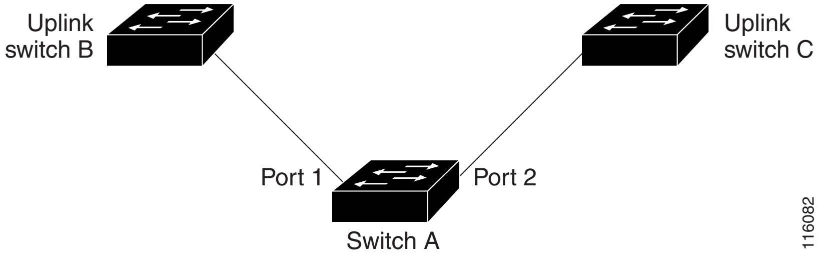
FlexLinks are a pair of a Layer 2 interfaces (switch ports or port channels) where one interface is configured to act as a backup to the other. The feature provides an alternative solution to the Spanning Tree Protocol (STP). Users can disable STP and still retain basic link redundancy. FlexLinks are typically configured in service provider or enterprise networks where customers do not want to run STP on the switch. If the switch is running STP, FlexLinks is not necessary because STP already provides link-level redundancy or backup.

You configure FlexLinks on one Layer 2 interface (the active link) by assigning another Layer 2 interface as the FlexLinks or backup link. When one of the links is up and forwarding traffic, the other link is in standby mode, ready to begin forwarding traffic if the other link shuts down. At any given time, only one of the interfaces is in the linkup state and forwarding traffic. If the primary link shuts down, the standby link starts forwarding traffic. When the active link comes back up, it goes into standby mode and does not forward traffic. STP is disabled on FlexLinks interfaces.

In Figure 57 on page 402, ports 1 and 2 on switch A are connected to uplink switches B and C. Because they are configured as FlexLinks, only one of the interfaces is forwarding traffic; the other is in standby mode. If port 1 is the active link, it begins forwarding traffic between port 1 and switch B; the link between port 2 (the backup link) and switch C is not forwarding traffic. If port 1 goes down, port 2 comes up and starts forwarding traffic to switch C. When port 1 comes back up, it goes into standby mode and does not forward traffic; port 2 continues forwarding traffic.

You can also choose to configure a preemption mechanism, specifying the preferred port for forwarding traffic. For example, in the example in Figure 57 on page 402, you can configure the FlexLinks pair with preemption mode. In the scenario shown, when port 1 comes back up and has more bandwidth than port 2, port 1 begins forwarding traffic after 60 seconds. Port 2 becomes the standby port. You do this by entering the interface configuration **switchport backup interface preemption mode bandwidth** and **switchport backup interface preemption delay** commands.

Figure 57 FlexLinks Configuration Example



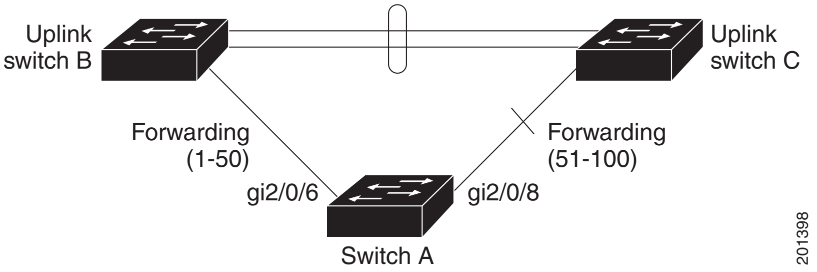
If a primary (forwarding) link goes down, a trap notifies the network management stations. If the standby link goes down, a trap notifies the users.

FlexLinks are supported only on Layer 2 ports and port channels, not on VLANs or on Layer 3 ports.

VLAN FlexLinks Load Balancing and Support

VLAN FlexLinks load-balancing allows you to configure a FlexLinks pair so that both ports simultaneously forward the traffic for some mutually exclusive VLANs. For example, if FlexLinks ports are configured for 1 to100 VLANs, the traffic of the first 50 VLANs can be forwarded on one port and the rest on the other port. If one of the ports fail, the other active port forwards all the traffic. When the failed port comes back up, it resumes forwarding traffic in the preferred VLANs. This way, apart from providing the redundancy, this FlexLinks pair can be used for load balancing. FlexLinks VLAN load balancing does not impose any restrictions on uplink switches.

Figure 58 VLAN FlexLinks Load Balancing Configuration Example



FlexLinks Multicast Fast Convergence

FlexLinks Multicast Fast Convergence reduces the multicast traffic convergence time after a FlexLinks failure.

Learning the Other FlexLinks Port as the mrouter Port

In a typical multicast network, there is a querier for each VLAN. A switch deployed at the edge of a network has one of its FlexLinks ports receiving queries. FlexLinks ports are also always forwarding at any given time.

A port that receives queries is added as an *mrouter* port on the switch. An mrouter port is part of all the multicast groups learned by the switch. After a changeover, queries are received by the other FlexLinks port. The other FlexLinks port is then learned as the mrouter port. After the changeover, multicast traffic flows through the other FlexLinks port. To achieve faster convergence of traffic, both FlexLinks ports are learned as mrouter ports whenever either FlexLinks port is learned as the mrouter port. Both FlexLinks ports are always part of multicast groups.

Though both FlexLinks ports are part of the groups in normal operation mode, all traffic on the backup port is blocked. So the normal multicast data flow is not affected by the addition of the backup port as an mrouter port. When the changeover happens, the backup port is unblocked, allowing the traffic to flow. In this case, the upstream multicast data flows as soon as the backup port is unblocked.

Generating IGMP Reports

When the backup link comes up after the changeover, the upstream new distribution switch does not start forwarding multicast data, because the port on the upstream router, which is connected to the blocked FlexLinks port, is not part of any multicast group. The reports for the multicast groups were not forwarded by the downstream switch because the backup link is blocked. The data does not flow on this port, until it learns the multicast groups, which occurs only after it receives reports.

The reports are sent by hosts when a general query is received, and a general query is sent within 60 seconds in normal scenarios. When the backup link starts forwarding, to achieve faster convergence of multicast data, the downstream switch immediately sends proxy reports for all the learned groups on this port without waiting for a general query.

Leaking IGMP Reports

To achieve multicast traffic convergence with minimal loss, a redundant data path must be set up before the FlexLinks active link goes down. This can be achieved by leaking only IGMP report packets on the FlexLinks backup link. These leaked IGMP report messages are processed by upstream distribution routers, so multicast data traffic gets forwarded to the backup interface. Because all incoming traffic on the backup interface is dropped at the ingress of the access switch, no duplicate multicast traffic is received by the host. When the FlexLinks active link fails, the access switch starts accepting traffic from the backup link immediately. The only disadvantage of this scheme is that it consumes bandwidth on the link between the distribution switches and on the backup link between the distribution and access switches. This feature is disabled by default and can be configured by using the **switchport backup interface** *interface-id* **multicast fast-convergence** command.

When this feature has been enabled at changeover, the switch does not generate the proxy reports on the backup port, which became the forwarding port.

MAC Address-Table Move Update

The MAC address-table move update feature allows the switch to provide rapid bidirectional convergence when a primary (forwarding) link goes down and the standby link begins forwarding traffic.

In Figure 59 on page 404, switch A is an access switch, and ports 1 and 2 on switch A are connected to uplink switches B and D through a FlexLinks pair. Port 1 is forwarding traffic, and port 2 is in the backup state. Traffic from the PC to the server is forwarded from port 1 to port 3. The MAC address of the PC has been learned on port 3 of switch C. Traffic from the server to the PC is forwarded from port 3 to port 1.

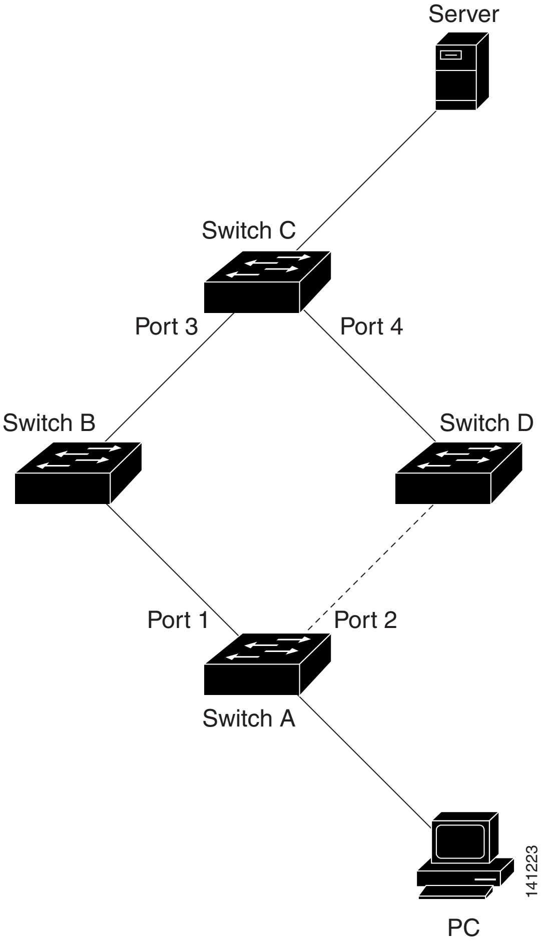
If the MAC address-table move update feature is not configured and port 1 goes down, port 2 starts forwarding traffic. However, for a short time, switch C keeps forwarding traffic from the server to the PC through port 3, and the PC does not get the traffic because port 1 is down. If switch C removes the MAC address of the PC on port 3 and relearns it on port 4, traffic can then be forwarded from the server to the PC through port 2.

If the MAC address-table move update feature is configured and enabled on the switches in Figure 59 on page 404 and port 1 goes down, port 2 starts forwarding traffic from the PC to the server. The switch sends a MAC address-table move update packet from port 2. Switch C gets this packet on port 4 and immediately learns the MAC address of the PC on port 4, which reduces the reconvergence time.

You can configure the access switch, switch A, to *send* MAC address-table move update messages. You can also configure the uplink switches B, C, and D to *get* and process the MAC address-table move update messages. When switch C gets a MAC address-table move update message from switch A, switch C learns the MAC address of the PC on port 4. Switch C updates the MAC address table, including the forwarding table entry for the PC.

Switch A does not need to wait for the MAC address-table update. The switch detects a failure on port 1 and immediately starts forwarding server traffic from port 2, the new forwarding port. This change occurs in 100 milliseconds (ms). The PC is directly connected to switch A, and the connection status does not change. Switch A does not need to update the PC entry in the MAC address table.

Figure 59 MAC Address-Table Move Update Example



Default Settings for FlexLinks and MAC Address-Table Move Update

Default Settings
FlexLinks is not configured, and there are no backup interfaces defined.
The preemption mode is off.
The preemption delay is 35 seconds.
MAC address-table move update is not configured on the switch.

Configuration Guidelines for FlexLinks and MAC Address-Table Move Update

Follow these guidelines to configure FlexLinks:

- You can configure up to 16 backup links.
- You can configure only one FlexLinks backup link for any active link, and it must be a different interface from the active interface.
- An interface can belong to only one FlexLinks pair. An interface can be a backup link for only one active link. An active link cannot belong to another FlexLinks pair.

- Neither of the links can be a port that belongs to an EtherChannel. However, you can configure two port channels (EtherChannel logical interfaces) as FlexLinks, and you can configure a port channel and a physical interface as FlexLinks, with either the port channel or the physical interface as the active link.
- A backup link does not have to be the same type (Gigabit Ethernet, or port channel) as the active link. However, you should configure both FlexLinks with similar characteristics so that there are no loops or changes in behavior if the standby link begins to forward traffic.
- STP is disabled on FlexLinks ports. A FlexLinks port does not participate in STP, even if the VLANs present on the port are configured for STP. When STP is not enabled, be sure that there are no loops in the configured topology. Once the FlexLinks configurations are removed, STP is reenabled on the ports.

Follow these guidelines to configure VLAN load balancing on the FlexLinks feature:

- For FlexLinks VLAN load balancing, you must choose the preferred VLANs on the backup interface.
- You cannot configure a preemption mechanism and VLAN load balancing for the same FlexLinks pair.

Follow these guidelines to configure the MAC address-table move update feature:

- You can enable and configure this feature on the access switch to *send* the MAC address-table move updates.
- You can enable and configure this feature on the uplink switches to *receive* the MAC address-table move updates.

How to Configure the FlexLinks and MAC Address-Table Move Update

Configuring FlexLinks

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the interface, and enters interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface). The port-channel range is 1 to 10.
3.	**switchport backup interface** *interface-id*	Configures a physical Layer 2 interface (or port channel) as part of a FlexLinks pair with the interface. When one link is forwarding traffic, the other interface is in standby mode.
4.	**end**	Returns to privileged EXEC mode.

Configuring a Preemption Scheme for FlexLinks

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the interface, and enter interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface). The port-channel range is 1 to 10.
3.	**switchport backup interface** *interface-id*	Configures a physical Layer 2 interface (or port channel) as part of a FlexLinks pair with the interface. When one link is forwarding traffic, the other interface is in standby mode.
4.	**switchport backup interface** *interface-id* **preemption mode** [**forced** | **bandwidth** | **off**]	Configures a preemption mechanism and delay for a FlexLinks interface pair. You can configure the preemption as: ■ **forced**—The active interface always preempts the backup. ■ **bandwidth**—The interface with the higher bandwidth always acts as the active interface. ■ **off**—No preemption happens from active to backup.
5.	**switchport backup interface** *interface-id* **preemption delay** *delay-time*	Configures the time delay until a port preempts another port. **Note:** Setting a delay time only works with forced and bandwidth modes.
6.	**end**	Returns to privileged EXEC mode.

Configuring VLAN Load Balancing on FlexLinks

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the interface, and enters interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface). The port-channel range is 1 to 10.
3.	**switchport backup interface** *interface-id* ***prefer vlan*** *vlan-range*	Configures a physical Layer 2 interface (or port channel) as part of a FlexLinks pair with the interface, and specifies the VLANs carried on the interface. The VLAN ID range is 1 to 4096.
4.	**end**	Returns to privileged EXEC mode.

Configuring the MAC Address-Table Move Update Feature

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the interface, and enters interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface). The port-channel range is 1 to 10.
3.	**switchport backup interface** *interface-id* or **switchport backup interface** *interface-id* **mmu primary vlan** *vlan-id*	Configures a physical Layer 2 interface (or port channel), as part of a FlexLinks pair with the interface. The MAC address-table move update VLAN is the lowest VLAN ID on the interface. Configures a physical Layer 2 interface (or port channel) and specifies the VLAN ID on the interface, which is used for sending the MAC address-table move update. When one link is forwarding traffic, the other interface is in standby mode.
4.	**end**	Returns to global configuration mode.
5.	**mac address-table move update transmit**	Enables the access switch to send MAC address-table move updates to other switches in the network if the primary link goes down and the switch starts forwarding traffic through the standby link.
6.	**end**	Returns to privileged EXEC mode.

Configuring the MAC Address-Table Move Update Messages

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**mac address-table move update receive**	Enables the switch to get and process the MAC address-table move updates.
3.	**end**	Returns to privileged EXEC mode.
4.	**show mac address-table move update**	Verifies the configuration.
5.	**copy running-config startup config**	(Optional) Saves your entries in the switch startup configuration file.

Maintaining and Monitoring the FlexLinks and MAC Address-Table Move Update

Command	Purpose
show interfaces [*interface-id*] **switchport backup**	Displays the FlexLinks backup interface configured for an interface or all the configured FlexLinks and the state of each active and backup interface (up or standby mode). When VLAN load balancing is enabled, the output displays the preferred VLANs on active and backup interfaces.
show mac address-table move update	Verifies the configuration.

Configuration Examples for the FlexLinks and MAC Address-Table Move Update

Configuring FlexLinks Port: Examples

These are configuration examples for learning the other FlexLinks port as the mrouter port when FlexLinks is configured, with output for the **show interfaces switchport backup** command:

This output shows a querier for VLANs 1 and 401, with their queries reaching the switch through the specified port:

```
Switch# show ip igmp snooping querier
Vlan    IP Address      IGMP Version       Port
-------------------------------------------------------------
1       1.1.1.1         v2                 Gi0/1
401     41.41.41.1      v2                 Gi0/1
```

Here is output for the **show ip igmp snooping mrouter** command for VLANs 1 and 401:

```
Switch# show ip igmp snooping mrouter
Vlan    ports
----    -----
1       Gi1/17(dynamic), Gi1/18(dynamic)
401     Gi1/17(dynamic), Gi1/18(dynamic)
```

Similarly, both FlexLinks ports are part of learned groups. In this example, GigabitEthernet1/17 is a receiver/host in VLAN 1, which is interested in two multicast groups:

```
Switch# show ip igmp snooping groups
Vlan    Group       Type    Version     Port List
-----------------------------------------------------------------------
1       228.1.5.1   igmp    v2          Gi1/17, Gi1/18, Fa2/1
1       228.1.5.2   igmp    v2          Gi1/17, Gi1/18, Fa2/1
```

When a host responds to the general query, the switch forwards this report on all the mrouter ports. In this example, when a host sends a report for the group 228.1.5.1, it is forwarded only on GigabitEthernet1/17, because the backup port GigabitEthernet1/18 is blocked. When the active link, GigabitEthernet1/17, goes down, the backup port, GigabitEthernet1/18, begins forwarding.

As soon as this port starts forwarding, the switch sends proxy reports for the groups 228.1.5.1 and 228.1.5.2 on behalf of the host. The upstream router learns the groups and starts forwarding multicast data. This is the default behavior of FlexLinks. This behavior changes when the user configures fast convergence using the **switchport backup interface GigabitEthernet1/18 multicast fast-convergence** command. This example shows how this feature is configured:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# switchport backup interface GigabitEthernet1/18 multicast fast-convergence
Switch(config-if)# exit
Switch# show interfaces switchport backup detail

Switch Backup Interface Pairs:
Active                Interface             Backup Interface State
------------------------------------------------------------------------
GigabitEthernet1/17  GigabitEthernet1/18  Active Up/Backup Standby
Preemption Mode : off
Multicast Fast Convergence : On
Mac Address Move Update Vlan : auto
```

This output shows a querier for VLAN 1 and 401 with their queries reaching the switch through the configured port:

```
Switch# show ip igmp snooping querier
Vlan       IP Address     IGMP Version   Port
-------------------------------------------------------------
1          1.1.1.1        v2             Gi1/17
401        41.41.41.1     v2             Gi1/17
```

This is output for the **show ip igmp snooping mrouter** command for VLAN 1 and 401:

```
Switch# show ip igmp snooping mrouter
Vlan       ports
----       -----
1          Gi1/17(dynamic), Gi1/18(dynamic)
401        Gi1/17(dynamic), Gi1/18(dynamic)
```

Similarly, both the FlexLinks ports are a part of the learned groups. In this example, the port is a receiver/host in VLAN 1, which is interested in two multicast groups:

```
Switch# show ip igmp snooping groups
Vlan    Group      Type    Version     Port List
-----------------------------------------------------------------------
1       228.1.5.1  igmp    v2          Gi1/17, Gi1/18, Gi1/17
1       228.1.5.2  igmp    v2          Gi1/17, Gi1/18, Gi1/17
```

Whenever a host responds to the general query, the switch forwards this report on all the mrouter ports. When you turn on this feature through the command-line port, and when a report is forwarded by the switch on the configured GigabitEthernet1/17, it is also leaked to the backup port GigabitEthernet1/18. The upstream router learns the groups and starts forwarding multicast data, which is dropped at the ingress because the GigabitEthernet1/18 is blocked. When the active link, GigabitEthernet1/17 goes down, the backup port, GigabitEthernet1/18, begins forwarding. You do not need to send any proxy reports because the multicast data is already being forwarded by the upstream router. By leaking reports to the backup port, a redundant multicast path has been set up, and the time taken for the multicast traffic convergence is minimal.

Configuring a Backup Interface: Example

This example shows how to configure an interface with a backup interface and to verify the configuration:

```
Switch# configure terminal
Switch(conf)# interface GigabitEthernet1/17
Switch(conf-if)# switchport backup interface GigabitEthernet1/18
Switch(conf-if)# end

Switch# show interfaces switchport backup
Switch Backup Interface Pairs:
```

```
Active Interface        Backup Interface        State
------------------------------------------------------------------------
Vlans Preferred on Active Interface: 1-3,5-4096
        Vlans Preferred on Backup Interface: 4
```

Configuring a Preemption Scheme: Example

This example shows how to configure the preemption mode as *forced* for a backup interface pair and to verify the configuration:

```
Switch# configure terminal
Switch(conf)# interface GigabitEthernet1/17
Switch(conf-if)#switchport backup interface GigabitEthernet1/18 preemption mode forced
Switch(conf-if)#switchport backup interface GigabitEthernet1/18 preemption delay 50
Switch(conf-if)# end

Switch# show interfaces switchport backup detail
Active Interface Backup Interface State
------------------------------------------------------------------------
GigabitEthernet1/17 GigabitEthernet1/18 Active Up/Backup Standby
Interface Pair : Gi1/17, Gi1/18
Preemption Mode : forced
Preemption Delay : 50 seconds
Bandwidth : 100000 Kbit (Gi1/17), 100000 Kbit (Gi1/18)
Mac Address Move Update Vlan : auto
```

Configuring VLAN Load Balancing on FlexLinks: Examples

In the following example, VLANs 1 to 50, 60, and 100 to 120 are configured on the switch:

```
Switch(config)# interface gigabitEthernet 1/2
Switch(config-if)# switchport backup interface gigabitEthernet 1/2 prefer vlan 60,100-120
```

When both interfaces are up, GigabitEthernet1/17 forwards traffic for VLANs 60 and 100 to 120, and GigabitEthernet1/18 forwards traffic for VLANs 1 to 50.

```
Switch# show interfaces switchport backup
Switch Backup Interface Pairs:

Active Interface        Backup Interface        State
------------------------------------------------------------------------
GigabitEthernet1/17    GigabitEthernet1/18    Active Up/Backup Standby
Vlans Preferred on Active Interface: 1-50
Vlans Preferred on Backup Interface: 60, 100-120
```

When a FlexLinks interface goes down (LINK_DOWN), VLANs preferred on this interface are moved to the peer interface of the FlexLinks pair. In this example, if interface Gigabit Ethernet1/1 goes down, Gigabit Ethernet1/2 carries all VLANs of the FlexLinks pair.

```
Switch# show interfaces switchport backup
Switch Backup Interface Pairs:

Active Interface        Backup Interface        State
------------------------------------------------------------------------
GigabitEthernet1/17    GigabitEthernet1/18    Active Down/Backup Up

Vlans Preferred on Active Interface: 1-50
Vlans Preferred on Backup Interface: 60, 100-120
```

When a FlexLinks interface comes up, VLANs preferred on this interface are blocked on the peer interface and moved to the forwarding state on the interface that has just come up. In this example, if interface Gigabit Ethernet1/1 comes up, VLANs preferred on this interface are blocked on the peer interface Gigabit Ethernet1/2 and forwarded on Gigabit Ethernet1/1.

```
Switch# show interfaces switchport backup
Switch Backup Interface Pairs:

Active Interface        Backup Interface        State
------------------------------------------------------------------------
GigabitEthernet1/17    GigabitEthernet1/18    Active Down/Backup Up

Vlans Preferred on Active Interface: 1-50
Vlans Preferred on Backup Interface: 60, 100-120

Switch# show interfaces switchport backup detail
Switch Backup Interface Pairs:

Active Interface        Backup Interface        State
------------------------------------------------------------------------
FastEthernet1/3       FastEthernet1/4       Active Down/Backup Up

Vlans Preferred on Active Interface: 1-2,5-4096
Vlans Preferred on Backup Interface: 3-4
Preemption Mode  : off
Bandwidth : 10000 Kbit (Fa1/3), 100000 Kbit (Fa1/4)
Mac Address Move Update Vlan : auto
```

Configuring MAC Address-Table Move Update: Example

This example shows how to configure an access switch to send MAC address-table move update messages:

```
Switch(conf)# interface GigabitEthernet1/17
Switch(conf-if)# switchport backup interface GigabitEthernet1/18 mmu primary vlan 2
Switch(conf-if)# exit
Switch(conf)# mac address-table move update transmit
Switch(conf)# end
```

This example shows how to verify the configuration:

```
Switch# show mac-address-table move update
Switch-ID : 010b.4630.1780
Dst mac-address : 0180.c200.0010
Vlans/Macs supported : 1023/8320
Default/Current settings: Rcv Off/On, Xmt Off/On
Max packets per min : Rcv 40, Xmt 60
Rcv packet count : 5
Rcv conforming packet count : 5
Rcv invalid packet count : 0
Rcv packet count this min : 0
Rcv threshold exceed count : 0
Rcv last sequence# this min : 0
Rcv last interface : Po2
Rcv last src-mac-address : 000b.462d.c502
Rcv last switch-ID : 0403.fd6a.8700
Xmt packet count : 0
Xmt packet count this min : 0
Xmt threshold exceed count : 0
Xmt pak buf unavail cnt : 0
Xmt last interface : None
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Configuring DHCP

Information About Configuring DHCP

This chapter describes how to configure Dynamic Host Configuration Protocol (DHCP) snooping and option-82 data insertion, and the DHCP server port-based address allocation features on the switch. It also describes how to configure the IP source guard feature.

DHCP Snooping

DHCP is widely used in LAN environments to dynamically assign host IP addresses from a centralized server, which significantly reduces the overhead of administration of IP addresses. DHCP also helps conserve the limited IP address space because IP addresses no longer need to be permanently assigned to hosts; only those hosts that are connected to the network consume IP addresses.

DHCP Server

The DHCP server assigns IP addresses from specified address pools on a switch or router to DHCP clients and manages them. If the DHCP server cannot give the DHCP client the requested configuration parameters from its database, it forwards the request to one or more secondary DHCP servers defined by the network administrator.

DHCP Relay Agent

A DHCP relay agent is a Layer 3 device that forwards DHCP packets between clients and servers. Relay agents forward requests and replies between clients and servers when they are not on the same physical subnet. Relay agent forwarding is different from the normal Layer 2 forwarding, in which IP datagrams are switched transparently between networks. Relay agents receive DHCP messages and generate new DHCP messages to send on output interfaces.

DHCP Snooping

DHCP snooping is a DHCP security feature that provides network security by filtering untrusted DHCP messages and by building and maintaining a DHCP snooping binding database, also referred to as a DHCP snooping binding table.

DHCP snooping acts like a firewall between untrusted hosts and DHCP servers. You use DHCP snooping to differentiate between untrusted interfaces connected to the end user and trusted interfaces connected to the DHCP server or another switch.

Note: For DHCP snooping to function properly, all DHCP servers must be connected to the switch through trusted interfaces.

An untrusted DHCP message is a message that is received from outside the network or firewall. When you use DHCP snooping in a service-provider environment, an untrusted message is sent from a device that is not in the service-provider network, such as a customer’s switch. Messages from unknown devices are untrusted because they can be sources of traffic attacks.

The DHCP snooping binding database has the MAC address, the IP address, the lease time, the binding type, the VLAN number, and the interface information that corresponds to the local untrusted interfaces of a switch. It does not have information regarding hosts interconnected with a trusted interface.

In a service-provider network, a trusted interface is connected to a port on a device in the same network. An untrusted interface is connected to an untrusted interface in the network or to an interface on a device that is not in the network.

When a switch receives a packet on an untrusted interface and the interface belongs to a VLAN in which DHCP snooping is enabled, the switch compares the source MAC address and the DHCP client hardware address. If the addresses match (the default), the switch forwards the packet. If the addresses do not match, the switch drops the packet.

The switch drops a DHCP packet when one of these situations occurs:

- A packet from a DHCP server, such as a DHCPOFFER, DHCPACK, DHCPNAK, or DHCPLEASEQUERY packet, is received from outside the network or firewall.
- A packet is received on an untrusted interface, and the source MAC address and the DHCP client hardware address do not match.
- The switch receives a DHCPRELEASE or DHCPDECLINE broadcast message that has a MAC address in the DHCP snooping binding database, but the interface information in the binding database does not match the interface on which the message was received.
- A DHCP relay agent forwards a DHCP packet that includes a relay-agent IP address that is not 0.0.0.0, or the relay agent forwards a packet that includes option-82 information to an untrusted port.

If the switch is an aggregation switch supporting DHCP snooping and is connected to an edge switch that is inserting DHCP option-82 information, the switch drops packets with option-82 information when packets are received on an untrusted interface. If DHCP snooping is enabled and packets are received on a trusted port, the aggregation switch does not learn the DHCP snooping bindings for connected devices and cannot build a complete DHCP snooping binding database.

When an aggregation switch can be connected to an edge switch through an untrusted interface and you enter the **ip dhcp snooping information option allow-untrusted** global configuration command, the aggregation switch accepts packets with option-82 information from the edge switch. The aggregation switch learns the bindings for hosts connected through an untrusted switch interface. The DHCP security features, such as dynamic ARP inspection or IP source guard, can still be enabled on the aggregation switch while the switch receives packets with option-82 information on untrusted input interfaces to which hosts are connected. The port on the edge switch that connects to the aggregation switch must be configured as a trusted interface.

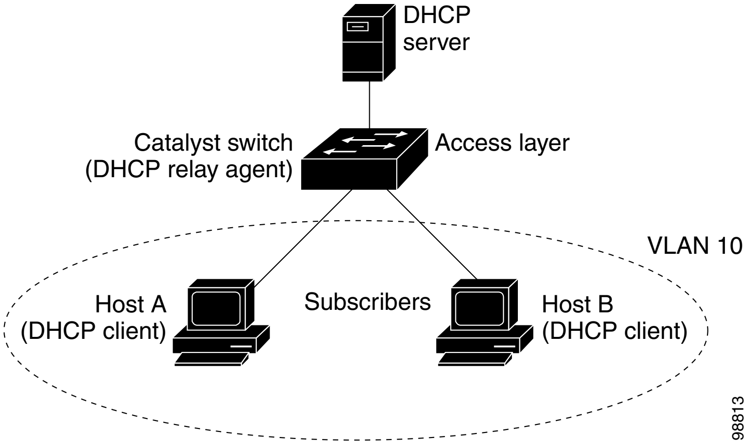
Option-82 Data Insertion

In residential, metropolitan Ethernet-access environments, DHCP can centrally manage the IP address assignments for a large number of subscribers. When the DHCP option-82 feature is enabled on the switch, a subscriber device is identified by the switch port through which it connects to the network (in addition to its MAC address). Multiple hosts on the subscriber LAN can be connected to the same port on the access switch and are uniquely identified.

Note: The DHCP option-82 feature is supported only when DHCP snooping is globally enabled and on the VLANs to which subscriber devices using this feature are assigned.

Figure 60 on page 415 is an example of a metropolitan Ethernet network in which a centralized DHCP server assigns IP addresses to subscribers connected to the switch at the access layer. Because the DHCP clients and their associated DHCP server do not reside on the same IP network or subnet, a DHCP relay agent (the Catalyst switch) is configured with a helper address to enable broadcast forwarding and to transfer DHCP messages between the clients and the server.

Figure 60 DHCP Relay Agent in a Metropolitan Ethernet Network



When you enable the DHCP snooping information option-82 on the switch, this sequence of events occurs:

- The host (DHCP client) generates a DHCP request and broadcasts it on the network.
- When the switch receives the DHCP request, it adds the option-82 information in the packet. By default, the remote-ID suboption is the switch MAC address, and the circuit-ID suboption is the port identifier, **vlan-mod-port**, from which the packet is received.
- If the IP address of the relay agent is configured, the switch adds this IP address in the DHCP packet.
- The switch forwards the DHCP request that includes the option-82 field to the DHCP server.
- The DHCP server receives the packet. If the server is option-82-capable, it can use the remote ID, the circuit ID, or both to assign IP addresses and implement policies, such as restricting the number of IP addresses that can be assigned to a single remote ID or circuit ID. The DHCP server then repeats the option-82 field in the DHCP reply.
- The DHCP server unicasts the reply to the switch if the request was relayed to the server by the switch. The switch verifies that it originally inserted the option-82 data by inspecting the remote ID and possibly the circuit ID fields. The switch removes the option-82 field and forwards the packet to the switch port that connects to the DHCP client that sent the DHCP request.

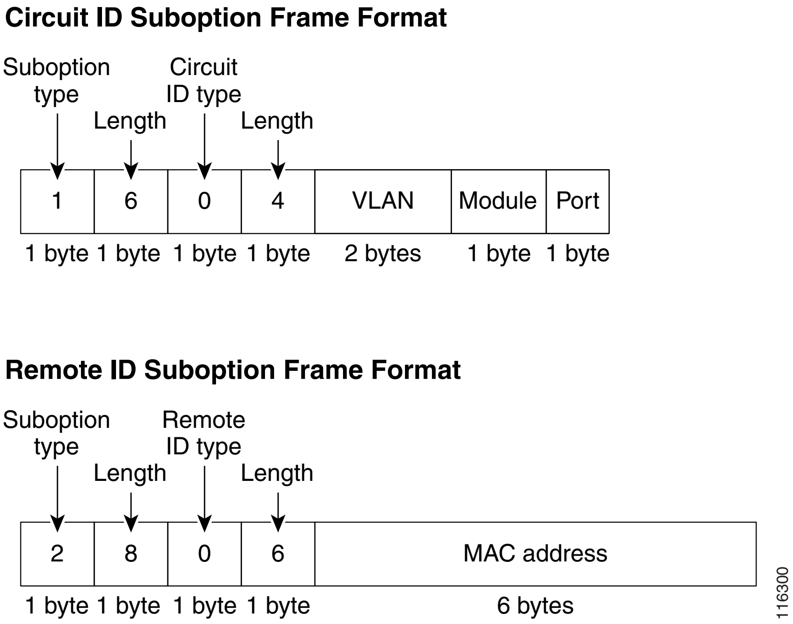
In the default suboption configuration, when the described sequence of events occurs, the values in these fields in Figure 61 on page 416 do not change:

- Circuit-ID suboption fields
  - Suboption type
  - Length of the suboption type
  - Circuit-ID type
  - Length of the circuit-ID type
- Remote-ID suboption fields
  - Suboption type
  - Length of the suboption type
  - Remote-ID type

  - Length of the remote-ID type

In the port field of the circuit-ID suboption, the port numbers start at 3. Figure 61 on page 416 shows the packet formats for the remote-ID suboption and the circuit-ID suboption when the default suboption configuration is used. The switch uses the packet formats when you globally enable DHCP snooping and enter the **ip dhcp snooping information option** global configuration command.

Figure 61 Suboption Packet Formats

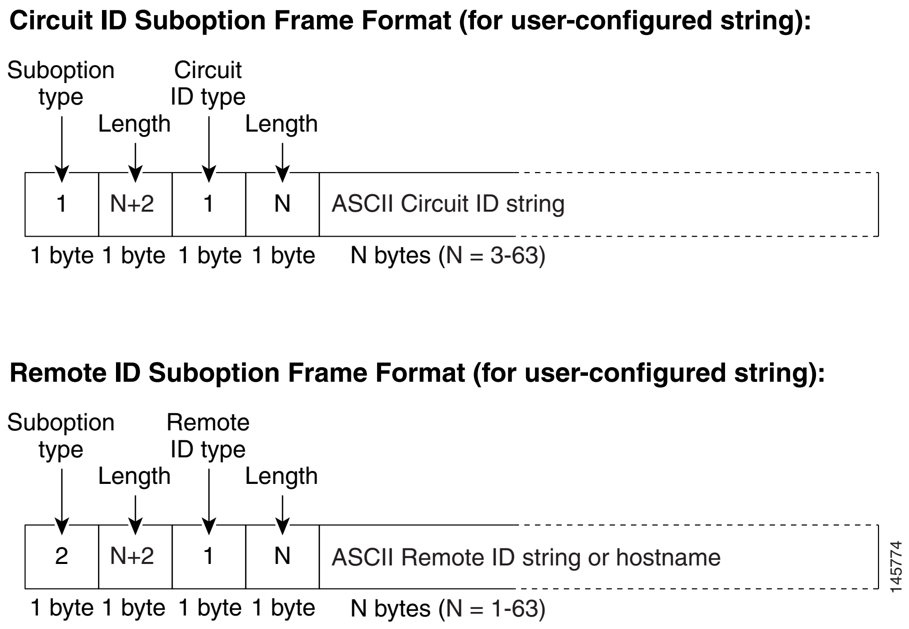


on page 417 shows the packet formats for user-configured remote-ID and circuit-ID suboptions The switch uses these packet formats when DHCP snooping is globally enabled and when the **ip dhcp snooping information option format remote-id** global configuration command **and the ip dhcp snooping vlan information option format-type circuit-id string** interface configuration command are entered.

The values for these fields in the packets change from the default values when you configure the remote-ID and circuit-ID suboptions:

- Circuit-ID suboption fields
  - The circuit-ID type is 1.
  - The length values are variable, depending on the length of the string that you configure.
- Remote-ID suboption fields
  - The remote-ID type is 1.
  - The length values are variable, depending on the length of the string that you configure.

Figure 62 User-Configured Suboption Packet Formats



Cisco IOS DHCP Server Database

During the DHCP-based autoconfiguration process, the designated DHCP server uses the Cisco IOS DHCP server database. It has IP addresses, *address bindings*, and configuration parameters, such as the boot file.

An address binding is a mapping between an IP address and a MAC address of a host in the Cisco IOS DHCP server database. You can manually assign the client IP address, or the DHCP server can allocate an IP address from a DHCP address pool.

DHCP Snooping Binding Database

When DHCP snooping is enabled, the switch uses the DHCP snooping binding database to store information about untrusted interfaces. The database can have up to 8192 bindings.

Each database entry (*binding*) has an IP address, an associated MAC address, the lease time (in hexadecimal format), the interface to which the binding applies, and the VLAN to which the interface belongs. The database agent stores the bindings in a file at a configured location. At the end of each entry is a checksum that accounts for all the bytes from the start of the file through all the bytes associated with the entry. Each entry is 72 bytes, followed by a space and then the checksum value.

To keep the bindings when the switch reloads, you must use the DHCP snooping database agent. If the agent is disabled, dynamic ARP inspection or IP source guard is enabled, and the DHCP snooping binding database has dynamic bindings, the switch loses its connectivity. If the agent is disabled and only DHCP snooping is enabled, the switch does not lose its connectivity, but DHCP snooping might not prevent DHCP spoofing attacks.

When reloading, the switch reads the binding file to build the DHCP snooping binding database. The switch updates the file when the database changes.

When a switch learns of new bindings or when it loses bindings, the switch immediately updates the entries in the database. The switch also updates the entries in the binding file. The frequency at which the file is updated is based on a configurable delay, and the updates are batched. If the file is not updated in a specified time (set by the write-delay and abort-timeout values), the update stops.

This is the format of the file with bindings:

```
<initial-checksum>
TYPE DHCP-SNOOPING
VERSION 1
BEGIN
<entry-1> <checksum-1>
<entry-2> <checksum-1-2>
...
...
<entry-n> <checksum-1-2-..-n>
END
```

Each entry in the file is tagged with a checksum value that the switch uses to verify the entries when it reads the file. The *initial-checksum* entry on the first line distinguishes entries associated with the latest file update from entries associated with a previous file update.

This is an example of a binding file:

```
2bb4c2a1
TYPE DHCP-SNOOPING
VERSION 1
BEGIN
192.1.168.1 3 0003.47d8.c91f 2BB6488E interface-id 21ae5fbb
192.1.168.3 3 0003.44d6.c52f 2BB648EB interface-id 1bdb223f
192.1.168.2 3 0003.47d9.c8f1 2BB648AB interface-id 584a38f0
END
```

When the switch starts and the calculated checksum value equals the stored checksum value, the switch reads entries from the binding file and adds the bindings to its DHCP snooping binding database. The switch ignores an entry when one of these situations occurs:

- The switch reads the entry and the calculated checksum value does not equal the stored checksum value. The entry and the ones following it are ignored.
- An entry has an expired lease time (the switch might not remove a binding entry when the lease time expires).
- The interface in the entry no longer exists on the system.
- The interface is a routed interface or a DHCP snooping-trusted interface.

Default DHCP Snooping Settings

Table 47 Default DHCP Snooping Settings

Feature	Default Setting
DHCP server	Enabled in Cisco IOS software, requires configuration[1]
DHCP relay agent	Enabled[2]
DHCP packet forwarding address	None configured
Checking the relay agent information	Enabled (invalid messages are dropped)2. on page 419
DHCP relay agent forwarding policy	Replace the existing relay agent information2. on page 419
DHCP snooping enabled globally	Disabled
DHCP snooping information option	Enabled
DHCP snooping option to accept packets on untrusted input interfaces[3]	Disabled
DHCP snooping limit rate	None configured

Table 47 Default DHCP Snooping Settings (continued)

Feature	Default Setting
DHCP snooping trust	Untrusted
DHCP snooping VLAN	Disabled
DHCP snooping MAC address verification	Enabled
Cisco IOS DHCP server binding database	Enabled in Cisco IOS software, requires configuration. **Note:** The switch gets network addresses and configuration parameters only from a device configured as a DHCP server.
DHCP snooping binding database agent	Enabled in Cisco IOS software, requires configuration. This feature is operational only when a destination is configured.

1. The switch responds to DHCP requests only if it is configured as a DHCP server.

2. The switch relays DHCP packets only if the IP address of the DHCP server is configured on the SVI of the DHCP client.

3. Use this feature when the switch is an aggregation switch that receives packets with option-82 information from an edge switch.

DHCP Snooping Configuration Guidelines

- You must globally enable DHCP snooping on the switch.
- DHCP snooping is not active until DHCP snooping is enabled on a VLAN.
- Before globally enabling DHCP snooping on the switch, make sure that the devices acting as the DHCP server and the DHCP relay agent are configured and enabled.
- Before configuring the DHCP snooping information option on your switch, be sure to configure the device that is acting as the DHCP server. For example, you must specify the IP addresses that the DHCP server can assign or exclude, or you must configure DHCP options for these devices.
- When configuring a large number of circuit IDs on a switch, consider the impact of lengthy character serstrings on the NVRAM or the flash memory. If the circuit-ID configurations, combined with other data, exceed the capacity of the NVRAM or the flash memory, an error message appears.
- Before configuring the DHCP relay agent on your switch, make sure to configure the device that is acting as the DHCP server. For example, you must specify the IP addresses that the DHCP server can assign or exclude, configure DHCP options for devices, or set up the DHCP database agent.
- If the DHCP relay agent is enabled but DHCP snooping is disabled, the DHCP option-82 data insertion feature is not supported.
- If a switch port is connected to a DHCP server, configure a port as trusted by entering the **ip dhcp snooping trust** interface configuration command.
- If a switch port is connected to a DHCP client, configure a port as untrusted by entering the **no ip dhcp snooping trust** interface configuration command.
- Do not enter the **ip dhcp snooping information option allow-untrusted** command on an aggregation switch to which an untrusted device is connected. If you enter this command, an untrusted device might spoof the option-82 information.
- You can display DHCP snooping statistics by entering the **show ip dhcp snooping statistics** user EXEC command, and you can clear the snooping statistics counters by entering the **clear ip dhcp snooping statistics** privileged EXEC command.

Note: Do not enable DHCP snooping on RSPAN VLANs. If DHCP snooping is enabled on RSPAN VLANs, DHCP packets might not reach the RSPAN destination port.

DHCP Snooping Binding Database Guidelines

- Because both NVRAM and the flash memory have limited storage capacity, we recommend that you store the binding file on a TFTP server.
- For network-based URLs (such as TFTP and FTP), you must create an empty file at the configured URL before the switch can write bindings to the binding file at that URL. See the documentation for your TFTP server to determine whether you must first create an empty file on the server; some TFTP servers cannot be configured this way.
- To ensure that the lease time in the database is accurate, we recommend that you enable and configure NTP. For more information, see Configuring Time and Date Manually, page 111.
- If NTP is configured, the switch writes binding changes to the binding file only when the switch system clock is synchronized with NTP.

Packet Forwarding Address

If the DHCP server and the DHCP clients are on different networks or subnets, you must configure the switch with the **ip helper-address** *address* interface configuration command. The general rule is to configure the command on the Layer 3 interface closest to the client. The address used in the **ip helper-address** command can be a specific DHCP server IP address, or it can be the network address if other DHCP servers are on the destination network segment. Using the network address enables any DHCP server to respond to requests.

DHCP Server Port-Based Address Allocation

DHCP server port-based address allocation is a feature that enables DHCP to maintain the same IP address on an Ethernet switch port regardless of the attached device client identifier or client hardware address.

When Ethernet switches are deployed in the network, they offer connectivity to the directly connected devices. In some environments, such as on a factory floor, if a device fails, the replacement device must be working immediately in the existing network. With the current DHCP implementation, there is no guarantee that DHCP would offer the same IP address to the replacement device. Control, monitoring, and other software expect a stable IP address associated with each device. If a device is replaced, the address assignment should remain stable even though the DHCP client has changed.

When configured, the DHCP server port-based address allocation feature ensures that the same IP address is always offered to the same connected port even as the client identifier or client hardware address changes in the DHCP messages received on that port. The DHCP protocol recognizes DHCP clients by the client identifier option in the DHCP packet. Clients that do not include the client identifier option are identified by the client hardware address. When you configure this feature, the port name of the interface overrides the client identifier or hardware address and the actual point of connection, the switch port, becomes the client identifier.

In all cases, by connecting the Ethernet cable to the same port, the same IP address is allocated through DHCP to the attached device.

The DHCP server port-based address allocation feature is only supported on a Cisco IOS DHCP server and not a third-party server.

By default, DHCP server port-based address allocation is disabled.

Port-Based Address Allocation Configuration Guidelines

These are the configuration guidelines for DHCP port-based address allocation:

- Only one IP address can be assigned per port.
- Reserved addresses (preassigned) cannot be cleared by using the **clear ip dhcp binding** global configuration command.
- Preassigned addresses are automatically excluded from normal dynamic IP address assignment. Preassigned addresses cannot be used in host pools, but there can be multiple preassigned addresses per DHCP address pool.
- To restrict assignments from the DHCP pool to preconfigured reservations (unreserved addresses are not offered to the client and other clients are not served by the pool), you can enter the **reserved-only** DHCP pool configuration command.

How to Configure DHCP

Configuring the DHCP Relay Agent

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**service dhcp**	Enables the DHCP server and relay agent on your switch. By default, this feature is enabled.
3.	**end**	Returns to privileged EXEC mode.

Specifying the Packet Forwarding Address

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface vlan** *vlan-id*	Creates a switch virtual interface by entering a VLAN ID, and enters interface configuration mode.
3.	**ip address** *ip-address subnet-mask*	Configures the interface with an IP address and an IP subnet.
4.	**ip helper-address** *address*	Specifies the DHCP packet forwarding address. The helper address can be a specific DHCP server address, or it can be the network address if other DHCP servers are on the destination network segment. Using the network address enables other servers to respond to DHCP requests. If you have multiple servers, you can configure one helper address for each server.
5.	**exit**	Returns to global configuration mode.

	Command	Purpose
6.	**interface range** *port-range* or **interface** *interface-id*	Configures multiple physical ports that are connected to the DHCP clients, and enters interface range configuration mode. or Configures a single physical port that is connected to the DHCP client, and enters interface configuration mode.
7.	**switchport mode access**	Defines the VLAN membership mode for the port.
8.	**switchport access vlan** *vlan-id*	Assigns the ports to the same VLAN as configured in Step 2.
9.	**end**	Returns to privileged EXEC mode.

Enabling DHCP Snooping and Option 82

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**ip dhcp snooping**	Enables DHCP snooping globally.
3.	**ip dhcp snooping vlan** *vlan-range*	Enables DHCP snooping on a VLAN or range of VLANs. The range is 1 to 4096. You can enter a single VLAN ID identified by VLAN ID number, a series of VLAN IDs separated by commas, a range of VLAN IDs separated by hyphens, or a range of VLAN IDs separated by entering the starting and ending VLAN IDs separated by a space.
4.	**ip dhcp snooping information option**	Enables the switch to insert and to remove DHCP relay information (option-82 field) in forwarded DHCP request messages to the DHCP server. This is the default setting.
5.	**ip dhcp snooping information option format remote-id [string** *ASCII-string* **|** *hostname*]	(Optional) Configures the remote-ID suboption. You can configure the remote ID as ■ String of up to 63 ASCII characters (no spaces) ■ Hostname for the switch **Note:** If the hostname is longer than 63 characters, it is truncated to 63 characters in the remote-ID configuration. The default remote ID is the switch MAC address.
6.	**ip dhcp snooping information option allow-untrusted**	(Optional) If the switch is an aggregation switch connected to an edge switch, enable the switch to accept incoming DHCP snooping packets with option-82 information from the edge switch. The default setting is disabled. **Note:** Enter this command only on aggregation switches that are connected to trusted devices.
7.	**interface** *interface-id*	Specifies the interface to be configured, and enters interface configuration mode.

	Command	Purpose
8.	**ip dhcp snooping vlan** *vlan* **information option format-type circuit-id [override] string** *ASCII-string*	(Optional) Configures the circuit-ID suboption for the specified interface. Specifies the VLAN and port identifier, using a VLAN ID in the range of 1 to 4096. The default circuit ID is the port identifier in the format **vlan-mod-port.** You can configure the circuit ID to be a string of 3 to 63 ASCII characters (no spaces). (Optional) Use the **override** keyword when you do not want the circuit-ID suboption inserted in TLV format to define subscriber information.
9.	**ip dhcp snooping trust**	(Optional) Configures the interface as trusted or as untrusted. Use the **no** keyword to configure an interface to receive messages from an untrusted client. The default setting is untrusted.
10.	**ip dhcp snooping limit rate** *rate*	(Optional) Configures the number of DHCP packets per second that an interface can receive. The range is 1 to 2048. By default, no rate limit is configured. **Note:** We recommend an untrusted rate limit of not more than 100 packets per second. If you configure rate limiting for trusted interfaces, you might need to increase the rate limit if the port is a trunk port assigned to more than one VLAN with DHCP snooping.
11.	**exit**	Returns to global configuration mode.
12.	**ip dhcp snooping verify mac-address**	(Optional) Configures the switch to verify that the source MAC address in a DHCP packet received on untrusted ports matches the client hardware address in the packet. The default is to verify that the source MAC address matches the client hardware address in the packet.
13.	**end**	Returns to privileged EXEC mode.

Enabling the DHCP Snooping Binding Database Agent

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**ip dhcp snooping database** {**flash:/***filename* | **ftp://***user***:***password***@***host***/***filename* | **http://**[[*username***:***password*]@]{*hostname* | *host-ip*}[/*directory*] **/***image-name***.tar** | **rcp://***user***@***host***/***filename*}| **tftp://***host***/***filename*	Specifies the URL for the database agent or the binding file by using one of these forms: ■ **flash:/***filename* ■ **ftp://***user***:***password***@***host***/***filename* ■ **http://**[[*username***:***password*]@]{*hostname* | *host-ip*}[/*directory*] **/***image-name***.tar** ■ **rcp://***user***@***host***/***filename* ■ **tftp://***host***/***filename*
3.	**ip dhcp snooping database timeout** *seconds*	Specifies (in seconds) how long to wait for the database transfer process to finish before stopping the process. The default is 300 seconds. The range is 0 to 86400. Use 0 to define an infinite duration, which means to continue trying the transfer indefinitely.

	Command	Purpose
4.	**ip dhcp snooping database write-delay** *seconds*	Specifies the duration for which the transfer should be delayed after the binding database changes. The range is from 15 to 86400 seconds. The default is 300 seconds (5 minutes).
5.	**end**	Returns to privileged EXEC mode.
6.	**ip dhcp snooping binding** *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id* **expiry** *seconds*	(Optional) Adds binding entries to the DHCP snooping binding database. The *vlan-id* range is from 1 to 4904. The *seconds* range is from 1 to 4294967295. Enter this command for each entry that you add. **Note:** Use this command when you are testing or debugging the switch.

Enabling DHCP Server Port-Based Address Allocation

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**ip dhcp use subscriber-id client-id**	Configures the DHCP server to globally use the subscriber identifier as the client identifier on all incoming DHCP messages.
3.	**ip dhcp subscriber-id interface-name**	Automatically generates a subscriber identifier based on the short name of the interface. A subscriber identifier configured on a specific interface takes precedence over this command.
4.	**interface** *interface-id*	Specifies the interface to be configured, and enters interface configuration mode.
5.	**ip dhcp server use subscriber-id client-id**	Configures the DHCP server to use the subscriber identifier as the client identifier on all incoming DHCP messages on the interface.
6.	**end**	Returns to privileged EXEC mode.

Preassigning an IP Address

After enabling DHCP port-based address allocation on the switch, use the **ip dhcp pool** global configuration command to preassign IP addresses and to associate them to clients. To restrict assignments from the DHCP pool to preconfigured reservations, you can enter the **reserved-only** DHCP pool configuration command. Unreserved addresses that are part of the network or on pool ranges are not offered to the client, and other clients are not served by the pool. By entering this command, users can configure a group of switches with DHCP pools that share a common IP subnet and that ignore requests from clients of other switches.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**ip dhcp pool** *poolname*	Enters DHCP pool configuration mode, and defines the name for the DHCP pool. The pool name can be a symbolic string (such as Engineering) or an integer (such as 0).
3.	**network** *network-number* [*mask* | */prefix-length*]	Specifies the subnet network number and mask of the DHCP address pool.

	Command	Purpose
4.	**address** *ip-address* **client-id** *string* [**ascii**]	Reserves an IP address for a DHCP client identified by the interface name. *string*—Can be an ASCII value or a hexadecimal value.
5.	**reserved-only**	(Optional) Uses only reserved addresses in the DHCP address pool. The default is to not restrict pool addresses.
6.	**end**	Returns to privileged EXEC mode.

Monitoring and Maintaining DHCP

Command	Purpose
show interface *interface id*	Displays the status and configuration of a specific interface.
show ip dhcp pool	Displays the DHCP address pools.
show ip dhcp binding	Displays address bindings on the Cisco IOS DHCP server.
ip dhcp snooping database timeout *seconds*	Specifies (in seconds) how long to wait for the database transfer process to finish before stopping.
ip dhcp snooping database write-delay *seconds*	Specifies (in seconds) the duration for which the transfer should be delayed after the binding database changes.
clear ip dhcp snooping database statistics	Clears the DHCP snooping binding database agent statistics.
renew ip dhcp snooping database	Renews the DHCP snooping binding database.
show ip dhcp snooping database [**detail**]	Displays the status and statistics of the DHCP snooping binding database agent.
show ip dhcp snooping	Displays the DHCP snooping configuration for a switch
show ip dhcp snooping binding	Displays only the dynamically configured bindings in the DHCP snooping binding database, also referred to as a binding table.
show ip dhcp snooping database	Displays the DHCP snooping binding database status and statistics.
show ip dhcp pool	Verifies DHCP pool configuration.
copy running-config startup-config	Saves your entries in the configuration file.

Configuration Examples for Configuring DHCP

Enabling DHCP Server Port-Based Address Allocation: Examples

In this example, a subscriber identifier is automatically generated, and the DHCP server ignores any client identifier fields in the DHCP messages and uses the subscriber identifier instead. The subscriber identifier is based on the short name of the interface and the client preassigned IP address 10.1.1.7.

```
switch# show running config
Building configuration...
Current configuration : 4899 bytes
!
version 12.2
!
```

```
hostname switch
!
no aaa new-model
clock timezone EST 0
ip subnet-zero
ip dhcp relay information policy removal pad
no ip dhcp use vrf connected
ip dhcp use subscriber-id client-id
ip dhcp subscriber-id interface-name
ip dhcp excluded-address 10.1.1.1 10.1.1.3
!
ip dhcp pool dhcppool
 network 10.1.1.0 255.255.255.0
 address 10.1.1.7 client-id "Et1/0" ascii
<output truncated>
```

This example shows that the preassigned address was correctly reserved in the DHCP pool:

```
switch# show ip dhcp pool dhcppool
Pool dhcp pool:
 Utilization mark (high/low) : 100 / 0
 Subnet size (first/next) : 0 / 0
 Total addresses : 254
 Leased addresses : 0
 Excluded addresses : 4
 Pending event : none
 1 subnet is currently in the pool:
 Current index   IP address range         Leased/Excluded/Total
 10.1.1.1        10.1.1.1 - 10.1.1.254     0     / 4 / 254
 1 reserved address is currently in the pool
 Address         Client
 10.1.1.7 Et1/0
```

Enabling DHCP Snooping: Example

This example shows how to enable DHCP snooping globally and on VLAN 10 and to configure a rate limit of 100 packets per second on a port:

```
Switch(config)# ip dhcp snooping
Switch(config)# ip dhcp snooping vlan 10
Switch(config)# ip dhcp snooping information option
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# ip dhcp snooping limit rate 100
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*
Cisco IOS DHCP Commands	*Cisco IOS IP Command Reference, Volume 1 of 3: Addressing and Services*
Cisco IOS DHCP Configuration Cisco IOS DHCP server port-based address allocation	"IP Addressing and Services" chapter of the *Cisco IOS IP Configuration Guide*
Cisco IOS DHCP Configuration Task List	"Configuring DHCP" chapter of the *Cisco IOS IP Configuration Guide*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	–

MIBs

MIBs	MIBs Link
–	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	–

Configuring Dynamic ARP Inspection

Prerequisites for Dynamic ARP Inspection

- Dynamic Address Resolution Protocol (ARP) inspection depends on the entries in the DHCP snooping binding database to verify IP-to-MAC address bindings in incoming ARP requests and ARP responses. Make sure to enable DHCP snooping to permit ARP packets that have dynamically assigned IP addresses.

Restrictions for Dynamic ARP Inspection

- To use this feature, the switch must be running the LAN Base image.

Information About Dynamic ARP Inspection

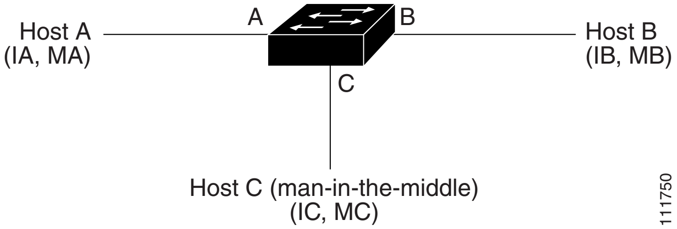
Dynamic ARP Inspection

Dynamic ARP inspection (DAI) helps prevent malicious attacks on the switch by not relaying invalid ARP requests and responses to other ports in the same VLAN.

ARP provides IP communication within a Layer 2 broadcast domain by mapping an IP address to a MAC address. For example, Host B wants to send information to Host A but does not have the MAC address of Host A in its ARP cache. Host B generates a broadcast message for all hosts within the broadcast domain to obtain the MAC address associated with the IP address of Host A. All hosts within the broadcast domain receive the ARP request, and Host A responds with its MAC address. However, because ARP allows a gratuitous reply from a host even if an ARP request was not received, an ARP spoofing attack and the poisoning of ARP caches can occur. After the attack, all traffic from the device under attack flows through the attacker's computer and then to the router, switch, or host.

A malicious user can attack hosts, switches, and routers connected to your Layer 2 network by poisoning the ARP caches of systems connected to the subnet and by intercepting traffic intended for other hosts on the subnet. Figure 63 on page 429 shows an example of ARP cache poisoning.

Figure 63 ARP Cache Poisoning



Hosts A, B, and C are connected to the switch on interfaces A, B and C, all of which are on the same subnet. Their IP and MAC addresses are shown in parentheses; for example, Host A uses IP address IA and MAC address MA. When Host A needs to communicate to Host B at the IP layer, it broadcasts an ARP request for the MAC address associated with IP address IB. When the switch and Host B receive the ARP request, they populate their ARP caches with an ARP binding for a host with the IP address IA and a MAC address MA; for example, IP address IA is bound to MAC address MA. When Host B responds, the switch and Host A populate their ARP caches with a binding for a host with the IP address IB and the MAC address MB.

Host C can poison the ARP caches of the switch, Host A, and Host B by broadcasting forged ARP responses with bindings for a host with an IP address of IA (or IB) and a MAC address of MC. Hosts with poisoned ARP caches use the MAC address MC as the destination MAC address for traffic intended for IA or IB. This means that Host C intercepts that traffic. Because Host C knows the true MAC addresses associated with IA and IB, it can forward the intercepted traffic to those hosts by using the correct MAC address as the destination. Host C has inserted itself into the traffic stream from Host A to Host B, the classic *man-in-the middle* attack.

DAI is a security feature that validates ARP packets in a network. It intercepts, logs, and discards ARP packets with invalid IP-to-MAC address bindings. This capability protects the network from certain man-in-the-middle attacks.

DAI ensures that only valid ARP requests and responses are relayed. The switch performs these activities:

- Intercepts all ARP requests and responses on untrusted ports
- Verifies that each of these intercepted packets has a valid IP-to-MAC address binding before updating the local ARP cache or before forwarding the packet to the appropriate destination
- Drops invalid ARP packets

DAI determines the validity of an ARP packet based on valid IP-to-MAC address bindings stored in a trusted database, the DHCP snooping binding database. This database is built by DHCP snooping if DHCP snooping is enabled on the VLANs and on the switch. If the ARP packet is received on a trusted interface, the switch forwards the packet without any checks. On untrusted interfaces, the switch forwards the packet only if it is valid.

Interface Trust States and Network Security

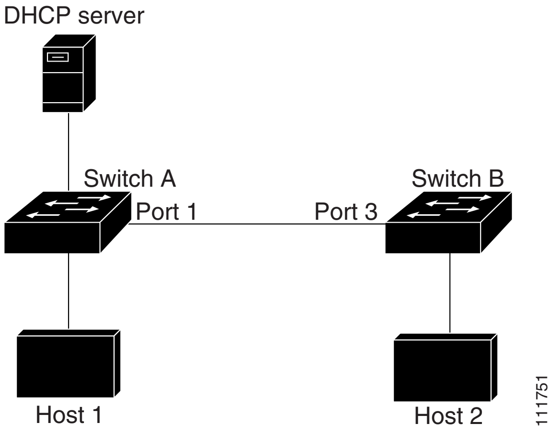
DAI associates a trust state with each interface on the switch. Packets arriving on trusted interfaces bypass all DAI validation checks, and those arriving on untrusted interfaces undergo the DAI validation process.

In a typical network configuration, you configure all switch ports connected to host ports as untrusted and configure all switch ports connected to switches as trusted. With this configuration, all ARP packets entering the network from a given switch bypass the security check. No other validation is needed at any other place in the VLAN or in the network. You configure the trust setting by using the **ip arp inspection trust** interface configuration command.

Caution: Use the trust state configuration carefully. Configuring interfaces as untrusted when they should be trusted can result in a loss of connectivity.

In Figure 64 on page 431, assume that both Switch A and Switch B are running DAI on the VLAN that includes Host 1 and Host 2. If Host 1 and Host 2 acquire their IP addresses from the DHCP server connected to Switch A, only Switch A binds the IP-to-MAC address of Host 1. Therefore, if the interface between Switch A and Switch B is untrusted, the ARP packets from Host 1 are dropped by Switch B. Connectivity between Host 1 and Host 2 is lost.

Figure 64 ARP Packet Validation on a VLAN Enabled for DAI



Configuring interfaces to be trusted when they are actually untrusted leaves a security hole in the network. If Switch A is not running DAI, Host 1 can easily poison the ARP cache of Switch B (and Host 2, if the link between the switches is configured as trusted). This condition can occur even though Switch B is running DAI.

DAI ensures that hosts (on untrusted interfaces) connected to a switch running DAI do not poison the ARP caches of other hosts in the network. However, DAI does not prevent hosts in other portions of the network from poisoning the caches of the hosts that are connected to a switch running DAI.

If some switches in a VLAN run DAI and other switches do not, configure the interfaces connecting these switches as untrusted. However, to validate the bindings of packets from non-DAI switches, configure the switch running DAI with ARP ACLs. When you cannot determine the bindings, at Layer 3 isolate switches running DAI from switches not running DAI switches.

Note: Depending on the setup of the DHCP server and the network, it might not be possible to validate a given ARP packet on all switches in the VLAN.

Rate Limiting of ARP Packets

The switch CPU performs DAI validation checks; therefore, the number of incoming ARP packets is rate-limited to prevent a denial-of-service attack. By default, the rate for untrusted interfaces is 15 packets per second (pps). Trusted interfaces are not rate-limited. You can change this setting by using the **ip arp inspection limit** interface configuration command.

When the rate of incoming ARP packets exceeds the configured limit, the switch places the port in the error-disabled state. The port remains in that state until you intervene. You can use the **errdisable recovery** global configuration command to enable error-disable recovery so that ports automatically emerge from this state after a specified timeout period.

Note: Unless you configure a rate limit on an interface, changing the trust state of the interface also changes its rate limit to the default value for that trust state. After you configure the rate limit, the interface retains the rate limit even when its trust state is changed. If you enter the **no ip arp inspection limit** interface configuration command, the interface reverts to its default rate limit.

Relative Priority of ARP ACLs and DHCP Snooping Entries

DAI uses the DHCP snooping binding database for the list of valid IP-to-MAC address bindings.

ARP ACLs take precedence over entries in the DHCP snooping binding database. The switch uses ACLs only if you configure them by using the **ip arp inspection filter vlan** global configuration command. The switch first compares ARP packets to user-configured ARP ACLs. If the ARP ACL denies the ARP packet, the switch also denies the packet even if a valid binding exists in the database populated by DHCP snooping.

Logging of Dropped Packets

When the switch drops a packet, it places an entry in the log buffer and then generates system messages on a rate-controlled basis. After the message is generated, the switch clears the entry from the log buffer. Each log entry contains flow information, such as the receiving VLAN, the port number, the source and destination IP addresses, and the source and destination MAC addresses.

You use the **ip arp inspection log-buffer** global configuration command to configure the number of entries in the buffer and the number of entries needed in the specified interval to generate system messages. You specify the type of packets that are logged by using the **ip arp inspection vlan logging** global configuration command.

A log-buffer entry can represent more than one packet. For example, if an interface receives many packets on the same VLAN with the same ARP parameters, the switch combines the packets as one entry in the log buffer and generates a single system message for the entry.

If the log buffer overflows, it means that a log event does not fit into the log buffer, and the display for the **show ip arp inspection log** privileged EXEC command is affected. Dashes in the display appears in place of all data except the packet count and the time. No other statistics are provided for the entry. If you see this entry in the display, increase the number of entries in the log buffer or increase the logging rate.

Default Dynamic ARP Inspection Settings

Feature	Default Setting
DAI	Disabled on all VLANs.
Interface trust state	All interfaces are untrusted.
Rate limit of incoming ARP packets	The rate is 15 pps on untrusted interfaces, assuming that the network is a switched network with a host connecting to as many as 15 new hosts per second. The rate is unlimited on all trusted interfaces. The burst interval is 1 second.
ARP ACLs for non-DHCP environments	No ARP ACLs are defined.
Validation checks	No checks are performed.
Log buffer	When DAI is enabled, all denied or dropped ARP packets are logged. The number of entries in the log is 32. The number of system messages is limited to 5 per second. The logging-rate interval is 1 second.
Per-VLAN logging	All denied or dropped ARP packets are logged.

Dynamic ARP Inspection Configuration Guidelines

- DAI is an ingress security feature; it does not perform any egress checking.

- DAI is not effective for hosts connected to switches that do not support DAI or that do not have this feature enabled. Because man-in-the-middle attacks are limited to a single Layer 2 broadcast domain, separate the domain with DAI checks from the one with no checking. This action secures the ARP caches of hosts in the domain enabled for DAI.
- DAI depends on the entries in the DHCP snooping binding database to verify IP-to-MAC address bindings in incoming ARP requests and ARP responses. Make sure to enable DHCP snooping to permit ARP packets that have dynamically assigned IP addresses. For configuration information, see Configuring Dynamic ARP Inspection, page 429

  When DHCP snooping is disabled or in non-DHCP environments, use ARP ACLs to permit or to deny packets.
- DAI is supported on access ports, trunk ports, EtherChannel ports, and private VLAN ports.

  Note: Do not enable DAI on RSPAN VLANs. If DAI is enabled on RSPAN VLANs, DAI packets might not reach the RSPAN destination port.
- A physical port can join an EtherChannel port channel only when the trust state of the physical port and the channel port match. Otherwise, the physical port remains suspended in the port channel. A port channel inherits its trust state from the first physical port that joins the channel. Consequently, the trust state of the first physical port need not match the trust state of the channel.

  Conversely, when you change the trust state on the port channel, the switch configures a new trust state on all the physical ports that comprise the channel.
- The operating rate for the port channel is cumulative across all the physical ports within the channel. For example, if you configure the port channel with an ARP rate-limit of 400 pps, all the interfaces combined on the channel receive an aggregate 400 pps. The rate of incoming ARP packets on EtherChannel ports is equal to the sum of the incoming rate of packets from all the channel members. Configure the rate limit for EtherChannel ports only after examining the rate of incoming ARP packets on the channel-port members.

  The rate of incoming packets on a physical port is checked against the port-channel configuration rather than the physical-ports configuration. The rate-limit configuration on a port channel is independent of the configuration on its physical ports.

  If the EtherChannel receives more ARP packets than the configured rate, the channel (including all physical ports) is placed in the error-disabled state.
- Make sure to limit the rate of ARP packets on incoming trunk ports. Configure trunk ports with higher rates to reflect their aggregation and to handle packets across multiple DAI-enabled VLANs. You also can use the **ip arp inspection limit none** interface configuration command to make the rate unlimited. A high rate-limit on one VLAN can cause a denial-of-service attack to other VLANs when the software places the port in the error-disabled state.
- When you enable DAI on the switch, policers that were configured to police ARP traffic are no longer effective. The result is that all ARP traffic is sent to the CPU.

How to Configure Dynamic ARP Inspection

Configuring Dynamic ARP Inspection in DHCP Environments

This procedure shows how to configure DAI when two switches support this feature. Host 1 is connected to Switch A, and Host 2 is connected to Switch B as shown in Figure 64 on page 431. Both switches are running DAI on VLAN 1 where the hosts are located. A DHCP server is connected to Switch A. Both hosts acquire their IP addresses from the same DHCP server. Therefore, Switch A has the bindings for Host 1 and Host 2, and Switch B has the binding for Host 2.

Before You Begin

You must perform this procedure on both switches. This procedure is required.

	Command	Purpose
1.	**show cdp neighbors**	Verifies the connection between the switches.
2.	**configure terminal**	Enters global configuration mode.
3.	**ip arp inspection vlan** *vlan-range*	Enables DAI on a per-VLAN basis. By default, DAI is disabled on all VLANs. *vlan-range*—Specifies a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4096. Specifies the same VLAN ID for both switches.
4.	**interface** *interface-id*	Specifies the interface connected to the other switch, and enters interface configuration mode.
5.	**ip arp inspection trust**	Configures the connection between the switches as trusted. By default, all interfaces are untrusted. The switch does not check ARP packets that it receives from the other switch on the trusted interface; it only forwards the packets. For untrusted interfaces, the switch intercepts all ARP requests and responses. It verifies that the intercepted packets have valid IP-to-MAC address bindings before updating the local cache and before forwarding the packet to the appropriate destination. The switch drops invalid packets and logs them in the log buffer according to the logging configuration specified with the **ip arp inspection vlan logging** global configuration command.
6.	**end**	Returns to privileged EXEC mode.

Configuring ARP ACLs for Non-DHCP Environments

This procedure shows how to configure DAI when Switch B shown in Figure 64 on page 431 does not support DAI or DHCP snooping.

If you configure port 1 on Switch A as trusted, a security hole is created because both Switch A and Host 1 could be attacked by either Switch B or Host 2. To prevent this possibility, you must configure port 1 on Switch A as untrusted. To permit ARP packets from Host 2, you must set up an ARP ACL and apply it to VLAN 1. If the IP address of Host 2 is not static (it is impossible to apply the ACL configuration on Switch A) you must separate Switch A from Switch B at Layer 3 and use a router to route packets between them.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**arp access-list** *acl-name*	Defines an ARP ACL, and enters ARP access-list configuration mode. By default, no ARP access lists are defined. **Note:** At the end of the ARP access list, there is an implicit **deny ip any mac any** command.
3.	**permit ip host** *sender-ip* **mac host** *sender-mac* [**log**]	Permits ARP packets from the specified host (Host 2). ■ *sender-ip*–Enters the IP address of Host 2. ■ *sender-mac*–Enters the MAC address of Host 2. ■ (Optional) **log**–Logs a packet in the log buffer when it matches the access control entry (ACE). Matches are logged if you also configure the **matchlog** keyword in the **ip arp inspection vlan logging** global configuration command. For more information, see Configuring the Log Buffer, page 438.
4.	**exit**	Returns to global configuration mode.
5.	**ip arp inspection filter** *arp-acl-name* **vlan** *vlan-range* [**static**]	Applies the ARP ACL to the VLAN. By default, no defined ARP ACLs are applied to any VLAN. ■ *arp-acl-name*–Specifies the name of the ACL created in Step 2. ■ *vlan-range*–Specifies the VLAN that the switches and hosts are in. You can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4096. ■ (Optional) **static**–Specifies to treat implicit denies in the ARP ACL as explicit denies and to drop packets that do not match any previous clauses in the ACL. DHCP bindings are not used. If you do not specify this keyword, it means that there is no explicit deny in the ACL that denies the packet, and DHCP bindings determine whether a packet is permitted or denied if the packet does not match any clauses in the ACL. ARP packets containing only IP-to-MAC address bindings are compared against the ACL. Packets are permitted only if the access list permits them.

	Command	Purpose
6.	**interface** *interface-id*	Specifies the Switch A interface that is connected to Switch B, and enters interface configuration mode.
7.	**no ip arp inspection trust**	Configures the Switch A interface that is connected to Switch B as untrusted. By default, all interfaces are untrusted. For untrusted interfaces, the switch intercepts all ARP requests and responses. It verifies that the intercepted packets have valid IP-to-MAC address bindings before updating the local cache and before forwarding the packet to the appropriate destination. The switch drops invalid packets and logs them in the log buffer according to the logging configuration specified with the **ip arp inspection vlan logging** global configuration command.
8.	**end**	Returns to privileged EXEC mode.

Limiting the Rate of Incoming ARP Packets

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the interface to be rate-limited, and enters interface configuration mode.
3.	**ip arp inspection limit** {**rate** *pps* [**burst interval** *seconds*] | **none**}	Limits the rate of incoming ARP requests and responses on the interface. The default rate is 15 pps on untrusted interfaces and unlimited on trusted interfaces. The burst interval is 1 second. ■ **rate** *pps*–Specifies an upper limit for the number of incoming packets processed per second. The range is 0 to 2048 pps. ■ (Optional) **burst interval** *seconds*–Specifies the consecutive interval in seconds, over which the interface is monitored for a high rate of ARP packets. The range is 1 to 15. ■ **rate none**–Specifies no upper limit for the rate of incoming ARP packets that can be processed.
4.	**exit**	Returns to global configuration mode.
5.	**errdisable recovery cause arp-inspection interval** *interval*	(Optional) Enables error recovery from the DAI error-disabled state. By default, recovery is disabled, and the recovery interval is 300 seconds. **interval** *interval*–Specifies the time in seconds to recover from the error-disabled state. The range is 30 to 86400.
6.	**exit**	Returns to privileged EXEC mode.

Performing Validation Checks

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**ip arp inspection validate** {[**src-mac**] [**dst-mac**] [**ip**]}	Performs a specific check on incoming ARP packets. By default, no checks are performed. ■ **src-mac**–Checks the source MAC address in the Ethernet header against the sender MAC address in the ARP body. This check is performed on both ARP requests and responses. When enabled, packets with different MAC addresses are classified as invalid and are dropped. ■ **dst-mac**–Checks the destination MAC address in the Ethernet header against the target MAC address in ARP body. This check is performed for ARP responses. When enabled, packets with different MAC addresses are classified as invalid and are dropped. ■ **ip**–Checks the ARP body for invalid and unexpected IP addresses. Addresses include 0.0.0.0, 255.255.255.255, and all IP multicast addresses. Sender IP addresses are checked in all ARP requests and responses, and target IP addresses are checked only in ARP responses. You must specify at least one of the keywords. Each command overrides the configuration of the previous command; that is, if a command enables **src** and **dst mac** validations, and a second command enables IP validation only, the **src** and **dst mac** validations are disabled as a result of the second command.
3.	**exit**	Returns to privileged EXEC mode.

Configuring the Log Buffer

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**ip arp inspection log-buffer** {**entries** *number* | **logs** *number* **interval** *seconds*}	Configures the DAI logging buffer. By default, when DAI is enabled, denied, or dropped, ARP packets are logged. The number of log entries is 32. The number of system messages is limited to 5 per second. The logging-rate interval is 1 second. ■ **entries** *number*–Specifies the number of entries to be logged in the buffer. The range is 0 to 1024. ■ **logs** *number* **interval** *seconds*–Specifies the number of entries to generate system messages in the specified interval. **logs** *number*–Specifies the range 0 to 1024. A 0 value means that the entry is placed in the log buffer, but a system message is not generated. **interval** *seconds*–Specifies the range 0 to 86400 seconds (1 day). A 0 value means that a system message is immediately generated (and the log buffer is always empty). An interval setting of 0 overrides a log setting of 0. The **logs** and **interval** settings interact. If the **logs** *number* X is greater than **interval** *seconds* Y, X divided by Y (X/Y) system messages are sent every second. Otherwise, one system message is sent every Y divided by X (Y/X) seconds.
3.	**ip arp inspection vlan** *vlan-range* **logging** {**acl-match** {**matchlog** | **none**} | **dhcp-bindings** {**all** | **none** | **permit**}}	Controls the type of packets that are logged per VLAN. By default, all denied or all dropped packets are logged. The term *logged* means the entry is placed in the log buffer and a system message is generated. ■ *vlan-range*–Specifies a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4096. ■ **acl-match matchlog**–Specifies log packets based on the ACE logging configuration. If you specify the **matchlog** keyword in this command and the **log** keyword in the **permit** or **deny** ARP access-list configuration command, ARP packets permitted or denied by the ACL are logged. ■ **acl-match none**–Does not log packets that match ACLs. ■ **dhcp-bindings all**–Logs all packets that match DHCP bindings. ■ **dhcp-bindings none**–Does not log packets that match DHCP bindings. ■ **dhcp-bindings permit**–Logs DHCP-binding permitted packets.
4.	**exit**	Returns to privileged EXEC mode.

Monitoring and Maintaining Dynamic ARP Inspection

Command	Description
clear ip arp inspection log	Clears the DAI log buffer.
clear ip arp inspection statistics	Clears the DAI statistics.
show arp access-list [*acl-name*]	Displays detailed information about ARP ACLs.
show errdisable recovery	Displays the error-disabled recovery timer information.
show ip arp inspection interfaces [*interface-id*]	Displays the trust state and the rate limit of ARP packets for the specified interface or all interfaces.
show ip arp inspection log	Displays the configuration and contents of the DAI log buffer.
show ip arp inspection vlan *vlan-range*	Displays the configuration and the operating state of DAI for the specified VLAN. If no VLANs are specified or if a range is specified, displays information only for VLANs with DAI enabled (active).
show ip arp inspection statistics [**vlan** *vlan-range*]	Displays statistics for forwarded, dropped, MAC validation failure, IP validation failure, ACL permitted and denied, and DHCP permitted and denied packets for the specified VLAN. If no VLANs are specified or if a range is specified, displays information only for VLANs with DAI enabled (active).
show ip dhcp snooping binding	Verifies the DHCP bindings.

Configuration Examples for Dynamic ARP Inspection

Configuring Dynamic ARP Inspection in DHCP Environments: Example

This example shows how to configure DAI on Switch A in VLAN 1. You would perform a similar procedure on Switch B:

```
Switch(config)# ip arp inspection vlan 1
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# ip arp inspection trust
```

Configuring ARP ACLs for Non-DHCP Environments: Example

This example shows how to configure an ARP ACL called *host2* on Switch A, to permit ARP packets from Host 2 (IP address 1.1.1.1 and MAC address 0001.0001.0001), to apply the ACL to VLAN 1, and to configure port 1 on Switch A as untrusted:

```
Switch(config)# arp access-list host2
Switch(config-arp-acl)# permit ip host 1.1.1.1 mac host 1.1.1
Switch(config-arp-acl)# exit
Switch(config)# ip arp inspection filter host2 vlan 1
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# no ip arp inspection trust
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*
DHCP configuration	“Configuring DHCP on the IE 5000 Switch”

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	–

MIBs

MIBs	MIBs Link
–	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	–

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Configuring IP Source Guard

Prerequisites for IP Source Guard

- You must globally configure the **ip device tracking maximum** *limit-number* interface configuration command globally for IPSG for static hosts to work. If you only configure this command on a port without enabling IP device tracking globally or setting an IP device tracking maximum on that interface, IPSG with static hosts will reject all the IP traffic from that interface. This requirement also applies to IPSG with static hosts on a Layer 2 access port.

Restrictions for IP Source Guard

- To use this feature, the switch must be running the LAN Base image.
- IP source guard (IPSG) is supported only on Layer 2 ports, including access and trunk ports.
- Do not use IPSG for static hosts on uplink ports or trunk ports.

Information About IP Source Guard

IP Source Guard

IPSG is a security feature that restricts IP traffic on nonrouted, Layer 2 interfaces by filtering traffic based on the DHCP snooping binding database and on manually configured IP source bindings. You can use IPSG to prevent traffic attacks if a host tries to use the IP address of its neighbor.

You can enable IPSG when DHCP snooping is enabled on an untrusted interface. After IPSG is enabled on an interface, the switch blocks all IP traffic received on the interface except for DHCP packets allowed by DHCP snooping. A port access control list (ACL) is applied to the interface. The port ACL allows only IP traffic with a source IP address in the IP source binding table and denies all other traffic.

Note: The port ACL takes precedence over any router ACLs or VLAN maps that affect the same interface.

The IP source binding table bindings are learned by DHCP snooping or are manually configured (static IP source bindings). An entry in this table has an IP address with its associated MAC address and VLAN number. The switch uses the IP source binding table only when IPSG is enabled.

You can configure IPSG with source IP address filtering or with source IP and MAC address filtering.

Source IP Address Filtering

When IPSG is enabled with this option, IP traffic is filtered based on the source IP address. The switch forwards IP traffic when the source IP address matches an entry in the DHCP snooping binding database or a binding in the IP source binding table.

When a DHCP snooping binding or static IP source binding is added, changed, or deleted on an interface, the switch modifies the port ACL by using the IP source binding changes and re-applies the port ACL to the interface.

If you enable IPSG on an interface on which IP source bindings (dynamically learned by DHCP snooping or manually configured) are not configured, the switch creates and applies a port ACL that denies all IP traffic on the interface. If you disable IPSG, the switch removes the port ACL from the interface.

Source IP and MAC Address Filtering

IP traffic is filtered based on the source IP and MAC addresses. The switch forwards traffic only when the source IP and MAC addresses match an entry in the IP source binding table.

When address filtering is enabled, the switch filters IP and non-IP traffic. If the source MAC address of an IP or non-IP packet matches a valid IP source binding, the switch forwards the packet. The switch drops all other types of packets except DHCP packets.

The switch uses port security to filter source MAC addresses. The interface can shut down when a port-security violation occurs.

IP Source Guard for Static Hosts

IPSG for static hosts extends the IPSG capability to non-DHCP and static environments. The previous IPSG used the entries created by DHCP snooping to validate the hosts connected to a switch. Any traffic received from a host without a valid DHCP binding entry is dropped. This security feature restricts IP traffic on nonrouted Layer 2 interfaces. It filters traffic based on the DHCP snooping binding database and on manually configured IP source bindings. The previous version of IPSG required a DHCP environment for IPSG to work.

IPSG for static hosts allows IPSG to work without DHCP. IPSG for static hosts relies on IP device tracking-table entries to install port ACLs. The switch creates static entries based on ARP requests or other IP packets to maintain the list of valid hosts for a given port. You can also specify the number of hosts allowed to send traffic to a given port. This is equivalent to port security at Layer 3.

IPSG for static hosts also supports dynamic hosts. If a dynamic host receives a DHCP-assigned IP address that is available in the IP DHCP snooping table, the same entry is learned by the IP device tracking table. When you enter the **show ip device tracking all** EXEC command, the IP device tracking table displays the entries as ACTIVE.

> **Note:** Some IP hosts with multiple network interfaces can inject some invalid packets into a network interface. The invalid packets contain the IP or MAC address for another network interface of the host as the source address. The invalid packets can cause IPSG for static hosts to connect to the host, to learn the invalid IP or MAC address bindings, and to reject the valid bindings. Consult the vendor of the corresponding operating system and the network interface to prevent the host from injecting invalid packets.

IPSG for static hosts initially learns IP or MAC bindings dynamically through an ACL-based snooping mechanism. IP or MAC bindings are learned from static hosts by ARP and IP packets. They are stored in the device tracking database. When the number of IP addresses that have been dynamically learned or statically configured on a given port reaches a maximum, the hardware drops any packet with a new IP address. To resolve hosts that have moved or gone away for any reason, IPSG for static hosts leverages IP device tracking to age out dynamically learned IP address bindings. This feature can be used with DHCP snooping. Multiple bindings are established on a port that is connected to both DHCP and static hosts. For example, bindings are stored in both the device tracking database as well as in the DHCP snooping binding database.

IP Source Guard Configuration Guidelines

- By default, IP source guard is disabled.
- You can configure static IP bindings only on nonrouted ports. If you enter the **ip source binding** *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id* global configuration command on a routed interface, this error message appears:

```
Static IP source binding can only be configured on switch port.
```

- When IP source guard with source IP filtering is enabled on an interface, DHCP snooping must be enabled on the access VLAN for that interface.
- If you are enabling IP source guard on a trunk interface with multiple VLANs and DHCP snooping is enabled on all the VLANs, the source IP address filter is applied on all the VLANs.

  If IP source guard is enabled and you enable or disable DHCP snooping on a VLAN on the trunk interface, the switch might not properly filter traffic.
- If you enable IP source guard with source IP and MAC address filtering, DHCP snooping and port security must be enabled on the interface. You must also enter the **ip dhcp snooping information option** global configuration command and ensure that the DHCP server supports option 82. When IP source guard is enabled with MAC address filtering, the DHCP host MAC address is not learned until the host is granted a lease. When forwarding packets from the server to the host, DHCP snooping uses option-82 data to identify the host port.
- When configuring IP source guard on interfaces on which a private VLAN is configured, port security is not supported.
- IP source guard is not supported on EtherChannels.
- You can enable this feature when 802.1x port-based authentication is enabled.
- If the number of ternary content addressable memory (TCAM) entries exceeds the maximum, the CPU usage increases.

How to Configure IP Source Guard

Enabling IP Source Guard

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the interface to be configured, and enters interface configuration mode.
3.	**ip verify source** or **ip verify source port-security**	Enables IPSG with source IP address filtering. Enables IPSG with source IP and MAC address filtering. **Note:** When you enable both IPSG and port security by using the **ip verify source port-security** interface configuration command, there are two caveats: • The DHCP server must support option-82, or the client is not assigned an IP address. • The MAC address in the DHCP packet is not learned as a secure address. The MAC address of the DHCP client is learned as a secure address only when the switch receives non-DHCP data traffic.

	Command	Purpose
4.	**exit**	Returns to global configuration mode.
5.	**ip source binding** *mac-address* **vlan** *vlan-id ip-address* **inteface** *interface-id*	Adds a static IP source binding. Enter this command for each static binding.
6.	**end**	Returns to privileged EXEC mode.

Configuring IP Source Guard for Static Hosts on a Layer 2 Access Port

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**ip device tracking**	Opens the IP host table, and globally enables IP device tracking.
3.	**interface** *interface-id*	Enters interface configuration mode.
4.	**switchport mode access**	Configures a port as access.
5.	**switchport access vlan** *vlan-id*	Configures the VLAN for this port.
6.	**ip verify source tracking port-security**	Enables IPSG for static hosts with MAC address filtering. **Note:** When you enable both IPSG and port security by using the **ip verify source port-security** interface configuration command: • The DHCP server must support option-82, or the client is not assigned an IP address. • The MAC address in the DHCP packet is not learned as a secure address. The MAC address of the DHCP client is learned as a secure address only when the switch receives non-DHCP data traffic.
7.	**ip device tracking maximum** *number*	Specifies a maximum limit for the number of static IPs that the IP device tracking table allows on the port. The range is 1to 10. The maximum number is 10. **Note:** You must configure the **ip device tracking maximum** *limit-number* interface configuration command.
8.	**switchport port-security**	(Optional) Activates port security for this port.
9.	**switchport port-security maximum** *value*	(Optional) Specifies a maximum of MAC addresses for this port.

	Command	Purpose
10.	**end**	Returns to privileged EXEC mode.
11.	**show ip verify source interface** *interface-id*	Verifies the configuration and displays IPSG permit ACLs for static hosts.
12.	**show ip device track all [active | inactive] count**	Verifies the configuration by displaying the IP-to-MAC binding for a given host on the switch interface. ■ **all active**—Displays only the active IP or MAC binding entries ■ **all inactive**—Displays only the inactive IP or MAC binding entries ■ **all**—Displays the active and inactive IP or MAC binding entries

Monitoring and Maintaining IP Source Guard

Command	Purpose
show ip device tracking	Displays the active IP or MAC binding entries for all interfaces.
show ip source binding	Displays the IP source bindings on a switch.
show ip verify source	Displays the IP source guard configuration on the switch.
copy running-config startup-config	Saves your entries in the configuration file.

Configuration Examples for IP Source Guard

Enabling IPSG with Source IP and MAC Filtering: Example

This example shows how to enable IPSG with source IP and MAC filtering on VLANs 10 and 11:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# ip verify source port-security
Switch(config-if)# exit
Switch(config)# ip source binding 0100.0022.0010 vlan 10 10.0.0.2 interface GigabitEthernet1/17
Switch(config)# ip source binding 0100.0230.0002 vlan 11 10.0.0.4 interface GigabitEthernet1/17
Switch(config)# end
```

Disabling IPSG with Static Hosts: Example

This example shows how to stop IPSG with static hosts on an interface:

```
Switch(config-if)# no ip verify source
Switch(config-if)# no ip device tracking max
```

Enabling IPSG for Static Hosts: Examples

This example shows how to enable IPSG with static hosts on a port:

```
Switch(config)# ip device tracking
Switch(config)# ip device tracking max 10
Switch(config-if)# ip verify source tracking port-security
```

This example shows how to enable IPSG for static hosts with IP filters on a Layer 2 access port and to verify the valid IP bindings on the interface Gi0/3:

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# ip device tracking
Switch(config)# interface gigabitethernet 0/3
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 10
Switch(config-if)# ip device tracking maximum 5
Switch(config-if)# ip verify source tracking
Switch(config-if)# end

Switch# show ip verify source
Interface  Filter-type  Filter-mode  IP-address       Mac-address        Vlan
---------  -----------  -----------  ---------------  -----------------  ----
Gi0/3      ip trk       active       40.1.1.24                           10
Gi0/3      ip trk       active       40.1.1.20                           10
Gi0/3      ip trk       active       40.1.1.21                           10
```

This example shows how to enable IPSG for static hosts with IP-MAC filters on a Layer 2 access port, to verify the valid IP-MAC bindings on the interface Gi0/3, and to verify that the number of bindings on this interface has reached the maximum:

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# ip device tracking
Switch(config)# interface gigabitethernet 0/3
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 1
Switch(config-if)# ip device tracking maximum 5
Switch(config-if)# switchport port-security
Switch(config-if)# switchport port-security maximum 5
Switch(config-if)# ip verify source tracking port-security
Switch(config-if)# end

Switch# show ip verify source
Interface  Filter-type  Filter-mode  IP-address       Mac-address        Vlan
---------  -----------  -----------  ---------------  -----------------  ----
Gi0/3      ip-mac trk   active       40.1.1.24        00:00:00:00:03:04  1
Gi0/3      ip-mac trk   active       40.1.1.20        00:00:00:00:03:05  1
Gi0/3      ip-mac trk   active       40.1.1.21        00:00:00:00:03:06  1
Gi0/3      ip-mac trk   active       40.1.1.22        00:00:00:00:03:07  1
Gi0/3      ip-mac trk   active       40.1.1.23        00:00:00:00:03:08  1
```

Displaying IP or MAC Binding Entries: Examples

This example displays all IP or MAC binding entries for all interfaces. The CLI displays all active as well as inactive entries. When a host is learned on a interface, the new entry is marked as active. When the same host is disconnected from that interface and connected to a different interface, a new IP or MAC binding entry displays as active as soon as the host is detected. The old entry for this host on the previous interface is marked as INACTIVE.

```
Switch# show ip device tracking all
IP Device Tracking = Enabled
IP Device Tracking Probe Count = 3
IP Device Tracking Probe Interval = 30
---------------------------------------------------------------------
  IP Address     MAC Address   Vlan  Interface              STATE
---------------------------------------------------------------------
200.1.1.8       0001.0600.0000  8    GigabitEthernet1/17     INACTIVE
200.1.1.9       0001.0600.0000  8    GigabitEthernet1/17     INACTIVE
200.1.1.10      0001.0600.0000  8    GigabitEthernet1/17     INACTIVE
200.1.1.1       0001.0600.0000  9    GigabitEthernet1/18     ACTIVE
200.1.1.1       0001.0600.0000  8    GigabitEthernet1/17     INACTIVE
200.1.1.2       0001.0600.0000  9    GigabitEthernet1/18     ACTIVE
200.1.1.2       0001.0600.0000  8    GigabitEthernet1/17     INACTIVE
200.1.1.3       0001.0600.0000  9    GigabitEthernet1/18     ACTIVE
200.1.1.3       0001.0600.0000  8    GigabitEthernet1/17     INACTIVE
200.1.1.4       0001.0600.0000  9    GigabitEthernet1/18     ACTIVE
200.1.1.4       0001.0600.0000  8    GigabitEthernet1/17     INACTIVE
200.1.1.5       0001.0600.0000  9    GigabitEthernet1/18     ACTIVE
200.1.1.5       0001.0600.0000  8    GigabitEthernet1/17     INACTIVE
200.1.1.6       0001.0600.0000  8    GigabitEthernet1/17     INACTIVE
200.1.1.7       0001.0600.0000  8    GigabitEthernet1/17     INACTIVE
```

This example displays all active IP or MAC binding entries for all interfaces:

```
Switch# show ip device tracking all active
IP Device Tracking = Enabled
IP Device Tracking Probe Count = 3
IP Device Tracking Probe Interval = 30
---------------------------------------------------------------------
  IP Address     MAC Address   Vlan  Interface              STATE
---------------------------------------------------------------------
200.1.1.1       0001.0600.0000  9    GigabitEthernet1/17     ACTIVE
200.1.1.2       0001.0600.0000  9    GigabitEthernet1/17     ACTIVE
200.1.1.3       0001.0600.0000  9    GigabitEthernet1/17     ACTIVE
200.1.1.4       0001.0600.0000  9    GigabitEthernet1/17     ACTIVE
200.1.1.5       0001.0600.0000  9    GigabitEthernet1/17     ACTIVE
```

This example displays all inactive IP or MAC binding entries for all interfaces. The host was first learned on GigabitEthernet 0/1 and then moved to GigabitEthernet 0/2. The IP or MAC binding entries learned on GigabitEthernet 0/1 are marked as inactive.

```
Switch# show ip device tracking all inactive
IP Device Tracking = Enabled
IP Device Tracking Probe Count = 3
IP Device Tracking Probe Interval = 30
---------------------------------------------------------------------
  IP Address     MAC Address   Vlan  Interface              STATE
---------------------------------------------------------------------
200.1.1.8       0001.0600.0000  8    GigabitEthernet1/17     INACTIVE
200.1.1.9       0001.0600.0000  8    GigabitEthernet1/17     INACTIVE
200.1.1.10      0001.0600.0000  8    GigabitEthernet1/17     INACTIVE
200.1.1.1       0001.0600.0000  8    GigabitEthernet1/17     INACTIVE
200.1.1.2       0001.0600.0000  8    GigabitEthernet1/17     INACTIVE
200.1.1.3       0001.0600.0000  8    GigabitEthernet1/17     INACTIVE
200.1.1.4       0001.0600.0000  8    GigabitEthernet1/17     INACTIVE
200.1.1.5       0001.0600.0000  8    GigabitEthernet1/17     INACTIVE
200.1.1.6       0001.0600.0000  8    GigabitEthernet1/17     INACTIVE
200.1.1.7       0001.0600.0000  8    GigabitEthernet1/17     INACTIVE
```

This example displays the count of all IP device tracking host entries for all interfaces:

```
Switch# show ip device tracking all count
Total IP Device Tracking Host entries: 5
---------------------------------------------------------------------
  Interface           Maximum Limit           Number of Entries
```

```
---------------------------------------------------------------------
Gi0/3                   5
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	–

MIBs

MIBs	MIBs Link
–	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	–

Configuring IGMP Snooping and MVR

Restrictions for IGMP Snooping and MVR

- You can set the maximum number of IGMP groups that a Layer 2 interface can join by using the **ip igmp max-groups** interface configuration command. Use the **no** form of this command to set the maximum back to the default, which is no limit. This restriction can be applied to Layer 2 ports only–you cannot set a maximum number of IGMP groups on routed ports or SVIs. You also can use this command on a logical EtherChannel interface but cannot use it on ports that belong to an EtherChannel port group.

Information About IGMP Snooping and MVR

This chapter describes how to configure Internet Group Management Protocol (IGMP) snooping on the switch, including an application of local IGMP snooping, Multicast VLAN Registration (MVR). It also includes procedures for controlling multicast group membership by using IGMP filtering and procedures for configuring the IGMP throttling action.

Note: For IP Version 6 (IPv6) traffic, Multicast Listener Discovery (MLD) snooping performs the same function as IGMP snooping for IPv4 traffic.

Note: You can either manage IP multicast group addresses through features such as IGMP snooping and MVR, or you can use static IP addresses.

IGMP Snooping

Layer 2 switches can use IGMP snooping to constrain the flooding of multicast traffic by dynamically configuring Layer 2 interfaces so that multicast traffic is forwarded to only those interfaces associated with IP multicast devices. As the name implies, IGMP snooping requires the LAN switch to snoop on the IGMP transmissions between the host and the router and to keep track of multicast groups and member ports. When the switch receives an IGMP report from a host for a particular multicast group, the switch adds the host port number to the forwarding table entry; when it receives an IGMP Leave Group message from a host, it removes the host port from the table entry. It also periodically deletes entries if it does not receive IGMP membership reports from the multicast clients.

Note: For more information on IP multicast and IGMP, see RFC 1112 and RFC 2236.

The multicast router sends out periodic general queries to all VLANs. All hosts interested in this multicast traffic send join requests and are added to the forwarding table entry. The switch creates one entry per VLAN in the IGMP snooping IP multicast forwarding table for each group from which it receives an IGMP join request.

The switch supports IP multicast group-based bridging, rather than MAC-addressed based groups. With multicast MAC address-based groups, if an IP address being configured translates (aliases) to a previously configured MAC address or to any reserved multicast MAC addresses (in the range 224.0.0.xxx), the command fails. Because the switch uses IP multicast groups, there are no address aliasing issues.

The IP multicast groups learned through IGMP snooping are dynamic. However, you can statically configure multicast groups by using the **ip igmp snooping vlan** *vlan-id* **static** *ip_address* **interface** *interface-id* global configuration command. If you specify group membership for a multicast group address statically, your setting supersedes any automatic manipulation by IGMP snooping. Multicast group membership lists can consist of both user-defined and IGMP snooping-learned settings.

You can configure an IGMP snooping querier to support IGMP snooping in subnets without multicast interfaces because the multicast traffic does not need to be routed. For more information about the IGMP snooping querier, see Configuring the IGMP Snooping Querier, page 461.

If a port spanning-tree, a port group, or a VLAN ID change occurs, the IGMP snooping-learned multicast groups from this port on the VLAN are deleted.

When you enable IGMP Immediate Leave, the switch immediately removes a port when it detects an IGMP Version 2 leave message on that port. You should only use the Immediate-Leave feature when there is a single receiver present on every port in the VLAN.

IGMP Versions

The switch supports IGMP Version 1, IGMP Version 2, and IGMP Version 3. These versions are interoperable on the switch. For example, if IGMP snooping is enabled on an IGMPv2 switch and the switch receives an IGMPv3 report from a host, the switch can forward the IGMPv3 report to the multicast router.

Note: The switch supports IGMPv3 snooping based only on the destination multicast MAC address. It does not support snooping based on the source MAC address or on proxy reports.

An IGMPv3 switch supports Basic IGMPv3 Snooping Support (BISS), which includes support for the snooping features on IGMPv1 and IGMPv2 switches and for IGMPv3 membership report messages. BISS constrains the flooding of multicast traffic when your network includes IGMPv3 hosts. It constrains traffic to approximately the same set of ports as the IGMP snooping feature on IGMPv2 or IGMPv1 hosts.

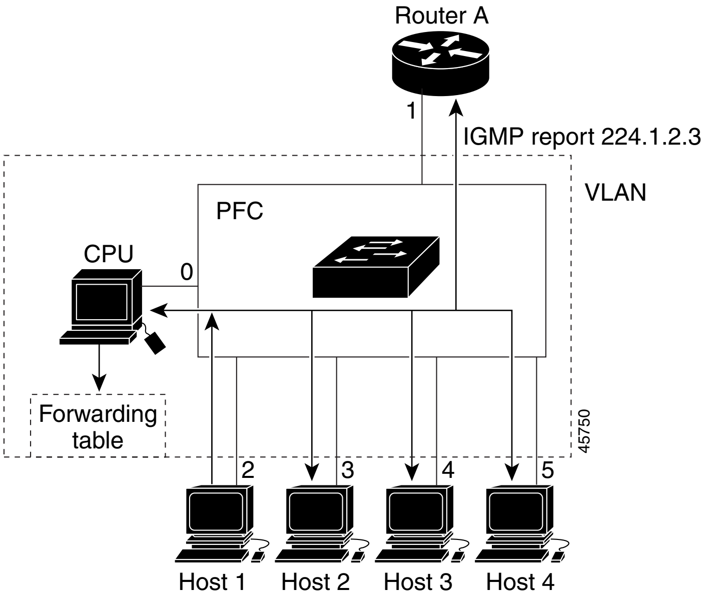
Note: IGMPv3 join and leave messages are not supported on switches running IGMP filtering or MVR.

An IGMPv3 switch can receive messages from and forward messages to a device running the Source Specific Multicast (SSM) feature.

Joining a Multicast Group

When a host connected to the switch wants to join an IP multicast group and it is an IGMP Version 2 client, it sends an unsolicited IGMP join message, specifying the IP multicast group to join. Alternatively, when the switch receives a general query from the router, it forwards the query to all ports in the VLAN. IGMP Version 1 or Version 2 hosts wanting to join the multicast group respond by sending a join message to the switch. The switch CPU creates a multicast forwarding-table entry for the group if it is not already present. The CPU also adds the interface where the join message was received to the forwarding-table entry. The host associated with that interface receives multicast traffic for that multicast group. See Figure 65 on page 451.

Figure 65 Initial IGMP Join Message



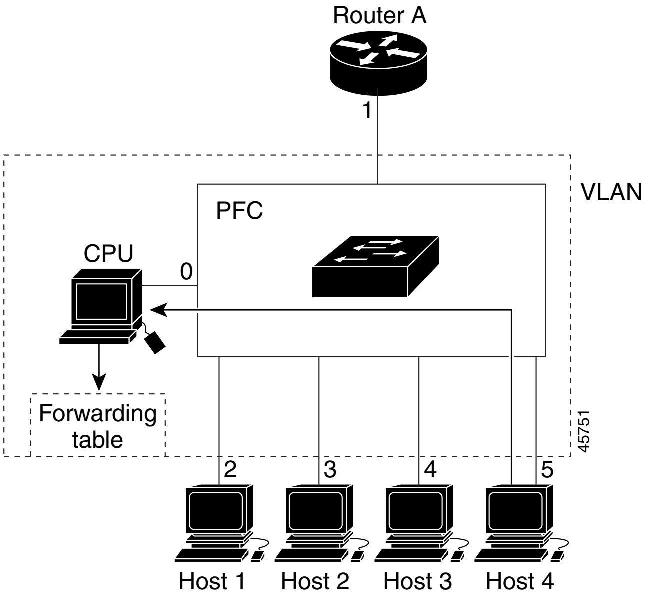
Router A sends a general query to the switch, which forwards the query to ports 2 through 5, which are all members of the same VLAN. Host 1 wants to join multicast group 224.1.2.3 and multicasts an IGMP membership report (IGMP join message) to the group. The switch CPU uses the information in the IGMP report to set up a forwarding-table entry, as shown in the table below, that includes the port numbers connected to Host 1 and the router.

Destination Address	Type of Packet	Ports
224.1.2.3	IGMP	1, 2

The switch hardware can distinguish IGMP information packets from other packets for the multicast group. The information in the table tells the switching engine to send frames addressed to the 224.1.2.3 multicast IP address that are not IGMP packets to the router and to the host that has joined the group.

If another host (for example, Host 4) sends an unsolicited IGMP join message for the same group (Figure 66 on page 452), the CPU receives that message and adds the port number of Host 4 to the forwarding table as shown below. Note that because the forwarding table directs IGMP messages only to the CPU, the message is not flooded to other ports on the switch. Any known multicast traffic is forwarded to the group and not to the CPU.

Figure 66 Second Host Joining a Multicast Group



Destination Address	Type of Packet	Ports
224.1.2.3	IGMP	1, 2, 5

Leaving a Multicast Group

The router sends periodic multicast general queries, and the switch forwards these queries through all ports in the VLAN. Interested hosts respond to the queries. If at least one host in the VLAN wishes to receive multicast traffic, the router continues forwarding the multicast traffic to the VLAN. The switch forwards multicast group traffic only to those hosts listed in the forwarding table for that IP multicast group maintained by IGMP snooping.

When hosts want to leave a multicast group, they can silently leave, or they can send a leave message. When the switch receives a leave message from a host, it sends a group-specific query to learn if any other devices connected to that interface are interested in traffic for the specific multicast group. The switch then updates the forwarding table for that MAC group so that only those hosts interested in receiving multicast traffic for the group are listed in the forwarding table. If the router receives no reports from a VLAN, it removes the group for the VLAN from its IGMP cache.

Immediate Leave

Immediate Leave is only supported on IGMP Version 2 hosts.

The switch uses IGMP snooping Immediate Leave to remove from the forwarding table an interface that sends a leave message without the switch sending group-specific queries to the interface. The VLAN interface is pruned from the multicast tree for the multicast group specified in the original leave message. Immediate Leave ensures optimal bandwidth management for all hosts on a switched network, even when multiple multicast groups are simultaneously in use.

Note: You should only use the Immediate Leave feature on VLANs where a single host is connected to each port. If Immediate Leave is enabled in VLANs where more than one host is connected to a port, some hosts might inadvertently be dropped.

When you enable IGMP Immediate Leave, the switch immediately removes a port when it detects an IGMP Version 2 leave message on that port. You should only use the Immediate-Leave feature when there is a single receiver present on every port in the VLAN.

IGMP Configurable-Leave Timer

You can configure the time that the switch waits after sending a group-specific query to determine if hosts are still interested in a specific multicast group. The IGMP leave response time can be configured from 100 to 5000 milliseconds. The default leave time is 1000 milliseconds. The timer can be set either globally or on a per-VLAN basis. The VLAN configuration of the leave time overrides the global configuration.

The actual leave latency in the network is usually the configured leave time. However, the leave time *might* vary around the configured time, depending on real-time CPU load conditions, network delays and the amount of traffic sent through the interface.

Note: The IGMP configurable leave time is only supported on hosts running IGMP Version 2.

IGMP Report Suppression

Note: IGMP report suppression is supported only when the multicast query has IGMPv1 and IGMPv2 reports. This feature is not supported when the query includes IGMPv3 reports.

The switch uses IGMP report suppression to forward only one IGMP report per multicast router query to multicast devices. When IGMP router suppression is enabled (the default), the switch sends the first IGMP report from all hosts for a group to all the multicast routers. The switch does not send the remaining IGMP reports for the group to the multicast routers. This feature prevents duplicate reports from being sent to the multicast devices.

If the multicast router query includes requests only for IGMPv1 and IGMPv2 reports, the switch forwards only the first IGMPv1 or IGMPv2 report from all hosts for a group to all the multicast routers.

If the multicast router query also includes requests for IGMPv3 reports, the switch forwards all IGMPv1, IGMPv2, and IGMPv3 reports for a group to the multicast devices.

If you disable IGMP report suppression, all IGMP reports are forwarded to the multicast routers. For configuration steps, see Disabling IGMP Report Suppression, page 462.

Default IGMP Snooping Configuration

Feature	Default Setting
IGMP snooping	Enabled globally and per VLAN
Multicast routers	None configured
Multicast router learning (snooping) method	PIM-DVMRP
IGMP snooping Immediate Leave	Disabled
Static groups	None configured
TCN[1] flood query count	2
TCN query solicitation	Disabled
IGMP snooping querier	Disabled
IGMP report suppression	Enabled

[1] TCN = Topology Change Notification

Snooping Methods

Multicast-capable router ports are added to the forwarding table for every Layer 2 multicast entry. The switch learns of such ports through one of these methods:

- Snooping on IGMP queries, Protocol Independent Multicast (PIM) packets, and Distance Vector Multicast Routing Protocol (DVMRP) packets
- Listening to Cisco Group Management Protocol (CGMP) packets from other routers
- Statically connecting to a multicast router port with the **ip igmp snooping mrouter** global configuration command

You can configure the switch either to snoop on IGMP queries and PIM/DVMRP packets or to listen to CGMP self-join or proxy-join packets. By default, the switch snoops on PIM/DVMRP packets on all VLANs. To learn of multicast router ports through only CGMP packets, use the **ip igmp snooping vlan** *vlan-id* **mrouter learn cgmp** global configuration command. When this command is entered, the router listens to only CGMP self-join and CGMP proxy-join packets and to no other CGMP packets. To learn of multicast router ports through only PIM-DVMRP packets, use the **ip igmp snooping vlan** *vlan-id* **mrouter learn pim-dvmrp** global configuration command.

Note: If you want to use CGMP as the learning method and no multicast routers in the VLAN are CGMP proxy-enabled, you must enter the **ip cgmp router-only** command to dynamically access the router.

Multicast Flooding Time After a TCN Event

You can control the time that multicast traffic is flooded after a topology change notification (TCN) event by using the **ip igmp snooping tcn flood query count** global configuration command. This command configures the number of general queries for which multicast data traffic is flooded after a TCN event. Some examples of TCN events are when the client changed its location and the receiver is on same port that was blocked but is now forwarding, and when a port went down without sending a leave message.

If you set the TCN flood query count to 1 by using the **ip igmp snooping tcn flood query count** command, the flooding stops after receiving 1 general query. If you set the count to 7, the flooding continues until 7 general queries are received. Groups are relearned based on the general queries received during the TCN event.

Flood Mode for TCN

When a topology change occurs, the spanning-tree root sends a special IGMP leave message (also known as global leave) with the group multicast address 0.0.0.0. However, when you enable the **ip igmp snooping tcn query solicit** global configuration command, the switch sends the global leave message whether or not it is the spanning-tree root. When the router receives this special leave, it immediately sends general queries, which expedite the process of recovering from the flood mode during the TCN event. Leaves are always sent if the switch is the spanning-tree root regardless of this configuration command. By default, query solicitation is disabled.

Multicast Flooding During a TCN Event

When the switch receives a TCN, multicast traffic is flooded to all the ports until 2 general queries are received. If the switch has many ports with attached hosts that are subscribed to different multicast groups, this flooding might exceed the capacity of the link and cause packet loss. You can use the **ip igmp snooping tcn flood** interface configuration command to control this behavior.

IGMP Snooping Querier Guidelines

- Configure the VLAN in global configuration mode.
- Configure an IP address on the VLAN interface. When enabled, the IGMP snooping querier uses the IP address as the query source address.
- If there is no IP address configured on the VLAN interface, the IGMP snooping querier tries to use the configured global IP address for the IGMP querier. If there is no global IP address specified, the IGMP querier tries to use the VLAN switch virtual interface (SVI) IP address (if one exists). If there is no SVI IP address, the switch uses the first available IP address configured on the switch. The first IP address available appears in the output of the **show ip interface** privileged EXEC command. The IGMP snooping querier does not generate an IGMP general query if it cannot find an available IP address on the switch.

- The IGMP snooping querier supports IGMP Versions 1 and 2.
- When administratively enabled, the IGMP snooping querier moves to the nonquerier state if it detects the presence of a multicast router in the network.
- When it is administratively enabled, the IGMP snooping querier moves to the operationally disabled state under these conditions:
  - IGMP snooping is disabled in the VLAN.
  - PIM is enabled on the SVI of the corresponding VLAN.

IGMP Report Suppression

IGMP report suppression is enabled by default. When it is enabled, the switch forwards only one IGMP report per multicast router query. When report suppression is disabled, all IGMP reports are forwarded to the multicast routers.

Multicast VLAN Registration

Note: To use this feature, the switch must be running the LAN Base image.

Multicast VLAN Registration (MVR) is designed for applications using wide-scale deployment of multicast traffic across an Ethernet ring-based service-provider network (for example, the broadcast of multiple television channels over a service-provider network). MVR allows a subscriber on a port to subscribe and unsubscribe to a multicast stream on the network-wide multicast VLAN. It allows the single multicast VLAN to be shared in the network while subscribers remain in separate VLANs. MVR provides the ability to continuously send multicast streams in the multicast VLAN, but to isolate the streams from the subscriber VLANs for bandwidth and security reasons.

MVR assumes that subscriber ports subscribe and unsubscribe (join and leave) these multicast streams by sending out IGMP join and leave messages. These messages can originate from an IGMP Version-2-compatible host with an Ethernet connection. Although MVR operates on the underlying mechanism of IGMP snooping, the two features operate independently of each other. One can be enabled or disabled without affecting the behavior of the other feature. However, if IGMP snooping and MVR are both enabled, MVR reacts only to join and leave messages from multicast groups configured under MVR. Join and leave messages from all other multicast groups are managed by IGMP snooping.

The switch CPU identifies the MVR IP multicast streams and their associated IP multicast group in the switch forwarding table, intercepts the IGMP messages, and modifies the forwarding table to include or remove the subscriber as a receiver of the multicast stream, even though the receivers might be in a different VLAN from the source. This forwarding behavior selectively allows traffic to cross between different VLANs.

You can set the switch for compatible or dynamic mode of MVR operation:

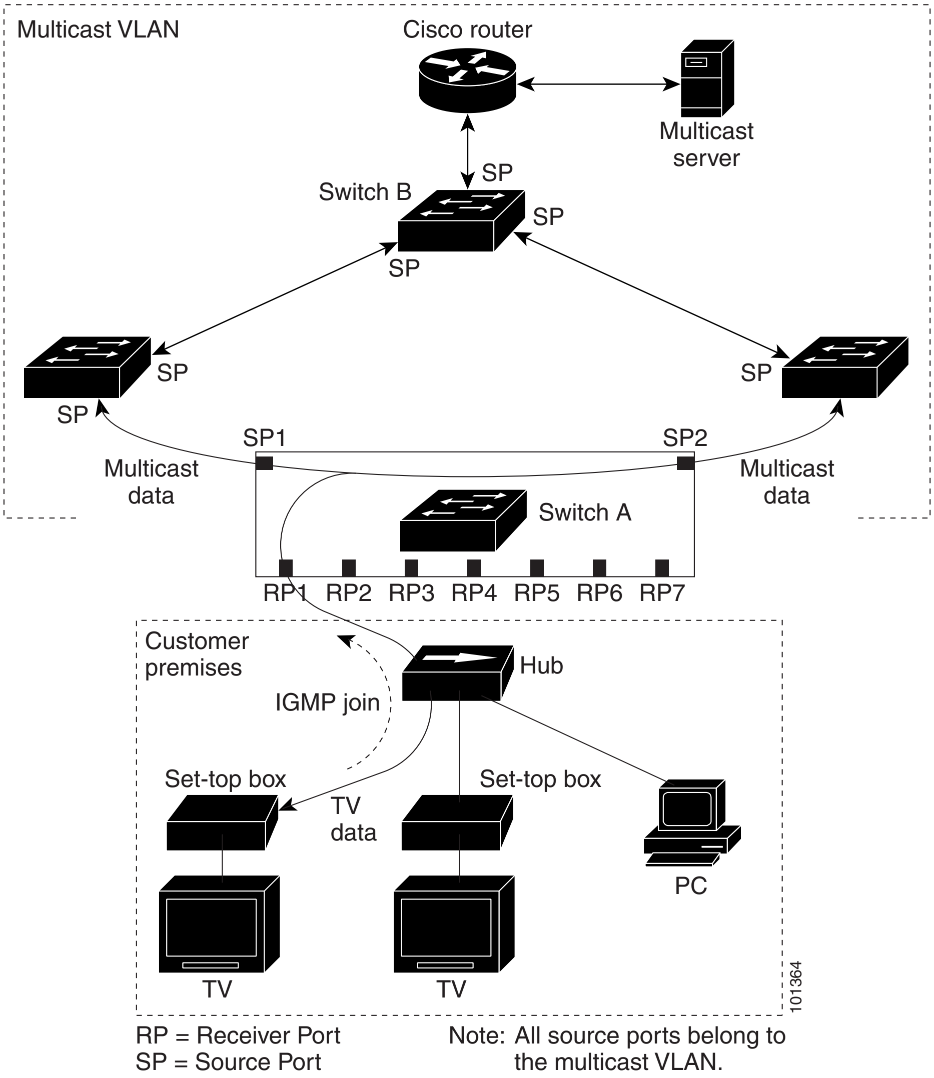
- In compatible mode, multicast data received by MVR hosts is forwarded to all MVR data ports, regardless of MVR host membership on those ports. The multicast data is forwarded only to those receiver ports that MVR hosts have joined, either by IGMP reports or by MVR static configuration. IGMP reports received from MVR hosts are never forwarded from MVR data ports that were configured in the switch.
- In dynamic mode, multicast data received by MVR hosts on the switch is forwarded from only those MVR data and client ports that the MVR hosts have joined, either by IGMP reports or by MVR static configuration. Any IGMP reports received from MVR hosts are also forwarded from all the MVR data ports in the switch. This eliminates using unnecessary bandwidth on MVR data port links, which occurs when the switch runs in compatible mode.

Only Layer 2 ports take part in MVR. You must configure ports as MVR receiver ports. Only one MVR multicast VLAN per switch is supported.

MVR in a Multicast Television Application

In a multicast television application, a PC or a television with a set-top box can receive the multicast stream. Multiple set-top boxes or PCs can be connected to one subscriber port, which is a switch port configured as an MVR receiver port. Figure 67 on page 456 is an example configuration. DHCP assigns an IP address to the set-top box or the PC. When a subscriber selects a channel, the set-top box or PC sends an IGMP report to Switch A to join the appropriate multicast. If the IGMP report matches one of the configured IP multicast group addresses, the switch CPU modifies the hardware address table to include this receiver port and VLAN as a forwarding destination of the specified multicast stream when it is received from the multicast VLAN. Uplink ports that send and receive multicast data to and from the multicast VLAN are called MVR source ports.

Figure 67 Multicast VLAN Registration Example



When a subscriber changes channels or turns off the television, the set-top box sends an IGMP leave message for the multicast stream. The switch CPU sends a MAC-based general query through the receiver port VLAN. If there is another set-top box in the VLAN still subscribing to this group, that set-top box must respond within the maximum response time specified in the query. If the CPU does not receive a response, it eliminates the receiver port as a forwarding destination for this group.

Without Immediate Leave, when the switch receives an IGMP leave message from a subscriber on a receiver port, it sends out an IGMP query on that port and waits for IGMP group membership reports. If no reports are received in a configured time period, the receiver port is removed from multicast group membership. With Immediate Leave, an IGMP query is not sent from the receiver port on which the IGMP leave was received. As soon as the leave message is received, the receiver port is removed from multicast group membership, which speeds up leave latency. Enable the Immediate Leave feature only on receiver ports to which a single receiver device is connected.

MVR eliminates the need to duplicate television-channel multicast traffic for subscribers in each VLAN. Multicast traffic for all channels is only sent around the VLAN trunk once—only on the multicast VLAN. The IGMP leave and join messages are in the VLAN to which the subscriber port is assigned. These messages dynamically register for streams of multicast traffic in the multicast VLAN on the Layer 3 device. Switch B. The access layer switch, Switch A, modifies the forwarding behavior to allow the traffic to be forwarded from the multicast VLAN to the subscriber port in a different VLAN, selectively allowing traffic to cross between two VLANs.

IGMP reports are sent to the same IP multicast group address as the multicast data. The Switch A CPU must capture all IGMP join and leave messages from receiver ports and forward them to the multicast VLAN of the source (uplink) port, based on the MVR mode.

Default MVR Settings

Feature	Default Setting
MVR	Disabled globally and per interface
Multicast addresses	None configured
Query response time	0.5 second
Multicast VLAN	VLAN 1
Mode	Compatible
Interface (per port) default	Neither a receiver nor a source port
Immediate Leave	Disabled on all ports

MVR Configuration Guidelines and Limitations

- Receiver ports can only be access ports; they cannot be trunk ports. Receiver ports on a switch can be in different VLANs, but should not belong to the multicast VLAN.
- The maximum number of multicast entries (MVR group addresses) that can be configured on a switch (that is, the maximum number of television channels that can be received) is 256.
- MVR multicast data received in the source VLAN and leaving from receiver ports has its time-to-live (TTL) decremented by 1 in the switch.
- Because MVR on the switch uses IP multicast addresses instead of MAC multicast addresses, aliased IP multicast addresses are allowed on the switch. However, if the switch is interoperating with Catalyst 3550 or Catalyst 3500 XL switches, you should not configure IP addresses that alias between themselves or with the reserved IP multicast addresses (in the range 224.0.0.xxx).
- Do not configure MVR on private VLAN ports.
- MVR is not supported when multicast routing is enabled on a switch. If you enable multicast routing and a multicast routing protocol while MVR is enabled, MVR is disabled, and you receive a warning message. If you try to enable MVR while multicast routing and a multicast routing protocol are enabled, the operation to enable MVR is cancelled, and you receive an error message.
- MVR can coexist with IGMP snooping on a switch.

- MVR data received on an MVR receiver port is not forwarded to MVR source ports.
- MVR does not support IGMPv3 messages.

IGMP Filtering and Throttling

In some environments, for example, metropolitan or multiple-dwelling unit (MDU) installations, you might want to control the set of multicast groups to which a user on a switch port can belong. You can control the distribution of multicast services, such as IP/TV, based on some type of subscription or service plan. You might also want to limit the number of multicast groups to which a user on a switch port can belong.

With the IGMP filtering feature, you can filter multicast joins on a per-port basis by configuring IP multicast profiles and associating them with individual switch ports. An IGMP profile can contain one or more multicast groups and specifies whether access to the group is permitted or denied. If an IGMP profile denying access to a multicast group is applied to a switch port, the IGMP join report requesting the stream of IP multicast traffic is dropped, and the port is not allowed to receive IP multicast traffic from that group. If the filtering action permits access to the multicast group, the IGMP report from the port is forwarded for normal processing. You can also set the maximum number of IGMP groups that a Layer 2 interface can join.

IGMP filtering controls only group-specific query and membership reports, including join and leave reports. It does not control general IGMP queries. IGMP filtering has no relationship with the function that directs the forwarding of IP multicast traffic. The filtering feature operates in the same manner whether CGMP or MVR is used to forward the multicast traffic.

IGMP filtering is applicable only to the dynamic learning of IP multicast group addresses, not static configuration.

With the IGMP throttling feature, you can set the maximum number of IGMP groups that a Layer 2 interface can join. If the maximum number of IGMP groups is set, the IGMP snooping forwarding table contains the maximum number of entries, and the interface receives an IGMP join report, you can configure an interface to drop the IGMP report or to replace the randomly selected multicast entry with the received IGMP report.

Note: IGMPv3 join and leave messages are not supported on switches running IGMP filtering.

Default IGMP Filtering and Throttling Configuration

Feature	Default Setting
IGMP filters	None applied
IGMP maximum number of IGMP groups	No maximum set
IGMP profiles	None defined
IGMP profile action	Deny the range addresses

When the maximum number of groups is in forwarding table, the default IGMP throttling action is to deny the IGMP report.

IGMP Profiles

To configure an IGMP profile, use the **ip igmp profile** global configuration command with a profile number to create an IGMP profile and to enter IGMP profile configuration mode. From this mode, you can specify the parameters of the IGMP profile to be used for filtering IGMP join requests from a port. When you are in IGMP profile configuration mode, you can create the profile by using these commands:

- **deny**—Specifies that matching addresses are denied; this is the default.
- **exit**—Exits from igmp-profile configuration mode.
- **no**—Negates a command or returns to its defaults.

- **permit**–Specifies that matching addresses are permitted.
- **range**–Specifies a range of IP addresses for the profile. You can enter a single IP address or a range with a start and an end address.

The default is for the switch to have no IGMP profiles configured. When a profile is configured, if neither the **permit** nor **deny** keyword is included, the default is to deny access to the range of IP addresses.

To control access as defined in an IGMP profile, use the **ip igmp filter** interface configuration command to apply the profile to the appropriate interfaces. You can apply IGMP profiles only to Layer 2 access ports; you cannot apply IGMP profiles to routed ports or SVIs. You cannot apply profiles to ports that belong to an EtherChannel port group. You can apply a profile to multiple interfaces, but each interface can have only one profile applied to it.

IGMP Throttling Action

After you set the maximum number of IGMP groups that a Layer 2 interface can join, you can configure an interface to replace the existing group with the new group for which the IGMP report was received by using the **ip igmp max-groups action replace** interface configuration command. Use the **no** form of this command to return to the default, which is to drop the IGMP join report.

Follow these guidelines when configuring the IGMP throttling action:

- This restriction can be applied only to Layer 2 ports. You can use this command on a logical EtherChannel interface but cannot use it on ports that belong to an EtherChannel port group.
- When the maximum group limitation is set to the default (no maximum), entering the **ip igmp max-groups action** {**deny** | **replace**} command has no effect.
- If you configure the throttling action and set the maximum group limitation after an interface has added multicast entries to the forwarding table, the forwarding-table entries are either aged out or removed, depending on the throttling action.
  - If you configure the throttling action as **deny**, the entries that were previously in the forwarding table are not removed but are aged out. After these entries are aged out and the maximum number of entries is in the forwarding table, the switch drops the next IGMP report received on the interface.
  - If you configure the throttling action as **replace**, the entries that were previously in the forwarding table are removed. When the maximum number of entries is in the forwarding table, the switch replaces a randomly selected entry with the received IGMP report.

  To prevent the switch from removing the forwarding-table entries, you can configure the IGMP throttling action before an interface adds entries to the forwarding table.

How to Configure IGMP Snooping and MVR

Configuring IGMP Snooping

Enabling or Disabling IGMP Snooping

By default, IGMP snooping is globally enabled on the switch. When globally enabled or disabled, it is also enabled or disabled in all existing VLAN interfaces. IGMP snooping is by default enabled on all VLANs, but can be enabled and disabled on a per-VLAN basis.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**ip igmp snooping** or **ip igmp snooping vlan** *vlan-id*	Globally enables IGMP snooping in all existing VLAN interfaces. or Enables IGMP snooping on the VLAN interface. The VLAN ID range is 1 to 1001 and 1006 to 4096. IGMP snooping must be globally enabled before you can enable VLAN snooping.
3.	**end**	Returns to privileged EXEC mode.

Setting IGMP Snooping Parameters

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**ip igmp snooping vlan** *vlan-id* **mrouter learn** {**cgmp** | **pim-dvmrp**}	(Optional) Enables IGMP snooping on a VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4096. Specifies the multicast router learning method: ■ **cgmp**–Listens for CGMP packets. This method is useful for reducing control traffic. ■ **pim-dvmrp**–Snoops on IGMP queries and PIM-DVMRP packets. This is the default.
3.	**ip igmp snooping vlan** *vlan-id* **mrouter interface** *interface-id*	Adds a multicast router port (adds a static connection to a multicast router). (Optional) Specifies the multicast router VLAN ID and the interface to the multicast router. ■ The VLAN ID range is 1 to 1001 and 1006 to 4096. ■ The interface can be a physical interface or a port channel. The port-channel range is 1 to 10. ■ Static connections to multicast routers are supported only on switch ports.
4.	**ip igmp snooping vlan** *vlan-id* **static** *ip_address* **interface** *interface-id*	(Optional) Statically configures a Layer 2 port as a member of a multicast group: ■ *vlan-id*–Multicast group VLAN ID. The range is 1 to 1001 and 1006 to 4096. ■ *ip-address*–Group IP address. ■ *interface-id*–Member port. It can be a physical interface or a port channel (1 to 6).
5.	**ip igmp snooping vlan** *vlan-id* **immediate-leave**	(Optional) Enables IGMP Immediate Leave on the VLAN interface. **Note:** Immediate Leave is supported only on IGMP Version 2 hosts.

	Command	Purpose
6.	**ip igmp snooping last-member-query-interval** *time*	(Optional) Configures the IGMP leave timer globally. The range is 100 to 32768 milliseconds. The default is 1000 seconds.
7.	**ip igmp snooping vlan** *vlan-id* **last-member-query-interval** *time*	(Optional) Configures the IGMP leave time on the VLAN interface. The range is 100 to 32768 milliseconds. **Note:** Configuring the leave time on a VLAN overrides the globally configured timer.
8.	**end**	Returns to privileged EXEC mode.

Configuring TCN

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**ip igmp snooping tcn flood query count** *count*	Specifies the number of IGMP general queries for which the multicast traffic is flooded. The range is 1 to 10. By default, the flooding query count is 2.
3.	**ip igmp snooping tcn query solicit**	Sends an IGMP leave message (global leave) to speed the process of recovering from the flood mode caused during a TCN event. By default, query solicitation is disabled. **Note:** Enable the switch to send the global leave message whether or not it is the spanning-tree root.
4.	**interface** *interface-id*	Specifies the interface to be configured, and enter interface configuration mode.
5.	**no ip igmp snooping tcn flood**	Disables the flooding of multicast traffic during a spanning-tree TCN event. By default, multicast flooding is enabled on an interface.
6.	**end**	Returns to privileged EXEC mode.

Configuring the IGMP Snooping Querier

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**ip igmp snooping querier**	Enables the IGMP snooping querier.
3.	**ip igmp snooping querier address** *ip_address*	(Optional) Specifies an IP address for the IGMP snooping querier. If you do not specify an IP address, the querier tries to use the global IP address configured for the IGMP querier. **Note:** The IGMP snooping querier does not generate an IGMP general query if it cannot find an IP address on the switch.
4.	**ip igmp snooping querier query-interval** *interval-count*	(Optional) Sets the interval between IGMP queriers. The range is 1 to 18000 seconds.
5.	**ip igmp snooping querier tcn query** [**count** *count* | **interval** *interval*]	(Optional) Sets the time between Topology Change Notification (TCN) queries. The count range is 1 to 10. The interval range is 1 to 255 seconds.

	Command	Purpose
6.	**ip igmp snooping querier timer expiry** *timeout*	(Optional) Sets the length of time until the IGMP querier expires. The range is 60 to 300 seconds.
7.	**ip igmp snooping querier version** *version*	(Optional) Selects the IGMP version number that the querier feature uses. Select 1 or 2.
8.	**end**	Returns to privileged EXEC mode.

Disabling IGMP Report Suppression

Before You Begin

IGMP report suppression is supported only when the multicast query has IGMPv1 and IGMPv2 reports. This feature is not supported when the query includes IGMPv3 reports.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**no ip igmp snooping report-suppression**	Disables IGMP report suppression.
3.	**end**	Returns to privileged EXEC mode.

Configuring MVR

Configuring MVR Global Parameters

You do not need to set the optional MVR parameters if you choose to use the default settings. If you do want to change the default parameters (except for the MVR VLAN), you must first enable MVR.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**mvr**	Enables MVR on the switch.
3.	**mvr group** *ip-address* [*count*]	Configures an IP multicast address on the switch or use the *count* parameter to configure a contiguous series of MVR group addresses (the range for *count* is 1 to 256; the default is 1). Any multicast data sent to this address is sent to all source ports on the switch and all receiver ports that have elected to receive data on that multicast address. Each multicast address would correspond to one television channel.
4.	**mvr querytime** *value*	(Optional) Defines the maximum time to wait for IGMP report memberships on a receiver port before removing the port from multicast group membership. The value is in units of tenths of a second. The range is 1 to 100, and the default is 5 tenths or one-half second.

	Command	Purpose
5.	**mvr vlan** *vlan-id*	(Optional) Specifies the VLAN in which multicast data is received; all source ports must belong to this VLAN. The VLAN range is 1 to 1001 and 1006 to 4096. The default is VLAN 1.
6.	**mvr mode** {**dynamic** | **compatible**}	(Optional) Specifies the MVR mode of operation: ■ **dynamic**–Allows dynamic MVR membership on source ports. ■ **compatible**–Is compatible with Catalyst 3500 XL and Catalyst 2900 XL switches and does not support IGMP dynamic joins on source ports. The default is **compatible** mode.
7.	**end**	Returns to privileged EXEC mode.

Configuring MVR Interfaces

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**mvr**	Enables MVR on the switch.
3.	**interface** *interface-id*	Specifies the Layer 2 port to configure, and enters interface configuration mode.
4.	**mvr type** {**source** | **receiver**}	Configures an MVR port as one of these: ■ **source**–Configures uplink ports that receive and send multicast data as source ports. Subscribers cannot be directly connected to source ports. All source ports on a switch belong to the single multicast VLAN. ■ **receiver**–Configures a port as a receiver port if it is a subscriber port and should only receive multicast data. It does not receive data unless it becomes a member of the multicast group, either statically or by using IGMP leave and join messages. Receiver ports cannot belong to the multicast VLAN. The default configuration is as a non-MVR port. If you attempt to configure a non-MVR port with MVR characteristics, the operation fails.
5.	**mvr vlan** *vlan-id* **group** [*ip-address*]	(Optional) Statically configures a port to receive multicast traffic sent to the multicast VLAN and the IP multicast address. A port statically configured as a member of a group remains a member of the group until statically removed. **Note:** In compatible mode, this command applies to only receiver ports. In dynamic mode, it applies to receiver ports and source ports. Receiver ports can also dynamically join multicast groups by using IGMP join and leave messages.
6.	**mvr immediate**	(Optional) Enables the Immediate-Leave feature of MVR on the port. **Note:** This command applies to only receiver ports and should only be enabled on receiver ports to which a single receiver device is connected.
7.	**end**	Returns to privileged EXEC mode.

Configuring IGMP

Configuring IGMP Profiles

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**ip igmp profile** *profile number*	Assigns a number to the profile you are configuring, and enter IGMP profile configuration mode. The profile number range is 1 to 4294967295.
3.	**permit | deny**	(Optional) Sets the action to permit or deny access to the IP multicast address. If no action is configured, the default for the profile is to deny access.
4.	**range** *ip multicast address*	Enters the IP multicast address or range of IP multicast addresses to which access is being controlled. If entering a range, enter the low IP multicast address, a space, and the high IP multicast address. You can use the **range** command multiple times to enter multiple addresses or ranges of addresses.
5.	**end**	Returns to privileged EXEC mode.

Configuring IGMP Interfaces

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the physical interface, and enter interface configuration mode. The interface must be a Layer 2 port that does not belong to an EtherChannel port group.
3.	**ip igmp filter** *profile number*	Applies the specified IGMP profile to the interface. The range is 1 to 4294967295.
4.	**ip igmp max-groups** *number*	Sets the maximum number of IGMP groups that the interface can join. The range is 0 to 4294967294. The default is to have no maximum set.
5.	**ip igmp max-groups action** {**deny | replace**}	When an interface receives an IGMP report and the maximum number of entries is in the forwarding table, specify the action that the interface takes: ■ **deny**—Drops the report. ■ **replace**—Replaces the existing group with the new group for which the IGMP report was received.
6.	**end**	Returns to privileged EXEC mode.

Monitoring and Maintaining IGMP Snooping and MVR

Command	Purpose
show ip igmp snooping [**vlan** *vlan-id*]	Displays the snooping configuration information for all VLANs on the switch or for a specified VLAN. (Optional) Enter **vlan** *vlan-id* to display information for a single VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4096.
show ip igmp snooping groups [**count** |**dynamic** [**count**] | **user** [**count**]]	Displays multicast table information for the switch or about a specific parameter: ■ **count**–Displays the total number of entries for the specified command options instead of the actual entries. ■ **dynamic**–Displays entries learned through IGMP snooping. ■ **user**–Displays only the user-configured multicast entries.
show ip igmp snooping groups vlan *vlan-id* [*ip_address* | **count** | **dynamic** [**count**] | **user**[**count**]]	Displays multicast table information for a multicast VLAN or about a specific parameter for the VLAN: ■ *vlan-id*–The VLAN ID range is 1 to 1001 and 1006 to 4096. ■ **count**–Displays the total number of entries for the specified command options instead of the actual entries. ■ **dynamic**–Displays entries learned through IGMP snooping. ■ *ip_address*–Displays characteristics of the multicast group with the specified group IP address. ■ **user**–Displays only the user-configured multicast entries.
show ip igmp snooping mrouter [**vlan** *vlan-id*]	Displays information on dynamically learned and manually configured multicast router interfaces. **Note:** When you enable IGMP snooping, the switch automatically learns the interface to which a multicast router is connected. These are dynamically learned interfaces. (Optional) Enter **vlan** *vlan-id* to display information for a single VLAN.
show ip igmp snooping querier [**vlan** *vlan-id*]	Displays information about the IP address and receiving port for the most-recently received IGMP query messages in the VLAN. (Optional) Enter **vlan** *vlan-id* to display information for a single VLAN.
show ip igmp snooping querier [**vlan** *vlan-id*] **detail**	Displays information about the IP address and receiving port of the most-recently received IGMP query message in the VLAN and the configuration and operational state of the IGMP snooping querier in the VLAN.
show ip igmp profile [*profile number*]	Displays the specified IGMP profile or all the IGMP profiles defined on the switch.
show mvr	Displays MVR status and values for the switch–whether MVR is enabled or disabled, the multicast VLAN, the maximum (256) and current (0 through 256) number of multicast groups, the query response time, and the MVR mode.

Command	Purpose
show mvr interface [*interface-id*] [**members** [**vlan** *vlan-**id***]]	Displays all MVR interfaces and their MVR configurations. When a specific interface is entered, displays this information: ■ Type—Receiver or Source ■ Status—One of these: – Active means the port is part of a VLAN. – Up/Down means that the port is forwarding or nonforwarding. – Inactive means that the port is not part of any VLAN. ■ Immediate Leave—Enabled or Disabled If the **members** keyword is entered, displays all multicast group members on this port or, if a VLAN identification is entered, all multicast group members on the VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4096.
show mvr members [*ip-address*]	Displays all receiver and source ports that are members of any IP multicast group or the specified IP multicast group IP address.
show ip igmp profile *profile number*	Verifies the profile configuration.
show ip igmp snooping mrouter [**vlan** *vlan-id*]	Verifies that IGMP snooping is enabled on the VLAN interface.

Configuration Examples for IGMP Snooping

Configuring IGMP Snooping: Example

This example shows how to configure IGMP snooping to use CGMP packets as the learning method:

```
Switch# configure terminal
Switch(config)# ip igmp snooping vlan 1 mrouter learn cgmp
Switch(config)# end
```

Disabling a Multicast Router Port: Example

To remove a multicast router port from the VLAN, use the **no ip igmp snooping vlan** *vlan-id* **mrouter interface** *interface-id* global configuration command.

This example shows how to enable a static connection to a multicast router:

```
Switch# configure terminal
Switch(config)# ip igmp snooping vlan 200 mrouter interface GigabitEthernet1/18
Switch(config)# end
```

Statically Configuring a Host on a Port: Example

This example shows how to statically configure a host on a port:

```
Switch# configure terminal
Switch(config)# ip igmp snooping vlan 105 static 224.2.4.12 interface gigabitethernet1/1
Switch(config)# end
```

Enabling IGMP Immediate Leave: Example

This example shows how to enable IGMP Immediate Leave on VLAN 130:

```
Switch# configure terminal
Switch(config)# ip igmp snooping vlan 130 immediate-leave
Switch(config)# end
```

Setting the IGMP Snoopng Querier Parameters: Examples

This example shows how to set the IGMP snooping querier source address to 10.0.0.64:

```
Switch# configure terminal
Switch(config)# ip igmp snooping querier 10.0.0.64
Switch(config)# end
```

This example shows how to set the IGMP snooping querier maximum response time to 25 seconds:

```
Switch# configure terminal
Switch(config)# ip igmp snooping querier query-interval 25
Switch(config)# end
```

This example shows how to set the IGMP snooping querier timeout to 60 seconds:

```
Switch# configure terminal
Switch(config)# ip igmp snooping querier timeout expiry 60
Switch(config)# end
```

This example shows how to set the IGMP snooping querier feature to version 2:

```
Switch# configure terminal
Switch(config)# no ip igmp snooping querier version 2
Switch(config)# end
```

Enabling MVR: Examples

This example shows how to enable MVR, configure the group address, set the query time to 1 second (10 tenths), specify the MVR multicast VLAN as VLAN 22, and set the MVR mode as dynamic:

```
Switch(config)# mvr
Switch(config)# mvr group 228.1.23.4
Switch(config)# mvr querytime 10
Switch(config)# mvr vlan 22
Switch(config)# mvr mode dynamic
Switch(config)# end
```

You can use the **show mvr members** privileged EXEC command to verify the MVR multicast group addresses on the switch.

This example shows how to configure a port as a receiver port, statically configure the port to receive multicast traffic sent to the multicast group address, configure Immediate Leave on the port, and verify the results:

```
Switch(config)# mvr
Switch(config)# interface GigabitEthernet1/18
Switch(config-if)# mvr type receiver
Switch(config-if)# mvr vlan 22 group 228.1.23.4
Switch(config-if)# mvr immediate
Switch(config)# end
Switch# show mvr interface
Port    Type        Status          Immediate Leave
```

```
----    ----        -------         ---------------
Gi1/18   RECEIVER    ACTIVE/DOWN     ENABLED
```

Creating an IGMP Profile: Example

This example shows how to create IGMP profile 4 allowing access to the single IP multicast address and how to verify the configuration. If the action was to deny (the default), it would not appear in the **show ip igmp profile** output display.

```
Switch(config)# ip igmp profile 4
Switch(config-igmp-profile)# permit
Switch(config-igmp-profile)# range 229.9.9.0
Switch(config-igmp-profile)# end
Switch# show ip igmp profile 4
IGMP Profile 4
    permit
    range 229.9.9.0 229.9.9.0
```

Applying an IGMP Profile: Example

This example shows how to apply IGMP profile 4 to a port:

```
Switch(config)# interface GigabitEthernet1/18
Switch(config-if)# ip igmp filter 4
Switch(config-if)# end
```

Limiting IGMP Groups: Example

This example shows how to limit to 25 the number of IGMP groups that a port can join:

```
Switch(config)# interface GigabitEthernet1/18
Switch(config-if)# ip igmp max-groups 25
Switch(config-if)# end
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*
Cisco IOS multicast commands	*Cisco IOS IP Command Reference, Volume 3 of 3:Multicast*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Configuring Port-Based Traffic Control

Restrictions for Port-Based Traffic Control

- To use this feature, the switch must be running the LAN Base image.

Information About Port-Based Traffic Control

Storm Control

Storm control prevents traffic on a LAN from being disrupted by a broadcast, multicast, or unicast storm on one of the physical interfaces. A LAN storm occurs when packets flood the LAN, creating excessive traffic and degrading network performance. Errors in the protocol-stack implementation, mistakes in network configurations, or users issuing a denial-of-service attack can cause a storm.

Storm control (or traffic suppression) monitors packets passing from an interface to the switching bus and determines if the packet is unicast, multicast, or broadcast. The switch counts the number of packets of a specified type received within the 1-second time interval and compares the measurement with a predefined suppression-level threshold.

Storm control uses one of these methods to measure traffic activity:

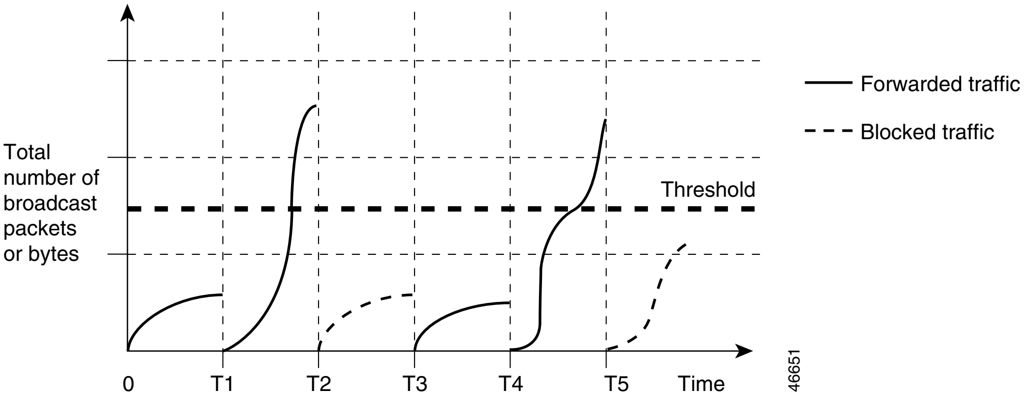
- Bandwidth as a percentage of the total available bandwidth of the port that can be used by the broadcast, multicast, or unicast traffic
- Traffic rate in packets per second at which broadcast, multicast, or unicast packets are received.
- Traffic rate in bits per second at which broadcast, multicast, or unicast packets are received.
- Traffic rate in packets per second and for small frames. This feature is enabled globally. The threshold for small frames is configured for each interface.

With each method, the port blocks traffic when the rising threshold is reached. The port remains blocked until the traffic rate drops below the falling threshold (if one is specified) and then resumes normal forwarding. If the falling suppression level is not specified, the switch blocks all traffic until the traffic rate drops below the rising suppression level. In general, the higher the level, the less effective the protection against broadcast storms.

Note: When the storm control threshold for multicast traffic is reached, all multicast traffic except control traffic, such as bridge protocol data unit (BDPU) and Cisco Discovery Protocol (CDP) frames, are blocked. However, the switch does not differentiate between routing updates, such as OSPF, and regular multicast data traffic, so both types of traffic are blocked.

The graph in Figure 68 on page 472 shows broadcast traffic patterns on an interface over a given period of time. The example can also be applied to multicast and unicast traffic. In this example, the broadcast traffic being forwarded exceeded the configured threshold between time intervals T1 and T2 and between T4 and T5. When the amount of specified traffic exceeds the threshold, all traffic of that kind is dropped for the next time period. Therefore, broadcast traffic is blocked during the intervals following T2 and T5. At the next time interval (for example, T3), if broadcast traffic does not exceed the threshold, it is again forwarded.

Figure 68 Broadcast Storm Control Example



The combination of the storm-control suppression level and the 1-second time interval controls the way the storm control algorithm works. A higher threshold allows more packets to pass through. A threshold value of 100 percent means that no limit is placed on the traffic. A value of 0.0 means that all broadcast, multicast, or unicast traffic on that port is blocked.

Note: Because packets do not arrive at uniform intervals, the 1-second time interval during which traffic activity is measured can affect the behavior of storm control.

You use the **storm-control** interface configuration commands to set the threshold value for each traffic type.

Default Storm Control Configuration

By default, unicast, broadcast, and multicast storm control are disabled on the switch interfaces; that is, the suppression level is 100 percent.

Storm Control and Threshold Levels

You configure storm control on a port and enter the threshold level that you want to be used for a particular type of traffic.

However, because of hardware limitations and the way in which packets of different sizes are counted, threshold percentages are approximations. Depending on the sizes of the packets making up the incoming traffic, the actual enforced threshold might differ from the configured level by several percentage points.

Note: Storm control is supported on physical interfaces. You can also configure storm control on an EtherChannel. When storm control is configured on an EtherChannel, the storm control settings propagate to the EtherChannel physical interfaces.

Small-Frame Arrival Rate

Incoming VLAN-tagged packets smaller than 67 bytes are considered *small frames*. They are forwarded by the switch, but they do not cause the switch storm-control counters to increment. In Cisco IOS Release 12.2(44)SE and later, you can configure a port to be error disabled if small frames arrive at a specified rate (threshold).

You globally enable the small-frame arrival feature on the switch and then configure the small-frame threshold for packets on each interface. Packets smaller than the minimum size and arriving at a specified rate (the threshold) are dropped since the port is error disabled.

If the **errdisable recovery cause small-frame** global configuration command is entered, the port is reenabled after a specified time. (You specify the recovery time by using **errdisable recovery** global configuration command.)

Protected Ports

Some applications require that no traffic be forwarded at Layer 2 between ports on the same switch so that one neighbor does not see the traffic generated by another neighbor. In such an environment, the use of protected ports ensures that there is no exchange of unicast, broadcast, or multicast traffic between these ports on the switch.

Protected ports have these features:

- A protected port does not forward any traffic (unicast, multicast, or broadcast) to any other port that is also a protected port. Data traffic cannot be forwarded between protected ports at Layer 2; only control traffic, such as PIM packets, is forwarded because these packets are processed by the CPU and forwarded in software. All data traffic passing between protected ports must be forwarded through a Layer 3 device.
- Forwarding behavior between a protected port and a nonprotected port proceeds as usual.

Protected Port Configuration Guidelines

You can configure protected ports on a physical interface (for example, Gigabit Ethernet port 1) or an EtherChannel group (for example, port-channel 5). When you enable protected ports for a port channel, it is enabled for all ports in the port-channel group.

Do not configure a private-VLAN port as a protected port. Do not configure a protected port as a private-VLAN port. A private-VLAN isolated port does not forward traffic to other isolated ports or community ports.

Port Blocking

By default, the switch floods packets with unknown destination MAC addresses out of all ports. If unknown unicast and multicast traffic is forwarded to a protected port, there could be security issues. To prevent unknown unicast or multicast traffic from being forwarded from one port to another, you can block a port (protected or nonprotected) from flooding unknown unicast or multicast packets to other ports.

Note: With multicast traffic, the port blocking feature blocks only pure Layer 2 packets. Multicast packets that contain IPv4 or IPv6 information in the header are not blocked.

Port Security

You can use the port security feature to restrict input to an interface by limiting and identifying MAC addresses of the stations allowed to access the port. When you assign secure MAC addresses to a secure port, the port does not forward packets with source addresses outside the group of defined addresses. If you limit the number of secure MAC addresses to one and assign a single secure MAC address, the workstation attached to that port is assured the full bandwidth of the port.

If a port is configured as a secure port and the maximum number of secure MAC addresses is reached, when the MAC address of a station attempting to access the port is different from any of the identified secure MAC addresses, a security violation occurs. Also, if a station with a secure MAC address configured or learned on one secure port attempts to access another secure port, a violation is flagged.

Secure MAC Addresses

You configure the maximum number of secure addresses allowed on a port by using the **switchport port-security maximum** *value* interface configuration command.

Note: If you try to set the maximum value to a number less than the number of secure addresses already configured on an interface, the command is rejected.

The switch supports these types of secure MAC addresses:

- Static secure MAC addresses—These are manually configured by using the **switchport port-security mac-address** *mac-address* interface configuration command, stored in the address table, and added to the switch running configuration.
- Dynamic secure MAC addresses—These are dynamically configured, stored only in the address table, and removed when the switch restarts.
- *Sticky* secure MAC addresses—These can be dynamically learned or manually configured, stored in the address table, and added to the running configuration. If these addresses are saved in the configuration file, when the switch restarts, the interface does not need to dynamically reconfigure them.

You can configure an interface to convert the dynamic MAC addresses to sticky secure MAC addresses and to add them to the running configuration by enabling *sticky learning*. To enable sticky learning, enter the **switchport port-security mac-address sticky** interface configuration command. When you enter this command, the interface converts all the dynamic secure MAC addresses, including those that were dynamically learned before sticky learning was enabled, to sticky secure MAC addresses. All sticky secure MAC addresses are added to the running configuration.

The sticky secure MAC addresses do not automatically become part of the configuration file, which is the startup configuration used each time the switch restarts. If you save the sticky secure MAC addresses in the configuration file, when the switch restarts, the interface does not need to relearn these addresses. If you do not save the sticky secure addresses, they are lost.

If sticky learning is disabled, the sticky secure MAC addresses are converted to dynamic secure addresses and are removed from the running configuration.

The maximum number of secure MAC addresses that you can configure on a switch is set by the maximum number of available MAC addresses allowed in the system. This number is determined by the active Switch Database Management (SDM) template. This number is the total of available MAC addresses, including those used for other Layer 2 functions and any other secure MAC addresses configured on interfaces.

Security Violations

It is a security violation when one of these situations occurs:

- The maximum number of secure MAC addresses have been added to the address table, and a station whose MAC address is not in the address table attempts to access the interface.
- An address learned or configured on one secure interface is seen on another secure interface in the same VLAN and on the same switch.

You can configure the interface for one of five violation modes, based on the action to be taken if a violation occurs:

- protect—When the number of secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses to drop below the maximum value or increase the number of maximum allowable addresses. You are not notified that a security violation has occurred.

  We do not recommend configuring the protect violation mode on a trunk port. The protect mode disables learning when any VLAN reaches its maximum limit, even if the port has not reached its maximum limit.

- restrict—When the number of secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses to drop below the maximum value or increase the number of maximum allowable addresses. In this mode, you are notified that a security violation has occurred. An SNMP trap is sent, a syslog message is logged, and the violation counter increments.
- shutdown—A port security violation causes the interface to become error-disabled and to shut down immediately, and the port LED turns off. An SNMP trap is sent, a syslog message is logged, and the violation counter increments. When a secure port is in the error-disabled state, you can bring it out of this state by entering the **errdisable recovery cause** *psecure-violation* global configuration command, or you can manually reenable it by entering the **shutdown** and **no shut down** interface configuration commands. This is the default mode.

- shutdown vlan—Use to set the security violation mode per-VLAN. In this mode, the VLAN is error disabled instead of the entire port when a violation occurs.
- report—This mode is similar to the restrict option, except that when the number of secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are not dropped. Packets from an unknown source are allowed, but you are notified that a security violation has occurred. A syslog message is logged with the unknown MAC address, and the violation counter also increments.

  Note: If you use report mode, do not configure the **switchport port-security** global command on the ingress interface.

  To switch between the report violation mode and another mode, you must explicitly disable all port-security commands of the previous mode using the corresponding **no** command before adding the new configuration.

Table 48 Security Violation Mode Actions

Violation Mode	Traffic is Forwarded[1]	Sends SNMP Trap	Sends syslog Message	Displays Error Message[2]	Violation Counter Increments	Shuts Down Port
protect	No	No	No	No	No	No
restrict	No	Yes	Yes	No	Yes	No
shutdown	No	No	No	No	Yes	Yes
shutdown vlan	No	No	Yes	No	Yes	No[3]
report	Yes	No	Yes	Yes	Yes	No

[1] Packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses.

[2] The switch returns an error message if you manually configure an address that would cause a security violation.

[3] Shuts down only the VLAN on which the violation occurred.

Default Port Security Configuration

Feature	Default Setting
Port security	Disabled on a port.
Sticky address learning	Disabled.
Maximum number of secure MAC addresses per port	1
Violation mode	Shutdown. The port shuts down when the maximum number of secure MAC addresses is exceeded.
Port security aging	Disabled. Aging time is 0. Static aging is disabled. Type is absolute.

Port Security Configuration Guidelines

- Port security can only be configured on static access ports or trunk ports. A secure port cannot be a dynamic access port.
- A secure port cannot be a destination port for Switched Port Analyzer (SPAN).

- A secure port cannot belong to a Fast EtherChannel port group.

  Voice VLAN is only supported on access ports and not on trunk ports, even though the configuration is allowed.

- When you enable port security on an interface that is also configured with a voice VLAN, set the maximum allowed secure addresses on the port to two. When the port is connected to a Cisco IP phone, the IP phone requires one MAC address. The Cisco IP phone address is learned on the voice VLAN, but is not learned on the access VLAN. If you connect a single PC to the Cisco IP phone, no additional MAC addresses are required. If you connect more than one PC to the Cisco IP phone, you must configure enough secure addresses to allow one for each PC and one for the phone.

- When a trunk port configured with port security and assigned to an access VLAN for data traffic and to a voice VLAN for voice traffic, entering the **switchport voice** and **switchport priority extend** interface configuration commands has no effect.

  When a connected device uses the same MAC address to request an IP address for the access VLAN and then an IP address for the voice VLAN, only the access VLAN is assigned an IP address.

- When configuring port security, first specify the total number of MAC addresses you want to allow, by using the **switchport port-security maximum** interface configuration command and then configure the number of access VLANs (**switchport port-security vlan access** interface configuration command) and voice VLANs (**switchport port-security vlan voice** interface configuration command) you want to allow. If you do not specify the total number first, the system returns to the default setting (1 MAC address).

- When you enter a maximum secure address value for an interface, and the new value is greater than the previous value, the new value overwrites the previously configured value. If the new value is less than the previous value and the number of configured secure addresses on the interface exceeds the new value, the command is rejected.

- The switch does not support port security aging of sticky secure MAC addresses.

Type of Port or Feature on Port	Compatible with Port Security
DTP[1] port[2]	No
Trunk port	Yes
Dynamic-access port[3]	No
Routed port	No
SPAN source port	Yes
SPAN destination port	No
EtherChannel	No
Tunneling port	Yes
Protected port	Yes
IEEE 802.1x port	Yes
Voice VLAN port[4]	Yes
Private VLAN port	Yes
IP source guard	Yes
Dynamic Address Resolution Protocol (ARP) inspection	Yes
FlexLinks	Yes

[1] DTP = Dynamic Trunking Protocol

[2] A port configured with the **switchport mode dynamic** interface configuration command.

[3] A VLAN Query Protocol (VQP) port configured with the **switchport access vlan dynamic** interface configuration command.

[4] You must set the maximum allowed secure addresses on the port to two plus the maximum number of secure addresses allowed on the access VLAN.

Port Security Aging

You can use port security aging to set the aging time for all secure addresses on a port. Two types of aging are supported per port:

- Absolute—The secure addresses on the port are deleted after the specified aging time.
- Inactivity—The secure addresses on the port are deleted only if the secure addresses are inactive for the specified aging time.

Use this feature to remove and add devices on a secure port without manually deleting the existing secure MAC addresses and to still limit the number of secure addresses on a port. You can enable or disable the aging of secure addresses on a per-port basis.

Port Security and Private VLANs

Ports that have both port security and private VLANs (PVLANs) configured can be labeled secure PVLAN ports. When a secure address is learned on a secure PVLAN port, the same secure address cannot be learned on another secure PVLAN port belonging to the same primary VLAN. However, an address learned on unsecure PVLAN port can be learned on a secure PVLAN port belonging to same primary VLAN.

Secure addresses that are learned on host port get automatically replicated on associated primary VLANs, and similarly, secure addresses learned on promiscuous ports automatically get replicated on all associated secondary VLANs. Static addresses (using the **mac-address-table static** command) cannot be user configured on a secure port.

Protocol Storm Protection

When a switch is flooded with Address Resolution Protocol (ARP) or control packets, high CPU utilization can cause the CPU to overload. These issues can occur:

- Routing protocol can flap because the protocol control packets are not received, and neighboring adjacencies are dropped.
- Spanning Tree Protocol (STP) reconverges because the STP bridge protocol data unit (BPDU) cannot be sent or received.
- CLI is slow or unresponsive.

Using protocol storm protection, you can control the rate at which control packets are sent to the switch by specifying the upper threshold for the packet flow rate. The supported protocols are ARP, ARP snooping, Dynamic Host Configuration Protocol (DHCP) v4, DHCP snooping, Internet Group Management Protocol (IGMP), and IGMP snooping.

When the packet rate exceeds the defined threshold, the switch drops all traffic arriving on the specified virtual port for 30 seconds. The packet rate is measured again, and protocol storm protection is again applied if necessary.

For further protection, you can manually error disable the virtual port, blocking all incoming traffic on the virtual port. You can manually enable the virtual port or set a time interval for automatic reenabling of the virtual port.

Note: Excess packets are dropped on no more than two virtual ports.
Virtual port error disabling is not supported for EtherChannel and Flex Link interfaces.

Protocol storm protection is disabled by default. When it is enabled, auto-recovery of the virtual port is disabled by default.

How to Configure Port-Based Traffic Control

Configuring Storm Control

Configuring Storm Control and Threshold Levels

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the interface to be configured, and enters interface configuration mode.
3.	**storm-control** {**broadcast** | **multicast** | **unicast**} **level** {*level* [*level-low*] | **bps** *bps* [*bps-low*] | **pps** *pps* [*pps-low*]}	Configures broadcast, multicast, or unicast storm control. By default, storm control is disabled. ■ *level*–Specifies the rising threshold level for broadcast, multicast, or unicast traffic as a percentage (up to two decimal places) of the bandwidth. The port blocks traffic when the rising threshold is reached. The range is 0.00 to 100.00. ■ (Optional) *level-low*–Specifies the falling threshold level as a percentage (up to two decimal places) of the bandwidth. This value must be less than or equal to the rising suppression value. The port forwards traffic when traffic drops below this level. If you do not configure a falling suppression level, it is set to the rising suppression level. The range is 0.00 to 100.00. If you set the threshold to the maximum value (100 percent), no limit is placed on the traffic. If you set the threshold to 0.0, all broadcast, multicast, and unicast traffic on that port is blocked. ■ **bps** *bps*–Specifies the rising threshold level for broadcast, multicast, or unicast traffic in bits per second (up to one decimal place). The port blocks traffic when the rising threshold is reached. The range is 0.0 to 10000000000.0. ■ (Optional) *bps-low*–Specifies the falling threshold level in bits per second (up to one decimal place). It can be less than or equal to the rising threshold level. The port forwards traffic when traffic drops below this level. The range is 0.0 to 10000000000.0.

	Command	Purpose
		■ **pps** *pps*–Specifies the rising threshold level for broadcast, multicast, or unicast traffic in packets per second (up to one decimal place). The port blocks traffic when the rising threshold is reached. The range is 0.0 to 10000000000.0. ■ (Optional) *pps-low*–Specifies the falling threshold level in packets per second (up to one decimal place). It can be less than or equal to the rising threshold level. The port forwards traffic when traffic drops below this level. The range is **0.0 to** 10000000000.0. For BPS and PPS settings, you can use metric suffixes such as k, m, and g for large number thresholds.
4.	**storm-control action** {**shutdown** | **trap**}	Specifies the action to be taken when a storm is detected. The default is to filter out the traffic and not to send traps. ■ **shutdown**–Error-disables the port during a storm. ■ **trap**–Generates an SNMP trap when a storm is detected.
5.	**end**	Returns to privileged EXEC mode.

Configuring Small-Frame Arrival Rate

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**errdisable detect cause small-frame**	Enables the small-frame rate-arrival feature on the switch.
3.	**errdisable recovery interval** *interval*	(Optional) Specifies the time to recover from the specified error-disabled state.
4.	**errdisable recovery cause small-frame**	(Optional) Configures the recovery time for error-disabled ports to be automatically reenabled after they are error disabled by the arrival of small frames
5.	**interface** *interface-id*	Enters interface configuration mode, and specifies the interface to be configured.
6.	**small violation-rate** *pps*	Configures the threshold rate for the interface to drop incoming packets and error disable the port. The range is 1 to 10,000 packets per second (pps).
7.	**end**	Returns to privileged EXEC mode.

Configuring Protected Ports

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the interface to be configured, and enter interface configuration mode.
3.	**switchport protected**	Configures the interface to be a protected port.
4.	**end**	Returns to privileged EXEC mode.

Configuring Port Blocking

Blocking Flooded Traffic on an Interface

Note: The interface can be a physical interface or an EtherChannel group. When you block multicast or unicast traffic for a port channel, it is blocked on all ports in the port-channel group.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the interface to be configured, and enters interface configuration mode.
3.	**switchport block multicast**	Blocks unknown multicast forwarding out of the port. **Note:** Only pure Layer 2 multicast traffic is blocked. Multicast packets that contain IPv4 or IPv6 information in the header are not blocked.
4.	**switchport block unicast**	Blocks unknown unicast forwarding out of the port.
5.	**end**	Returns to privileged EXEC mode.

Configuring Port Security

Enabling and Configuring Port Security

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the interface to be configured, and enters interface configuration mode.
3.	**switchport mode** {**access | trunk**}	Sets the interface switchport mode as access or trunk. An interface in the default mode (dynamic auto) cannot be configured as a secure port.
4.	**switchport voice vlan** *vlan-id*	Enables voice VLAN on a port. *vlan-id*–Specifies the VLAN to be used for voice traffic.
5.	**switchport port-security**	Enables port security on the interface.

	Command	Purpose
6.	**switchport port-security** [**maximum** *value* [**vlan** {*vlan-list* | {**access** | **voice**}}]]	(Optional) **maximum**–Specifies the maximum number of secure MAC addresses on the port. By default only 1 MAC address is allowed. The maximum number of secure MAC addresses that you can configure on a switch is set by the maximum number of available MAC addresses allowed in the system. This number is set by the active Switch Database Management (SDM) template. This number is the total of available MAC addresses, including those used for other Layer 2 functions and any other secure MAC addresses configured on interfaces. (Optional) **vlan**–Sets a per-VLAN maximum value. Enter one of these options after you enter the **vlan** keyword: ■ *vlan-list*–On a trunk port, sets a per-VLAN maximum value on a range of VLANs separated by a hyphen or a series of VLANs separated by commas. For nonspecified VLANs, the per-VLAN maximum value is used. ■ **access**–On an access port, specifies the VLAN as an access VLAN. ■ **voice**–On an access port, specifies the VLAN as a voice VLAN. **Note:** The **voice** keyword is available only if a voice VLAN is configured on a port and if that port is not the access VLAN. If an interface is configured for voice VLAN, configure a maximum of two secure MAC addresses.

	Command	Purpose
7.	**switchport port-security [violation** {**protect | restrict | shutdown** | **shutdown vlan | report**}]	(Optional) Sets the violation mode, the action to be taken when a security violation is detected, as one of these: ■ **protect**–When the number of port secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses to drop below the maximum value or increase the number of maximum allowable addresses. You are not notified that a security violation has occurred. **Note:** We do not recommend configuring the protect mode on a trunk port. The protect mode disables learning when any VLAN reaches its maximum limit, even if the port has not reached its maximum limit. ■ **restrict**–When the number of secure MAC addresses reaches the limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses or increase the number of maximum allowable addresses. An SNMP trap is sent, a syslog message is logged, and the violation counter increments. ■ **shutdown**–The interface is error-disabled when a violation occurs, and the port LED turns off. An SNMP trap is sent, a syslog message is logged, and the violation counter increments. ■ **shutdown vlan**–Sets the security violation mode per VLAN. In this mode, the VLAN is error disabled instead of the entire port when a violation occurs. ■ **report**–This mode is similar to the restrict option, except that when the number of secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are not dropped. Packets from an unknown source are allowed, but you are notified that a security violation has occurred. A syslog message is logged with the unknown MAC address, and the violation counter also increments. **Note:** If you use report mode, do not configure the **switchport port-security** global command on the ingress interface. To switch between the report violation mode and another mode, you must explicitly disable all port-security commands of the previous mode using the corresponding **no** command before adding the new configuration. **Note:** When a secure port is in the error-disabled state, you can bring it out of this state by entering the **errdisable recovery cause psecure-violation** global configuration command. You can manually reenable it by entering the **shutdown** and **no shutdown** interface configuration commands or by using the **clear errdisable interface vlan** privileged EXEC command.

	Command	Purpose
8.	**switchport port-security** [**mac-address** *mac-address* [**vlan** {*vlan-id* | {**access / voice**}}]	(Optional) Enters a secure MAC address for the interface. You can use this command to enter the maximum number of secure MAC addresses. If you configure fewer secure MAC addresses than the maximum, the remaining MAC addresses are dynamically learned. **Note:** If you enable sticky learning after you enter this command, the secure addresses that were dynamically learned are converted to sticky secure MAC addresses and are added to the running configuration. (Optional) **vlan**–Sets a per-VLAN maximum value. Enter one of these options after you enter the **vlan** keyword: ■ *vlan-id*–On a trunk port, specifies the VLAN ID and the MAC address. If you do not specify a VLAN ID, the native VLAN is used. ■ **access**–On an access port, specifies the VLAN as an access VLAN. ■ **voice**–On an access port, specifies the VLAN as a voice VLAN. **Note:** The **voice** keyword is available only if a voice VLAN is configured on a port and if that port is not the access VLAN. If an interface is configured for voice VLAN, configure a maximum of two secure MAC addresses.
9.	**switchport port-security mac-address sticky**	(Optional) Enables sticky learning on the interface.
10.	**switchport port-security mac-address sticky** [*mac-address* | **vlan** {*vlan-id* | {**access** / **voice**}}]	(Optional) Enters a sticky secure MAC address, repeating the command as many times as necessary. If you configure fewer secure MAC addresses than the maximum, the remaining MAC addresses are dynamically learned, are converted to sticky secure MAC addresses, and are added to the running configuration. **Note:** If you do not enable sticky learning before this command is entered, an error message appears, and you cannot enter a sticky secure MAC address. (Optional) **vlan**–Sets a per-VLAN maximum value. Enter one of these options after you enter the **vlan** keyword: ■ *vlan-id*–On a trunk port, specifies the VLAN ID and the MAC address. If you do not specify a VLAN ID, the native VLAN is used. ■ **access**–On an access port, specifies the VLAN as an access VLAN. ■ **voice**–On an access port, specifies the VLAN as a voice VLAN. **Note:** The **voice** keyword is available only if a voice VLAN is configured on a port and if that port is not the access VLAN.
11.	**end**	Returns to privileged EXEC mode.

Enabling and Configuring Port Security Aging

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the interface to be configured, and enters interface configuration mode.
3.	**switchport port-security aging** {**static | time** *time* | **type** {**absolute | inactivity**}}	Enables or disables static aging for the secure port, or sets the aging time or type. **Note:** The switch does not support port security aging of sticky secure addresses. **static**–Enables aging for statically configured secure addresses on this port. **time**–Specifies the aging time for this port. The valid range is from 0 to 1440 minutes. **type**–Specifies the aging type as either absolute or inactivity. ■ **absolute**–All the secure addresses on this port age out exactly after the time *(*minutes) specified lapses and are removed from the secure address list. ■ **inactivity**–The secure addresses on this port age out only if there is no data traffic from the secure source addresses for the specified time period.
4.	**end**	Returns to privileged EXEC mode.

Configuring Protocol Storm Protection

Enabling Protocol Storm Protection

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**psp** {**arp** | **dhcp** | **igmp**} **pps** *value*	Configures protocol storm protection for ARP, IGMP, or DHCP. *value*–Specifies the threshold value for the number of packets per second. If the traffic exceeds this value, protocol storm protection is enforced. The range is from 5 to 50 packets per second.
3.	**errdisable detect cause psp**	(Optional) Enables error-disable detection for protocol storm protection. If this feature is enabled, the virtual port is error-disabled. If this feature is disabled, the port drops excess packets without error-disabling the port.
4.	**errdisable recovery interval** *time*	(Optional) Configures an auto-recovery time (in seconds) for error-disabled virtual ports. When a virtual port is error-disabled, the switch auto-recovers after this time. The range is from 30 to 86400 seconds.
5.	**end**	Returns to privileged EXEC mode.

Monitoring and Maintaining Port-Based Traffic Control

Command	Purpose
show interfaces [*interface-id*] **switchport**	Displays the administrative and operational status of all switching (nonrouting) ports or the specified port, including port blocking and port protection settings.
show storm-control [*interface-id*] [**broadcast** | **multicast** | **unicast**]	Displays storm control suppression levels set on all interfaces or the specified interface for the specified traffic type or for broadcast traffic if no traffic type is entered.
show port-security [**interface** *interface-id*]	Displays port security settings for the switch or for the specified interface, including the maximum allowed number of secure MAC addresses for each interface, the number of secure MAC addresses on the interface, the number of security violations that have occurred, and the violation mode.
show port-security [**interface** *interface-id*] **address**	Displays all secure MAC addresses configured on all switch interfaces or on a specified interface with aging information for each address.
show port-security interface *interface-id* **vlan**	Displays the number of secure MAC addresses configured per VLAN on the specified interface.
show storm-control [*interface-id*] [**broadcast** | **multicast** | **unicast**]	Displays the storm control suppression levels set on the interface for the specified traffic type. If you do not enter a traffic type, broadcast storm control settings are displayed.
show interfaces *interface-id*	Displays the interface configuration.
show interfaces *interface-id* **switchport**	Displays switch-port information.
show port-security	Displays port-security settings for an interface or for the switch.
show psp config {**arp** | **dhcp** | **igmp**}	Displays PSP configuration details for protocols.

Configuration Examples for Port-Based Traffic Control

Enabling Unicast Storm Control: Example

This example shows how to enable unicast storm control on a port with an 87-percent rising suppression level and a 65-percent falling suppression level:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/1
Switch(config-if)# storm-control unicast level 87 65
```

Enabling Broadcast Address Storm Control on a Port: Example

This example shows how to enable broadcast address storm control on a port to a level of 20 percent. When the broadcast traffic exceeds the configured level of 20 percent of the total available bandwidth of the port within the traffic-storm-control interval, the switch drops all broadcast traffic until the end of the traffic-storm-control interval:

```
Switch# configure terminal
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# storm-control broadcast level 20
```

Enabling Small-Frame Arrival Rate: Example

This example shows how to enable the small-frame arrival-rate feature, configure the port recovery time, and configure the threshold for error-disabling a port:

```
Switch# configure terminal
Switch# errdisable detect cause small-frame
Switch# errdisable recovery cause small-frame
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# small-frame violation rate 10000
Switch(config-if)# end
```

Configuring a Protected Port: Example

This example shows how to configure a port as a protected port:

```
Switch# configure terminal
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# switchport protected
Switch(config-if)# end
```

Blocking Flooding on a Port: Example

This example shows how to block unicast and Layer 2 multicast flooding on a port:

```
Switch# configure terminal
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# switchport block multicast
Switch(config-if)# switchport block unicast
Switch(config-if)# end
```

Configuring Port Security: Examples

This example shows how to enable port security on a port and to set the maximum number of secure addresses to 50. The violation mode is the default, no static secure MAC addresses are configured, and sticky learning is enabled:

```
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# switchport mode access
Switch(config-if)# switchport port-security
Switch(config-if)# switchport port-security maximum 50
Switch(config-if)# switchport port-security mac-address sticky
```

This example shows how to configure a static secure MAC address on VLAN 3 on a port:

```
Switch(config)# interface GigabitEthernet1/18
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport port-security
Switch(config-if)# switchport port-security mac-address 0000.02000.0004 vlan 3
```

This example shows how to enable sticky port security on a port, to manually configure MAC addresses for data VLAN and voice VLAN, and to set the total maximum number of secure addresses to 20 (10 for data VLAN and 10 for voice VLAN).

```
Switch(config)# interface FastEthernet1/1
Switch(config-if)# switchport access vlan 21
Switch(config-if)# switchport mode access
Switch(config-if)# switchport voice vlan 22
Switch(config-if)# switchport port-security
Switch(config-if)# switchport port-security maximum 20
Switch(config-if)# switchport port-security violation restrict
Switch(config-if)# switchport port-security mac-address sticky
Switch(config-if)# switchport port-security mac-address sticky 0000.0000.0002
Switch(config-if)# switchport port-security mac-address 0000.0000.0003
Switch(config-if)# switchport port-security mac-address sticky 0000.0000.0001 vlan voice
Switch(config-if)# switchport port-security mac-address 0000.0000.0004 vlan voice
Switch(config-if)# switchport port-security maximum 10 vlan access
Switch(config-if)# switchport port-security maximum 10 vlan voice
```

Configuring Port Security Aging: Examples

This example shows how to set the aging time as 2 hours for the secure addresses on a port:

```
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# switchport port-security aging time 120
```

This example shows how to set the aging time as 2 minutes for the inactivity aging type with aging enabled for the configured secure addresses on the interface:

```
Switch(config-if)# switchport port-security aging time 2
Switch(config-if)# switchport port-security aging type inactivity
Switch(config-if)# switchport port-security aging static
```

You can verify the previous commands by entering the **show port-security interface** *interface-id* privileged EXEC command.

Configuring Protocol Storm Protection: Example

This example shows how to configure protocol storm protection to drop incoming DHCP traffic on DHCP when it exceeds 35 packets per second:

```
Switch# configure terminal
Switch(config)# psp dhcp pps 35
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Configuring SPAN and RSPAN

Prerequisites for SPAN and RSPAN

- You must globally configure the **ip device tracking maximum** *limit-number* interface configuration command globally for IPSG for static hosts to work. If you only configure this command on a port without enabling IP device tracking globally or setting an IP device tracking maximum on that interface, IPSG with static hosts will reject all the IP traffic from that interface. This requirement also applies to IPSG with static hosts on a Layer 2 access port.

Information About SPAN and RSPAN

SPAN and RSPAN

You can analyze network traffic passing through ports or VLANs by using Switched Port Analyzer (SPAN) or Remote SPAN (RSPAN) to send a copy of the traffic to another port on the switch or on another switch that has been connected to a network analyzer or other monitoring or security device. SPAN copies (or mirrors) traffic received or sent (or both) on source ports or source VLANs to a destination port for analysis. SPAN does not affect the switching of network traffic on the source ports or VLANs. You must dedicate the destination port for SPAN use. Except for traffic that is required for the SPAN or RSPAN session, destination ports do not receive or forward traffic.

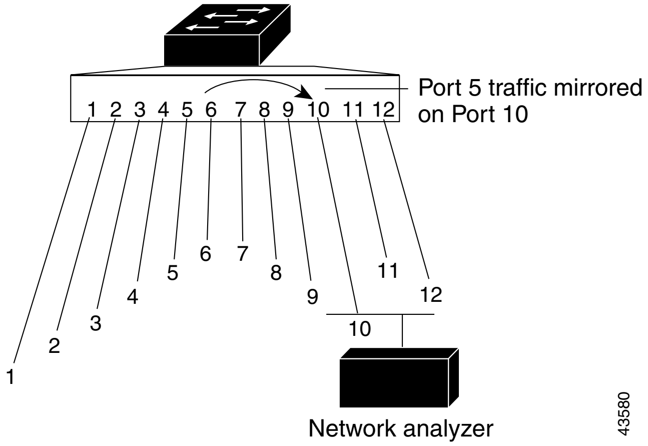
Only traffic that enters or leaves source ports or traffic that enters or leaves source VLANs can be monitored by using SPAN; traffic routed to a source VLAN cannot be monitored. For example, if incoming traffic is being monitored, traffic that gets routed from another VLAN to the source VLAN cannot be monitored; however, traffic that is received on the source VLAN and routed to another VLAN can be monitored.

You can use the SPAN or RSPAN destination port to inject traffic from a network security device. For example, if you connect a Cisco Intrusion Detection System (IDS) sensor appliance to a destination port, the IDS device can send TCP reset packets to close down the TCP session of a suspected attacker.

Local SPAN

Local SPAN supports a SPAN session entirely within one switch; all source ports or source VLANs and destination ports are in the same switch. Local SPAN copies traffic from one or more source ports in any VLAN or from one or more VLANs to a destination port for analysis. For example, in Figure 69 on page 492, all traffic on port 5 (the source port) is mirrored to port 10 (the destination port). A network analyzer on port 10 receives all network traffic from port 5 without being physically attached to port 5.

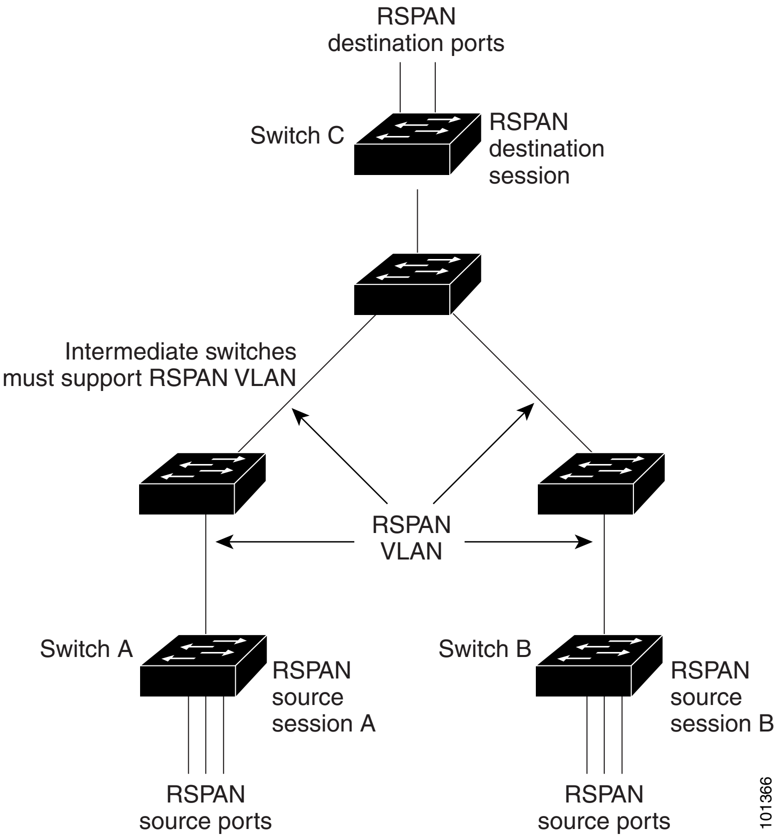
Figure 69 Example of Local SPAN Configuration on a Single Switch



Remote SPAN

RSPAN supports source ports, source VLANs, and destination ports on different switches, enabling remote monitoring of multiple switches across your network. Figure 70 on page 493 shows source ports on Switch A and Switch B. 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 RSPAN traffic from the source ports or VLANs is copied into the RSPAN VLAN and forwarded over trunk ports carrying the RSPAN VLAN to a destination session monitoring the RSPAN VLAN. Each RSPAN source switch must have either ports or VLANs as RSPAN sources. The destination is always a physical port, as shown on Switch C in the figure.

Figure 70 Example of RSPAN Configuration



SPAN Sessions

SPAN sessions (local or remote) allow you to monitor traffic on one or more ports, or one or more VLANs, and send the monitored traffic to one or more destination ports.

A local SPAN session is an association of a destination port with source ports or source VLANs, all on a single network device. Local SPAN does not have separate source and destination sessions. Local SPAN sessions gather a set of ingress and egress packets specified by the user and form them into a stream of SPAN data, which is directed to the destination port.

RSPAN consists of at least one RSPAN source session, an RSPAN VLAN, and at least one RSPAN destination session. You separately configure RSPAN source sessions and RSPAN destination sessions on different network devices. To configure an RSPAN source session on a device, you associate a set of source ports or source VLANs with an RSPAN VLAN. The output of this session is the stream of SPAN packets that are sent to the RSPAN VLAN. To configure an RSPAN destination session on another device, you associate the destination port with the RSPAN VLAN. The destination session collects all RSPAN VLAN traffic and sends it out the RSPAN destination port.

An RSPAN source session is very similar to a local SPAN session, except for where the packet stream is directed. In an RSPAN source session, SPAN packets are relabeled with the RSPAN VLAN ID and directed over normal trunk ports to the destination switch.

An RSPAN destination session takes all packets received on the RSPAN VLAN, strips off the VLAN tagging, and presents them on the destination port. Its purpose is to present a copy of all RSPAN VLAN packets (except Layer 2 control packets) to the user for analysis.

There can be more than one source session and more than one destination session active in the same RSPAN VLAN. There can also be intermediate switches separating the RSPAN source and destination sessions. These switches need not be capable of running RSPAN, but they must respond to the requirements of the RSPAN VLAN (see RSPAN VLAN, page 497).

Traffic monitoring in a SPAN session has these restrictions:

- Sources can be ports or VLANs, but you cannot mix source ports and source VLANs in the same session.
- The switch supports up to 4 source sessions (local SPAN and RSPAN source sessions). You can run both a local SPAN and an RSPAN source session in the same switch. The switch supports a total of 68 source and RSPAN destination sessions.
- You can have multiple destination ports in a SPAN session, but no more than 64 destination ports.
- You can configure two separate SPAN or RSPAN source sessions with separate or overlapping sets of SPAN source ports and VLANs. Both switched and routed ports can be configured as SPAN sources and destinations.
- SPAN sessions do not interfere with the normal operation of the switch. However, an oversubscribed SPAN destination, for example, a 10-Mb/s port monitoring a 100-Mb/s port, can result in dropped or lost packets.
- When RSPAN is enabled, each packet being monitored is transmitted twice, once as normal traffic and once as a monitored packet. Therefore monitoring a large number of ports or VLANs could potentially generate large amounts of network traffic.
- You can configure SPAN sessions on disabled ports; however, a SPAN session does not become active unless you enable the destination port and at least one source port or VLAN for that session.
- The switch does not support a combination of local SPAN and RSPAN in a single session. That is, an RSPAN source session cannot have a local destination port, an RSPAN destination session cannot have a local source port, and an RSPAN destination session and an RSPAN source session that are using the same RSPAN VLAN cannot run on the same switch.

Monitored Traffic Types for SPAN Sessions

- Receive (Rx) SPAN—The goal of receive (or ingress) SPAN is to monitor as much as possible all the packets received by the source interface or VLAN before any modification or processing is performed by the switch. A copy of each packet received by the source is sent to the destination port for that SPAN session.

  Packets that are modified because of routing or quality of service (QoS)—for example, modified Differentiated Services Code Point (DSCP)—are copied before modification.

  Features that can cause a packet to be dropped during receive processing have no effect on ingress SPAN; the destination port receives a copy of the packet even if the actual incoming packet is dropped. These features include IP standard and extended input access control lists (ACLs), ingress QoS policing, VLAN ACLs, and egress QoS policing.

- Transmit (Tx) SPAN—The goal of transmit (or egress) SPAN is to monitor as much as possible all the packets sent by the source interface after all modification and processing is performed by the switch. A copy of each packet sent by the source is sent to the destination port for that SPAN session. The copy is provided after the packet is modified.

  Packets that are modified because of routing—for example, with modified time-to-live (TTL), MAC-address, or QoS values—are duplicated (with the modifications) at the destination port.

  Features that can cause a packet to be dropped during transmit processing also affect the duplicated copy for SPAN. These features include IP standard and extended output ACLs and egress QoS policing.

- Both—In a SPAN session, you can also monitor a port or VLAN for both received and sent packets. This is the default.

The default configuration for local SPAN session ports is to send all packets untagged. SPAN also does not normally monitor bridge protocol data unit (BPDU) packets and Layer 2 protocols, such as Cisco Discovery Protocol (CDP), VLAN Trunk Protocol (VTP), Dynamic Trunking Protocol (DTP), Spanning Tree Protocol (STP), and Port Aggregation Protocol (PAgP). However, when you enter the **encapsulation replicate** keywords when configuring a destination port, these changes occur:

- Packets are sent on the destination port with the same encapsulation—untagged or IEEE 802.1Q—that they had on the source port.
- Packets of all types, including BPDU and Layer 2 protocol packets, are monitored.

Therefore, a local SPAN session with encapsulation replicate enabled can have a mixture of untagged and IEEE 802.1Q tagged packets appear on the destination port.

Switch congestion can cause packets to be dropped at ingress source ports, egress source ports, or SPAN destination ports. In general, these characteristics are independent of one another. For example:

- A packet might be forwarded normally but dropped from monitoring due to an oversubscribed SPAN destination port.
- An ingress packet might be dropped from normal forwarding, but still appear on the SPAN destination port.
- An egress packet dropped because of switch congestion is also dropped from egress SPAN.

In some SPAN configurations, multiple copies of the same source packet are sent to the SPAN destination port. For example, a bidirectional (both Rx and Tx) SPAN session is configured for the Rx monitor on port A and Tx monitor on port B. If a packet enters the switch through port A and is switched to port B, both incoming and outgoing packets are sent to the destination port. Both packets are the same (unless a Layer-3 rewrite occurs, in which case the packets are different because of the packet modification).

Source Ports

A source port (also called a *monitored port*) is a switched or routed port that you monitor for network traffic analysis. In a local SPAN session or RSPAN source session, you can monitor source ports or VLANs for traffic in one or both directions. The switch supports any number of source ports (up to the maximum number of available ports on the switch) and any number of source VLANs (up to the maximum number of VLANs supported). However, the switch supports a maximum of two sessions (local or RSPAN) with source ports or VLANs, and you cannot mix ports and VLANs in a single session.

A source port has these characteristics:

- It can be monitored in multiple SPAN sessions.
- Each source port can be configured with a direction (ingress, egress, or both) to monitor.
- It can be any port type (for example, EtherChannel, Gigabit Ethernet, and so forth).
- For EtherChannel sources, you can monitor traffic for the entire EtherChannel or individually on a physical port as it participates in the port channel.
- It can be an access port, trunk port, routed port, or voice VLAN port.
- It cannot be a destination port.
- Source ports can be in the same or different VLANs.
- You can monitor multiple source ports in a single session.

Source VLANs

VLAN-based SPAN (VSPAN) is the monitoring of the network traffic in one or more VLANs. The SPAN or RSPAN source interface in VSPAN is a VLAN ID, and traffic is monitored on all the ports for that VLAN.

VSPAN has these characteristics:

- All active ports in the source VLAN are included as source ports and can be monitored in either or both directions.
- On a given port, only traffic on the monitored VLAN is sent to the destination port.
- If a destination port belongs to a source VLAN, it is excluded from the source list and is not monitored.
- If ports are added to or removed from the source VLANs, the traffic on the source VLAN received by those ports is added to or removed from the sources being monitored.
- You cannot use filter VLANs in the same session with VLAN sources.
- You can monitor only Ethernet VLANs.

VLAN Filtering

When you monitor a trunk port as a source port, by default, all VLANs active on the trunk are monitored. You can limit SPAN traffic monitoring on trunk source ports to specific VLANs by using VLAN filtering.

- VLAN filtering applies only to trunk ports or to voice VLAN ports.
- VLAN filtering applies only to port-based sessions and is not allowed in sessions with VLAN sources.
- When a VLAN filter list is specified, only those VLANs in the list are monitored on trunk ports or on voice VLAN access ports.
- SPAN traffic coming from other port types is not affected by VLAN filtering; that is, all VLANs are allowed on other ports.
- VLAN filtering affects only traffic forwarded to the destination SPAN port and does not affect the switching of normal traffic.

Destination Port

Each local SPAN session or RSPAN destination session must have a destination port (also called a *monitoring port*) that receives a copy of traffic from the source ports or VLANs and sends the SPAN packets to the user, usually a network analyzer.

A destination port has these characteristics:

- For a local SPAN session, the destination port must reside on the same switch as the source port. For an RSPAN session, it is located on the switch containing the RSPAN destination session. There is no destination port on a switch running only an RSPAN source session.
- When a port is configured as a SPAN destination port, the configuration overwrites the original port configuration. When the SPAN destination configuration is removed, the port reverts to its previous configuration. If a configuration change is made to the port while it is acting as a SPAN destination port, the change does not take effect until the SPAN destination configuration had been removed.
- If the port was in an EtherChannel group, it is removed from the group while it is a destination port. If it was a routed port, it is no longer a routed port.
- It can be any Ethernet physical port.
- It cannot be a secure port.

- It cannot be a source port.
- It cannot be an EtherChannel group or a VLAN.
- It can participate in only one SPAN session at a time (a destination port in one SPAN session cannot be a destination port for a second SPAN session).
- When it is active, incoming traffic is disabled. The port does not transmit any traffic except that required for the SPAN session. Incoming traffic is never learned or forwarded on a destination port.
- If ingress traffic forwarding is enabled for a network security device, the destination port forwards traffic at Layer 2.
- It does not participate in any of the Layer 2 protocols (STP, VTP, CDP, DTP, PagP).
- A destination port that belongs to a source VLAN of any SPAN session is excluded from the source list and is not monitored.
- The maximum number of destination ports in a switch is 64.

Local SPAN and RSPAN destination ports behave differently regarding VLAN tagging and encapsulation:

- For local SPAN, if the **encapsulation replicate** keywords are specified for the destination port, these packets appear with the original encapsulation (untaggedor IEEE 802.1Q). If these keywords are not specified, packets appear in the untagged format. Therefore, the output of a local SPAN session with **encapsulation replicate** enabled can contain a mixture of untagged or IEEE 802.1Q-tagged packets.
- For RSPAN, the original VLAN ID is lost because it is overwritten by the RSPAN VLAN identification. Therefore, all packets appear on the destination port as untagged.

RSPAN VLAN

The RSPAN VLAN carries SPAN traffic between RSPAN source and destination sessions. It has these special characteristics:

- All traffic in the RSPAN VLAN is always flooded.
- No MAC address learning occurs on the RSPAN VLAN.
- RSPAN VLAN traffic only flows on trunk ports.
- RSPAN VLANs must be configured in VLAN configuration mode by using the **remote-span** VLAN configuration mode command.
- STP can run on RSPAN VLAN trunks but not on SPAN destination ports.
- An RSPAN VLAN cannot be a private-VLAN primary or secondary VLAN.

For VLANs 1 to 1005 that are visible to VLAN Trunking Protocol (VTP), the VLAN ID and its associated RSPAN characteristic are propagated by VTP. If you assign an RSPAN VLAN ID in the extended VLAN range (1006 to 4096), you must manually configure all intermediate switches.

It is normal to have multiple RSPAN VLANs in a network at the same time with each RSPAN VLAN defining a network-wide RSPAN session. That is, multiple RSPAN source sessions anywhere in the network can contribute packets to the RSPAN session. It is also possible to have multiple RSPAN destination sessions throughout the network, monitoring the same RSPAN VLAN and presenting traffic to the user. The RSPAN VLAN ID separates the sessions.

Spanned Traffic Timestamping (IE 5000 only)

The Spanned Traffic Timestamping feature for IE 5000 switches provides ingress timestamping (timestamping of received packets) for a single SPAN/RSPAN session. Egress timestamping (timestamping of transmitted packets) is not supported. Spanned Traffic Timestamping is available in Cisco IOS Release 15.2(7)E1a and later.

Spanned Traffic Timestamping is implemented through the switch hardware, which is synchronized to the PTP Grandmaster Clock through the IEEE Std 1588-2008 PTP protocol. IE 5000 switch network interfaces connected to sensor/end devices are configured as SPAN session source interfaces with timestamping enabled. This configuration results in all ingress packets from sensor/end devices to be timestamped at the IE 5000 timestamping switch network interface physical layer. The RSPAN VLAN is configured as the SPAN session destination. See Spanned Traffic Timestamping Configuration Guidelines, page 500, Creating a Local SPAN Session with Timestamp, page 510, and Creating an RSPAN Source Session with Timestamp, page 512.

SPAN and RSPAN Interaction with Other Features

- Routing—SPAN does not monitor routed traffic. VSPAN only monitors traffic that enters or exits the switch, not traffic that is routed between VLANs. For example, if a VLAN is being Rx-monitored and the switch routes traffic from another VLAN to the monitored VLAN, that traffic is not monitored and not received on the SPAN destination port.
- STP—A destination port does not participate in STP while its SPAN or RSPAN session is active. The destination port can participate in STP after the SPAN or RSPAN session is disabled. On a source port, SPAN does not affect the STP status. STP can be active on trunk ports carrying an RSPAN VLAN.
- CDP—A SPAN destination port does not participate in CDP while the SPAN session is active. After the SPAN session is disabled, the port again participates in CDP.
- VTP—You can use VTP to prune an RSPAN VLAN between switches.
- VLAN and trunking—You can modify VLAN membership or trunk settings for source or destination ports at any time. However, changes in VLAN membership or trunk settings for a destination port do not take effect until you remove the SPAN destination configuration. Changes in VLAN membership or trunk settings for a source port immediately take effect, and the respective SPAN sessions automatically adjust accordingly.
- EtherChannel—You can configure an EtherChannel group as a source port but not as a SPAN destination port. When a group is configured as a SPAN source, the entire group is monitored.

  If a physical port is added to a monitored EtherChannel group, the new port is added to the SPAN source port list. If a port is removed from a monitored EtherChannel group, it is automatically removed from the source port list.

  A physical port that belongs to an EtherChannel group can be configured as a SPAN source port and still be a part of the EtherChannel. In this case, data from the physical port is monitored as it participates in the EtherChannel. However, if a physical port that belongs to an EtherChannel group is configured as a SPAN destination, it is removed from the group. After the port is removed from the SPAN session, it rejoins the EtherChannel group. Ports removed from an EtherChannel group remain members of the group, but they are in the *inactive* or *suspended* state.

  If a physical port that belongs to an EtherChannel group is a destination port and the EtherChannel group is a source, the port is removed from the EtherChannel group and from the list of monitored ports.
- Multicast traffic can be monitored. For egress and ingress port monitoring, only a single unedited packet is sent to the SPAN destination port. It does not reflect the number of times the multicast packet is sent.
- A private-VLAN port cannot be a SPAN destination port.
- A secure port cannot be a SPAN destination port.

  For SPAN sessions, do not enable port security on ports with monitored egress when ingress forwarding is enabled on the destination port. For RSPAN source sessions, do not enable port security on any ports with monitored egress.

- An IEEE 802.1x port can be a SPAN source port. You can enable IEEE 802.1x on a port that is a SPAN destination port; however, IEEE 802.1x is disabled until the port is removed as a SPAN destination.

  For SPAN sessions, do not enable IEEE 802.1x on ports with monitored egress when ingress forwarding is enabled on the destination port. For RSPAN source sessions, do not enable IEEE 802.1x on any ports that are egress monitored.

Local SPAN Configuration Guidelines

- For SPAN sources, you can monitor traffic for a single port or VLAN or a series or range of ports or VLANs for each session. You cannot mix source ports and source VLANs within a single SPAN session.
- The destination port cannot be a source port; a source port cannot be a destination port.
- You cannot have two SPAN sessions using the same destination port.
- When you configure a switch port as a SPAN destination port, it is no longer a normal switch port; only monitored traffic passes through the SPAN destination port.
- Entering SPAN configuration commands does not remove previously configured SPAN parameters. You must enter the **no monitor session** {*session_number* | **all** | **local** | **remote**} global configuration command to delete configured SPAN parameters.
- For local SPAN, outgoing packets through the SPAN destination port carry the original encapsulation headers—untagged or IEEE 802.1Q—if the **encapsulation replicate** keywords are specified. If the keywords are not specified, the packets are sent in native form. For RSPAN destination ports, outgoing packets are not tagged.
- You can configure a disabled port to be a source or destination port, but the SPAN function does not start until the destination port and at least one source port or source VLAN are enabled.
- You can limit SPAN traffic to specific VLANs by using the **filter vlan** keyword. If a trunk port is being monitored, only traffic on the VLANs specified with this keyword is monitored. By default, all VLANs are monitored on a trunk port.
- You cannot mix source VLANs and filter VLANs within a single SPAN session.

RSPAN Configuration Guidelines

- All the items in the Local SPAN Configuration Guidelines, page 499 apply to RSPAN.
- Because RSPAN VLANs have special properties, you should reserve a few VLANs across your network for use as RSPAN VLANs; do not assign access ports to these VLANs.
- You can apply an output ACL to RSPAN traffic to selectively filter or monitor specific packets. Specify these ACLs on the RSPAN VLAN in the RSPAN source switches.
- For RSPAN configuration, you can distribute the source ports and the destination ports across multiple switches in your network.
- RSPAN does not support BPDU packet monitoring or other Layer 2 switch protocols.
- The RSPAN VLAN is configured only on trunk ports and not on access ports. To avoid unwanted traffic in RSPAN VLANs, make sure that the VLAN remote-span feature is supported in all the participating switches.
- Access ports (including voice VLAN ports) on the RSPAN VLAN are put in the inactive state.

- RSPAN VLANs are included as sources for port-based RSPAN sessions when source trunk ports have active RSPAN VLANs. RSPAN VLANs can also be sources in SPAN sessions. However, since the switch does not monitor spanned traffic, it does not support egress spanning of packets on any RSPAN VLAN identified as the destination of an RSPAN source session on the switch.
- You can configure any VLAN as an RSPAN VLAN as long as these conditions are met:
  - The same RSPAN VLAN is used for an RSPAN session in all the switches.
  - All participating switches support RSPAN.
- We recommend that you configure an RSPAN VLAN before you configure an RSPAN source or a destination session.
- If you enable VTP and VTP pruning, RSPAN traffic is pruned in the trunks to prevent the unwanted flooding of RSPAN traffic across the network for VLAN IDs that are lower than 1005.

Spanned Traffic Timestamping Configuration Guidelines

- The spanned traffic timestamping feature supports only Ingress timestamping (timestamping of received packets). Egress timestamping (timestamping of transmitted packets) is not supported.
- Spanned traffic timestamping is supported on all IE 5000 downlink (both copper and fiber) interfaces in 100M and 1G interface speeds.
- Ingress timestamping is disabled by default and is enabled when the timestamping option is specified in SPAN session CLI configuration.
- SPAN timestamping is supported only for a single SPAN session. SPAN timestamp configuration CLI is rejected if you attempt to enable timestamping for multiple SPAN sessions.
- SPAN sources (**monitor session** <*session number*> **source**) configured in a SPAN session with timestamp option enabled cannot be used in other SPAN sessions. SPAN timestamp configuration CLI is rejected if you attempt to enable timestamping on SPAN session where sources (**monitor session** <*session number*> **source**) are already configured in other SPAN sessions.
- SPAN source configuration CLI is rejected if the SPAN session has timestamping enabled and the SPAN sources are already configured in other SPAN sessions.
- All interface configuration (such as interface access/trunk configuration, specifying VLANs for RSPAN traffic, and so on) should be completed before enabling the SPAN timestamp configuration.
- The SPAN timestamp configuration option is not supported in a SPAN session that has the SPAN source as Port-channel, RSPAN VLAN, or uplink interfaces.
- The SPAN timestamp configuration option is allowed only if the SPAN session source and destination are configured first.
- Spanned Traffic Timestamping is supported in lanbase and higher license levels.
- With ingress timestamping enabled, the maximum rate that can be supported is (N / (N + 18)) * 100%, where N is the original packet size. For 64-byte packets, operation is at 78% of the line rate. For 1500-byte packets, operation is at 98.8% of the line rate.
- Timestamp is added only to the following packets: Ethernet II, IEEE 802.3 with SNAP, IEEE 802.3 CSMA/CD, IPv4, IPv6 and UDP.

Default SPAN and RSPAN Settings

Feature	Default Setting
SPAN state (SPAN and RSPAN)	Disabled.
Source port traffic to monitor	Both received and sent traffic (**both**).
Encapsulation type (destination port)	Native form (untagged packets).
Ingress forwarding (destination port)	Disabled,
VLAN filtering	On a trunk interface used as a source port, all VLANs are monitored.
RSPAN VLANs	None configured.

How to Configure SPAN and RSPAN

Creating a Local SPAN Session

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**no monitor session** {*session_number* | **all** | **local** | **remote**}	Removes any existing SPAN configuration for the session. *session_number*–The range is 1 to 68. Specify **all** to remove all SPAN sessions, **local** to remove all local sessions, or **remote** to remove all remote SPAN sessions.

	Command	Purpose
3.	**monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**]	Specifies the SPAN session and the source port (monitored port). *session_number*–The range is 1 to 68. *interface-id*–Specifies the source port or source VLAN to monitor. ■ source *interface-id*–Specifies the source port to monitor. Valid interfaces include physical interfaces and port-channel logical interfaces (**port-channel** *port-channel-number*). Valid port-channel numbers are 1 to 10. ■ *vlan-id*–Specifies the source VLAN to monitor. The range is 1 to 4096 (excluding the RSPAN VLAN). **Note:** A single session can include multiple sources (ports or VLANs), defined in a series of commands, but you cannot combine source ports and source VLANs in one session. (Optional) [**,** | **-**] Specify a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. (Optional) Specify the direction of traffic to monitor. If you do not specify a traffic direction, the SPAN monitors both sent and received traffic. ■ **both**–Monitors both received and sent traffic. This is the default. ■ **rx**–Monitors received traffic. ■ **tx**–Monitors sent traffic. **Note:** You can use the **monitor session** *session_number* **source** command multiple times to configure multiple source ports.
4.	**monitor session** *session_number* **destination** {**interface** *interface-id* [, | -] [**encapsulation replicate**]}	Specifies the SPAN session and the destination port (monitoring port). *session_number*–Specifies the session number entered in step 3. **Note:** For local SPAN, you must use the same session number for the source and destination interfaces. ■ *interface-id*–Specifies the destination port. The destination interface must be a physical port; it cannot be an EtherChannel, and it cannot be a VLAN. ■ (Optional) [**,** | **-**]–Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. ■ (Optional) **encapsulation replicate**–Specifies that the destination interface replicates the source interface encapsulation method. If not selected, the default is to send packets in native form (untagged). **Note:** You can use **monitor session** *session_number* **destination** command multiple times to configure multiple destination ports.
5.	**end**	Returns to privileged EXEC mode.

Creating a Local SPAN Session and Configuring Incoming Traffic

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**no monitor session** {*session_number* | **all** | **local** | **remote**}	Removes any existing SPAN configuration for the session.
3.	**monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**]	Specifies the SPAN session and the source port (monitored port).
4.	**monitor session** *session_number* **destination** {**interface** *interface-id* [, | -] [**encapsulation replicate**] [**ingress** {**dot1q vlan** *vlan-id* | **untagged vlan** *vlan-id* | **vlan** *vlan-id*}]}	Specifies the SPAN session, the destination port, the packet encapsulation, and the ingress VLAN and encapsulation. *session_number*–Specifies the session number entered in Step 3. *interface-id*–Specifies the destination port. The destination interface must be a physical port; it cannot be an EtherChannel, and it cannot be a VLAN. (Optional) [**,** | **-**]–Specifies a series or range of interfaces. Enter a space before and after the comma or hyphen. (Optional) **encapsulation replicate**–Specifies that the destination interface replicates the source interface encapsulation method. If not selected, the default is to send packets in native form (untagged). **ingress**–Enables forwarding of incoming traffic on the destination port and specifies the encapsulation type: ■ **dot1q vlan** *vlan-id*–Accepts incoming packets with IEEE 802.1Q encapsulation with the specified VLAN as the default VLAN. ■ **untagged vlan** *vlan-id* or **vlan** *vlan-id*–Accepts incoming packets with untagged encapsulation type with the specified VLAN as the default VLAN.
5.	**end**	Returns to privileged EXEC mode.

Specifying VLANs to Filter

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**no monitor session** {*session_number* | **all** | **local** | **remote**}	Removes any existing SPAN configuration for the session. *session_number*–The range is 1 to 68. **all**–Removes all SPAN sessions. **local**–Removes all local sessions. **remote–**Removes all remote SPAN sessions.
3.	**monitor session** *session_number* **source interface** *interface-id*	Specifies the characteristics of the source port (monitored port) and SPAN session. *session_number*–The range is 1 to 68. *interface-id*–Specifies the source port to monitor. The interface specified must already be configured as a trunk port.
4.	**monitor session** *session_number* **filter vlan** *vlan-id* [**,** | **-**]	Limits the SPAN source traffic to specific VLANs. *session_number*–Enters the session number specified in Step 3. *vlan-id*–The range is 1 to 4096. (Optional) Use a comma (**,**) to specify a series of VLANs, or use a hyphen (**-**) to specify a range of VLANs. Enter a space before and after the comma; enter a space before and after the hyphen.
5.	**monitor session** *session_number* **destination** {**interface** *interface-id* [, | -] [**encapsulation replicate**]}	Specifies the SPAN session and the destination port (monitoring port). *session_number*–Specifies the session number entered in Step 3. *interface-id*–Specifies the destination port. The destination interface must be a physical port; it cannot be an EtherChannel, and it cannot be a VLAN. (Optional) [**,** | **-**]–Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. (Optional) **encapsulation replicate**–Specifies that the destination interface replicates the source interface encapsulation method. If not selected, the default is to send packets in native form (untagged).
6.	**end**	Returns to privileged EXEC mode.

Configuring a VLAN as an RSPAN VLAN

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**vlan** *vlan-id*	Enters a VLAN ID to create a VLAN, or enters the VLAN ID of an existing VLAN, and enter VLAN configuration mode. The range is 2 to 1001 and 1006 to 4096. The RSPAN VLAN cannot be VLAN 1 (the default VLAN) or VLAN IDs 1002 through 1005 (reserved for Token Ring and FDDI VLANs).
3.	**remote-span**	Configures the VLAN as an RSPAN VLAN.
4.	**end**	Returns to privileged EXEC mode.
5.	**copy running-config startup-config**	(Optional) Saves the configuration in the configuration file.

Creating an RSPAN Source Session

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**no monitor session** {*session_number* | **all** | **local** | **remote**}	Removes any existing RSPAN configuration for the session. *session_number*—The range is 1 to 68. **all**—Removes all RSPAN sessions **local–**Removes all local sessions **remote–**Removes all remote SPAN sessions.
3.	**monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**]	Specifies the RSPAN session and the source port (monitored port). *session_number*—The range is 1 to 68. Enter a source port or source VLAN for the RSPAN session: ■ *interface-id*—Specifies the source port to monitor. Valid interfaces include physical interfaces and port-channel logical interfaces (**port-channel** *port-channel-number*). Valid port-channel numbers are 1 to 10. ■ *vlan-id*—Specifies the source VLAN to monitor. The range is 1 to 4096 (excluding the RSPAN VLAN). A single session can include multiple sources (ports or VLANs), defined in a series of commands, but you cannot combine source ports and source VLANs in one session. (Optional) [**,** | **-**]—Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. (Optional) Specify the direction of traffic to monitor. If you do not specify a traffic direction, the source interface sends both sent and received traffic. ■ **both**—Monitors both received and sent traffic. ■ **rx**—Monitors received traffic. ■ **tx**—Monitors sent traffic.
4.	**monitor session** *session_number* **destination remote vlan** *vlan-id*	Specifies the RSPAN session and the destination RSPAN VLAN. *session_number*—Enters the number defined in Step 3. *vlan-id*—Specifies the source RSPAN VLAN to monitor.
5.	**end**	Returns to privileged EXEC mode.
6.	**show monitor** [**session** *session_number*] **show running-config**	Verifies the configuration.
7.	**copy running-config startup-config**	(Optional) Saves the configuration in the configuration file.

Creating an RSPAN Destination Session

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**vlan** *vlan-id*	Enters the VLAN ID of the RSPAN VLAN created from the source switch, and enters VLAN configuration mode. If both switches are participating in VTP and the RSPAN VLAN ID is from 2 to 1005, Steps 2 through 4 are not required because the RSPAN VLAN ID is propagated through the VTP network.
3.	**remote-span**	Identifies the VLAN as the RSPAN VLAN.
4.	**exit**	Returns to global configuration mode.
5.	**no monitor session** {*session_number* | **all** | **local** | **remote**}	Removes any existing RSPAN configuration for the session. *session_number*–The range is 1 to 68. **all–**Removes all RSPAN sessions **local**–Removes all local sessions **remote–**Removes all remote SPAN sessions.
6.	**monitor session** *session_number* **source remote vlan** *vlan-id*	Specifies the RSPAN session and the source RSPAN VLAN. *session_number*–The range is 1 to 68. *vlan-id*–Specifies the source RSPAN VLAN to monitor.
7.	**monitor session** *session_number* **destination interface** *interface-id*	Specifies the RSPAN session and the destination interface. *session_number*–Enters the number defined in Step 6. In an RSPAN destination session, you must use the same session number for the source RSPAN VLAN and the destination port. *interface-id*–Specifies the destination interface. The destination interface must be a physical interface. Though visible in the command-line help string, **encapsulation replicate** is not supported for RSPAN. The original VLAN ID is overwritten by the RSPAN VLAN ID, and all packets appear on the destination port as untagged.
8.	**end**	Returns to privileged EXEC mode.

Creating an RSPAN Destination Session and Configuring Incoming Traffic

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**no monitor session** {*session_number* | **all** | **local** | **remote**}	Removes any existing SPAN configuration for the session.
3.	**monitor session** *session_number* **source remote vlan** *vlan-id*	Specifies the RSPAN session and the source RSPAN VLAN. *session_number*–The range is 1 to 68. *vlan-id*–Specifies the source RSPAN VLAN to monitor.
4.	**monitor session** *session_number* **destination** {**interface** *interface-id* [, | -] [**ingress** {**dot1q vlan** *vlan-id* **|** **untagged vlan** *vlan-id* **|** **vlan** *vlan-id*}]}	Specifies the SPAN session, the destination port, the packet encapsulation, and the incoming VLAN and encapsulation. *session_number*–Enters the number defined in Step 4. In an RSPAN destination session, you must use the same session number for the source RSPAN VLAN and the destination port. *interface-id*–Specifies the destination interface. The destination interface must be a physical interface. Though visible in the command-line help string, **encapsulation replicate** is not supported for RSPAN. The original VLAN ID is overwritten by the RSPAN VLAN ID, and all packets appear on the destination port as untagged. (Optional) [**,** **|** **-**]–Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. Enter **ingress** with additional keywords to enable forwarding of incoming traffic on the destination port and to specify the encapsulation type: ■ **dot1q vlan** *vlan-id*–Forwards incoming packets with IEEE 802.1Q encapsulation with the specified VLAN as the default VLAN. ■ **untagged vlan** *vlan-id* or **vlan** *vlan-id*–Forwards incoming packets with untagged encapsulation type with the specified VLAN as the default VLAN.
5.	**end**	Returns to privileged EXEC mode.

Specifying VLANs to Filter

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**no monitor session** {*session_number* | **all** | **local** | **remote**}	Removes any existing SPAN configuration for the session. *session_number*–The range is 1 to 68. **all–**Removes all SPAN sessions. **local–**Removes all local sessions. **remote**–Removes all remote SPAN sessions.
3.	**monitor session** *session_number* **source interface** *interface-id*	Specifies the characteristics of the source port (monitored port) and SPAN session. *session_number*–The range is 1 to 68. *interface-id*–Specifies the source port to monitor. The interface specified must already be configured as a trunk port.
4.	**monitor session** *session_number* **filter vlan** *vlan-id* [**,** | **-**]	Limits the SPAN source traffic to specific VLANs. *session_number*–Enters the session number specified in step 3. *vlan-id*–The range is 1 to 4096. (Optional) Use a comma (**,**) to specify a series of VLANs or use a hyphen (**-**) to specify a range of VLANs. Enter a space before and after the comma; enter a space before and after the hyphen.
5.	**monitor session** *session_number* **destination remote vlan** *vlan-id*	Specifies the RSPAN session and the destination remote VLAN (RSPAN VLAN). *session_number*–Enter the session number specified in step 3. *vlan-id*–Specifies the RSPAN VLAN to carry the monitored traffic to the destination port.
6.	**end**	Returns to privileged EXEC mode.

Creating a Local SPAN Session with Timestamp

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**license right-to-use activate STT**	Activates the Spanned Traffic Timestamping feature.
3.	**monitor** **session** *session_number* **source interface** *interface-id* [**,** | **-**] [**rx**]	Specifies the SPAN session and the source port (monitored port). ■ *session_number*–The range is 1 to 68. ■ *interface-id*–Specifies the source port or source VLAN to monitor. ■ source *interface-id* –Specifies the source port to monitor. Valid interfaces include physical interfaces. **Note:** A single session can include multiple sources (ports), defined in a series of commands. ■ (Optional) [**,** | **-**]–Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. ■ (Optional) Specifies the direction of traffic to monitor. – **rx**–*Monitors received traffic.* **Note:** You can use the **monitor session** *session_number* **source** command multiple times to configure multiple source ports.

	Command	Purpose
4.	**monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**]}	Specifies the SPAN session and the destination port (monitoring port). ■ *session_number*–Specifies the session number entered in step 3. **Note:** For local SPAN, you must use the same session number for the source and destination interfaces. ■ *interface-id*–Specifies the destination port. The destination interface must be a physical port; it cannot be an EtherChannel, and it cannot be a VLAN. ■ (Optional) [**,** | **-**]–Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. ■ (Optional) **encapsulation replicate**–Specifies that the destination interface replicates the source interface encapsulation method. If not selected, the default is to send packets in native form (untagged). **Note:** You can use **monitor session** *session_number* **destination** command multiple times to configure multiple destination ports.
5.	**monitor session** *session_number* **timestamp**	Enables the timestamp in the session traffic.
6.	**end**	Returns to privileged EXEC mode.

Creating an RSPAN Source Session with Timestamp

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**license right-to-use activate STT**	Activates the Spanned Traffic Timestamping feature.
3.	**monitor** **session** *session_number* **source interface** *interface-id* [**,** | **-**] [**rx**]	Specifies the RSPAN session and the source port (monitored port). ■ *session_number*—The range is 1 to 68. Enter a source port or source VLAN for the RSPAN session: ■ *interface-id*—Specifies the source port or source VLAN to monitor. ■ source *interface-id* —Specifies the source port to monitor. Valid interfaces include physical interfaces. **Note:** A single session can include multiple sources (ports), defined in a series of commands. ■ (Optional) [**,** | **-**]—Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. ■ (Optional) Specifies the direction of traffic to monitor. – **rx**—*Monitors received traffic.* **Note:** You can use the **monitor session** *session_number* **source** command multiple times to configure multiple source ports.
4.	**monitor session** *session_number* **destination remote vlan** *vlan-id*	Specifies the RSPAN session and the destination RSPAN VLAN. *vlan-id*—Specifies the source RSPAN VLAN to monitor.
5.	**monitor session** *session_number* **timestamp**	Enables the timestamp in the session traffic. **Note:** The RSPAN VLAN and trunk configuration to allow the RSPAN VLAN must be configured before using this command.
6.	**end**	Returns to privileged EXEC mode.
7.	**show monitor** [**session** *session_number*] **show running-config**	Verifies the configuration.
8.	**copy running-config startup-config**	(Optional) Saves the configuration in the configuration file.

Monitoring and Maintaining SPAN and RSPAN

show monitor [**session** *session_number*]	Verifies the SPAN or RSPAN configuration.

Spanned Traffic Timestamping Statistics

show platform timestamp-trailer counters	Display the timestamp statistics.

```
Switch#show platform timestamp-trailer counters

Port          Timestamped packets  Timestamp Removed  Dropped Packets
Gi1/1                  0              0            0
Gi1/2                  0              0            0
Gi1/3                  0              0            0
Gi1/4                  0              0            0
Gi1/5                  0              0            0
Gi1/6                  0              0            0
Gi1/7                  0              0            0
Gi1/8                  0              0            0
Gi1/9                  0              0            0
Gi1/10                 0              0            0
Gi1/11                 0              0            0
Gi1/12                 0              0            0
Gi1/13             10000              0            0
Gi1/14                 0              0            0
Gi1/15                 0              0            0
Gi1/16                 0              0            0
Gi1/17                 0              0            0
Gi1/18                 0              0            0
Gi1/19                 0              0            0
Gi1/20                 0              0            0
Gi1/21                 0              0            0
Gi1/22                 0              0            0
Gi1/23                 0              0            0
Gi1/24                 0              0            0
Gi1/25                 0              0        10000
Gi1/26                 0              0            0
Gi1/27                 0              0            0
Gi1/28                 0              0            0
```

“Timestamped packets” is the count of ingress packets to which the timestamp trailer is added.

“Timestamp Removed” is the count of egress packets from which the timestamp trailer is removed.

“Dropped Packets” is the count of ingress packets to which the timestamp trailer is added and dropped (not forwarded to any port).

To clear the timestamp statistics, enter the command:

clear platform timestamp-trailer counters

Configuration Examples for SPAN and RSPAN

Configuring a Local SPAN Session: Example

This example shows how to set up SPAN session 1 for monitoring source port traffic to a destination port. First, any existing SPAN configuration for session 1 is deleted, and then bidirectional traffic is mirrored from source Gigabit Ethernet port 1 to destination Gigabit Ethernet port 2, retaining the encapsulation method.

```
Switch(config)# no monitor session 1
Switch(config)# monitor session 1 source interface GigabitEthernet1/17
Switch(config)# monitor session 1 destination interface GigabitEthernet1/18 encapsulation replicate
Switch(config)# end
```

Modifying Local SPAN Sessions: Examples

This example shows how to remove port 1 as a SPAN source for SPAN session 1:

```
Switch(config)# no monitor session 1 source interface GigabitEthernet1/17
Switch(config)# end
```

This example shows how to disable received traffic monitoring on port 1, which was configured for bidirectional monitoring:

```
Switch(config)# no monitor session 1 source interface GigabitEthernet1/17 rx
```

The monitoring of traffic received on port 1 is disabled, but traffic sent from this port continues to be monitored.

This example shows how to remove any existing configuration on SPAN session 2, configure SPAN session 2 to monitor received traffic on all ports belonging to VLANs 1 through 3, and send it to destination Gigabit Ethernet port 2. The configuration is then modified to also monitor all traffic on all ports belonging to VLAN 10.

```
Switch(config)# no monitor session 2
Switch(config)# monitor session 2 source vlan 1 - 3 rx
Switch(config)# monitor session 2 destination interface GigabitEthernet1/18
Switch(config)# monitor session 2 source vlan 10
Switch(config)# end
```

This example shows how to remove any existing configuration on SPAN session 2, configure SPAN session 2 to monitor received traffic on Gigabit Ethernet source port 1, and send it to destination Gigabit Ethernet port 2 with the same egress encapsulation type as the source port, and to enable ingress forwarding with IEEE 802.1Q encapsulation and VLAN 6 as the default ingress VLAN.

```
Switch(config)# no monitor session 2
Switch(config)# monitor session 2 source GigabitEthernet1/17 rx
Switch(config)# monitor session 2 destination interface GigabitEthernet1/18 encapsulation replicate
ingress dot1q vlan 6
Switch(config)# end
```

To monitor all VLANs on the trunk port, use the **no monitor session** *session_number* **filter** global configuration command.

This example shows how to remove any existing configuration on SPAN session 2, configure SPAN session 2 to monitor traffic received on Gigabit Ethernet trunk port 2, and send traffic for only VLANs 1 through 5 and VLAN 9 to destination Gigabit Ethernet port 1:

```
Switch(config)# no monitor session 2
Switch(config)# monitor session 2 source interface GigabitEthernet1/18 rx
Switch(config)# monitor session 2 filter vlan 1 - 5, 9
Switch(config)# monitor session 2 destination interface GigabitEthernet1/17
Switch(config)# end
```

Configuring an RSPAN: Example

This example shows how to create RSPAN VLAN 901:

```
Switch(config)# vlan 901
Switch(config-vlan)# remote span
Switch(config-vlan)# end
```

Configuring a VLAN for a SPAN Session: Example

This example shows how to configure VLAN 901 as the source remote VLAN and port 1 as the destination interface:

```
Switch(config)# monitor session 1 source remote vlan 901
Switch(config)# monitor session 1 destination interface GigabitEthernet1/17
Switch(config)# end
```

Modifying RSPAN Sessions: Examples

This example shows how to remove any existing RSPAN configuration for session 1, configure RSPAN session 1 to monitor multiple source interfaces, and configure the destination as RSPAN VLAN 901:

```
Switch(config)# no monitor session 1
Switch(config)# monitor session 1 source interface GigabitEthernet1/17 tx
Switch(config)# monitor session 1 source interface GigabitEthernet1/18 rx

Switch(config)# monitor session 1 source interface port-channel 2
Switch(config)# monitor session 1 destination remote vlan 901
Switch(config)# end
```

This example shows how to configure VLAN 901 as the source remote VLAN in RSPAN session 2, to configure Gigabit Ethernet source port 2 as the destination interface, and to enable forwarding of incoming traffic on the interface with VLAN 6 as the default receiving VLAN:

```
Switch(config)# monitor session 2 source remote vlan 901
Switch(config)# monitor session 2 destination interface GigabitEthernet1/18 ingress vlan 6
Switch(config)# end
```

This example shows how to remove any existing configuration on RSPAN session 2, configure RSPAN session 2 to monitor traffic received on trunk port 2, and send traffic for only VLANs 1 through 5 and 9 to destination RSPAN VLAN 902:

```
Switch(config)# no monitor session 2
(config)# monitor session 2 source interface GigabitEthernet1/18 rx
Switch(config)# monitor session 2 filter vlan 1 - 5, 9
Switch(config)# monitor session 2 destination remote vlan 902
Switch(config)# end
```

Configuring an RSPAN Session with Timestamp: Example

```
Switch(config)# vlan 4
Switch(config-vlan)# remote span
Switch(config-vlan)# end
```

Configure downlink Gigabit Ethernet interfaces 1/13 - 1/24 as access port:

```
Switch(config)# interface range gigabitEthernet 1/13 - 24
Switch(config-if-range)# switchport mode access
Switch(config-if-range) end
```

Configure uplink Gigabit Ethernet interface 1/25 as trunk port:

```
Switch(config)# interface gigabitEthernet 1/25
Switch(config-if)# switchport mode trunk
Switch(config-if)# end
```

Configure RSPAN session:

```
Switch(config)# monitor session 1 source interface gigabitEthernet 1/13 - 24
Switch(config)# monitor session 1 destination remote vlan 4
Switch(config)# monitor session 1 timestamp
Switch(config)# end
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	–

MIBs

MIBs	MIBs Link
–	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	–

Configuring LLDP, LLDP-MED, and Wired Location Service

Information About LLDP, LLDP-MED, and Wired Location Service

The Cisco Discovery Protocol (CDP) is a device discovery protocol that runs over Layer 2 (the data link layer) on all Cisco-manufactured devices (routers, bridges, access servers, and switches). CDP allows network management applications to automatically discover and learn about other Cisco devices connected to the network.

To support non-Cisco devices and to allow for interoperability between other devices, the switch supports the IEEE 802.1AB Link Layer Discovery Protocol (LLDP). LLDP is a neighbor discovery protocol that is used for network devices to advertise information about themselves to other devices on the network. This protocol runs over the data-link layer, which allows two systems running different network layer protocols to learn about each other.

LLDP supports a set of attributes that it uses to discover neighbor devices. These attributes contain type, length, and value descriptions and are referred to as TLVs. LLDP supported devices can use TLVs to receive and send information to their neighbors. This protocol can advertise details such as configuration information, device capabilities, and device identity.

The switch supports these basic management TLVs. These are mandatory LLDP TLVs.

- Port description TLV
- System name TLV
- System description TLV
- System capabilities TLV
- Management address TLV

These organizationally specific LLDP TLVs are also advertised to support LLDP-MED:

- Port VLAN ID TLV ((IEEE 802.1 organizationally specific TLVs)
- MAC/PHY configuration/status TLV(IEEE 802.3 organizationally specific TLVs)

Note: A switch stack appears as a single switch in the network. Therefore, LLDP discovers the switch stack, not the individual stack members.

LLDP-MED

LLDP for Media Endpoint Devices (LLDP-MED) is an extension to LLDP that operates between endpoint devices such as IP phones and network devices such as switches. It specifically provides support for voice over IP (VoIP) applications and provides additional TLVs for capabilities discovery, network policy, Power over Ethernet, inventory management and location information. By default, all LLDP-MED TLVs are enabled.

LLDP-MED supports these TLVs:

- LLDP-MED capabilities TLV

  Allows LLDP-MED endpoints to determine the capabilities that the connected device supports and has enabled.

- Network policy TLV

  Allows both network connectivity devices and endpoints to advertise VLAN configurations and associated Layer 2 and Layer 3 attributes for the specific application on that port. For example, the switch can notify a phone of the VLAN number that it should use. The phone can connect to any switch, obtain its VLAN number, and then start communicating with the call control.

  By defining a network-policy profile TLV, you can create a profile for voice and voice-signalling by specifying the values for VLAN, class of service (CoS), differentiated services code point (DSCP), and tagging mode. These profile attributes are then maintained centrally on the switch and propagated to the phone.

- Power management TLV

  Enables advanced power management between LLDP-MED endpoint and network connectivity devices. Allows switches and phones to convey power information, such as how the device is powered, power priority, and how much power the device needs.

- Inventory management TLV

  Allows an endpoint to send detailed inventory information about itself to the switch, including information hardware revision, firmware version, software version, serial number, manufacturer name, model name, and asset ID TLV.

- Location TLV

  Provides location information from the switch to the endpoint device. The location TLV can send this information:

  - Civic location information

    Provides the civic address information and postal information. Examples of civic location information are street address, road name, and postal community name information.

  - ELIN location information

    Provides the location information of a caller. The location is determined by the emergency location identifier number (ELIN), which is a phone number that routes an emergency call to the local public safety answering point (PSAP) and which the PSAP can use to call back the emergency caller.

Wired Location Service

The switch uses the wired location service feature to send location and attachment tracking information for its connected devices to a Cisco Mobility Services Engine (MSE). The tracked device can be a wireless endpoint, a wired endpoint, or a wired switch or controller. The switch notifies the MSE of device link up and link down events through the Network Mobility Services Protocol (NMSP) location and attachment notifications.

The MSE starts the NMSP connection to the switch, which opens a server port. When the MSE connects to the switch there are a set of message exchanges to establish version compatibility and service exchange information followed by location information synchronization. After connection, the switch periodically sends location and attachment notifications to the MSE. Any link up or link down events detected during an interval are aggregated and sent at the end of the interval.

When the switch determines the presence or absence of a device on a link-up or link-down event, it obtains the client-specific information such as the MAC address, IP address, and username. If the client is LLDP-MED- or CDP-capable, the switch obtains the serial number and UDI through the LLDP-MED location TLV or CDP.

Depending on the device capabilities, the switch obtains this client information at link up:

- Slot and port specified in port connection

- MAC address specified in the client MAC address
- IP address specified in port connection
- 802.1X username if applicable
- Device category is specified as a *wired station*
- State is specified as *new*
- Serial number, UDI
- Model number
- Time in seconds since the switch detected the association

Depending on the device capabilities, the switch obtains this client information at link down:

- Slot and port that was disconnected
- MAC address
- IP address
- 802.1X username if applicable
- Device category is specified as a *wired station*
- State is specified as *delete*
- Serial number, UDI
- Time in seconds since the switch detected the disassociation

When the switch shuts down, it sends an attachment notification with the state *delete* and the IP address before closing the NMSP connection to the MSE. The MSE interprets this notification as disassociation for all the wired clients associated with the switch.

If you change a location address on the switch, the switch sends an NMSP location notification message that identifies the affected ports and the changed address information.

Default LLDP Configuration

Feature	Default Setting
LLDP global state	Disabled.
LLDP holdtime (before discarding)	120 seconds.
LLDP timer (packet update frequency)	30 seconds.
LLDP reinitialization delay	2 seconds.
LLDP tlv-select	Disabled to send and receive all TLVs.
LLDP interface state	Disabled.
LLDP receive	Disabled.
LLDP transmit	Disabled.
LLDP med-tlv-select	Disabled to send all LLDP-MED TLVs. When LLDP is globally enabled, LLDP-MED-TLV is also enabled.

LLDP, LLDP-MED, and Wired Location Service Configuration Guidelines

- If the interface is configured as a tunnel port, LLDP is automatically disabled.
- If you first configure a network-policy profile on an interface, you cannot apply the **switchport voice vlan** command on the interface. If the **switchport voice vlan** *vlan-id* is already configured on an interface, you can apply a network-policy profile on the interface. This way the interface has the voice or voice-signaling VLAN network-policy profile applied on the interface.
- You cannot configure static secure MAC addresses on an interface that has a network-policy profile.
- You cannot configure a network-policy profile on a private-VLAN port.
- For wired location to function, you must first enter the **ip device tracking** global configuration command.

LLDP-MED TLVs

By default, the switch only sends LLDP packets until it receives LLDP-MED packets from the end device. It then sends LLDP packets with MED TLVs. When the LLDP-MED entry has been aged out, it only sends LLDP packets.

By using the **lldp** interface configuration command, you can configure the interface not to send the TLVs listed in this table.

LLDP-MED TLV	Description
inventory-management	LLDP-MED inventory management TLV
location	LLDP-MED location TLV
network-policy	LLDP-MED network policy TLV
power-management	LLDP-MED power management TLV

How to Configure LLDP, LLDP-MED, and Wired Location Service

Enabling LLDP

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**lldp run**	Enables LLDP globally on the switch.
3.	**interface** *interface-id*	Specifies the interface on which you are enabling LLDP, and enter interface configuration mode.
4.	**lldp transmit**	Enables the interface to send LLDP packets.
5.	**lldp receive**	Enables the interface to receive LLDP packets.
6.	**end**	Returns to privileged EXEC mode.

Configuring LLDP Characteristics

You can configure the frequency of LLDP updates, the amount of time to hold the information before discarding it, and the initialization delay time. You can also select the LLDP and LLDP-MED TLVs to send and receive.

Note: Steps 2 through 5 are optional and can be performed in any order.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**lldp holdtime** *seconds*	(Optional) Specifies the amount of time a receiving device should hold the information from your device before discarding it. The range is 0 to 65535 seconds; the default is 120 seconds.
3.	**lldp reinit** *delay*	(Optional) Specifies the delay time in seconds for LLDP to initialize on an interface. The range is 2 to 5 seconds; the default is 2 seconds.
4.	**lldp timer** *rate*	(Optional) Sets the sending frequency of LLDP updates in seconds. The range is 5 to 65534 seconds; the default is 30 seconds.
5.	**lldp tlv-select**	(Optional) Specifies the LLDP TLVs to send or receive.
6.	**lldp med-tlv-select**	(Optional) Specifies the LLDP-MED TLVs to send or receive.
7.	**end**	Returns to privileged EXEC mode.

Configuring LLDP-MED TLVs

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the interface on which you are configuring an LLDP-MED TLV, and enters interface configuration mode.
3.	**lldp med-tlv-select** ***tlv***	Specifies the TLV to enable.
4.	**end**	Returns to privileged EXEC mode.

Configuring Network-Policy TLV

This task explains how to create a network-policy profile, configure the policy attributes, and apply it to an interface.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**network-policy profile** *profile number*	Specifies the network-policy profile number, and enters network-policy configuration mode. The range is 1 to 4294967295.
3.	**{voice | voice-signaling} vlan [***vlan-id* **{cos** *cvalue* **| dscp** *dvalue***}***]* **| [[dot1p {cos** *cvalue* **| dscp** *dvalue***}] | none | untagged]**	Configures the policy attributes: **voice**–Specifies the voice application type. **voice-signaling**–Specifies the voice-signaling application type. **vlan**–Specifies the native VLAN for voice traffic. *vlan-id*–(Optional) Specifies the VLAN for voice traffic. The range is 1 to 4096. **cos** *cvalue*–(Optional) Specifies the Layer 2 priority class of service (CoS) for the configured VLAN. The range is 0 to 7; the default is 0. **dscp** *dvalue*–(Optional) Specifies the differentiated services code point (DSCP) value for the configured VLAN. The range is 0 to 63; the default is 0. **dot1p**–(Optional) Configures the telephone to use IEEE 802.1p priority tagging and use VLAN 0 (the native VLAN). **none**–(Optional) Does not instruct the IP telephone about the voice VLAN. The telephone uses the configuration from the telephone key pad. **untagged**–(Optional) Configures the telephone to send untagged voice traffic. This is the default for the telephone.
4.	**exit**	Returns to global configuration mode.
5.	**interface** *interface-id*	Specifies the interface on which you are configuring a network-policy profile, and enter interface configuration mode.
6.	**network-policy** ***profile number***	Specifies the network-policy profile number.
7.	**lldp med-tlv-select network-policy**	Specifies the network-policy TLV.
8.	**end**	Returns to privileged EXEC mode.

Configuring Location TLV and Wired Location Service

This task explains how to configure location information for an endpoint and to apply it to an interface.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**location {admin-tag** *string* | **civic-location identifier** *id* | **elin-location** *string* **identifier** *id}*	Specifies the location information for an endpoint. ■ **admin-tag**–Specifies an administrative tag or site information. ■ **civic-location**–Specifies civic location information. ■ **elin-location**–Specifies emergency location information (ELIN). ■ **identifier** *id*–Specifies the ID for the civic location. ■ *string*–Specifies the site or location information in alphanumeric format.
3.	**exit**	Returns to global configuration mode.
4.	**interface** *interface-id*	Specifies the interface on which you are configuring the location information, and enters interface configuration mode.
5.	**location {additional-location-information** ***word*** **| civic-location-id** *id* **| elin-location-id** *id}*	Enters location information for an interface: **additional-location-information**–Specifies additional information for a location or place. **civic-location-id**–Specifies global civic location information for an interface. **elin-location-id**–Specifies emergency location information for an interface. *id*–Specifies the ID for the civic location or the ELIN location. The ID range is 1 to 4095. ***word***–Specifies a word or phrase with additional location information.
6.	**end**	Returns to privileged EXEC mode.
7.	**nmsp enable**	Enables the NMSP features on the switch.
8.	**nmsp notification interval** *{***attachment** *|* **location***}* *interval-seconds*	Specifies the NMSP notification interval. **attachment**–Specifies the attachment notification interval. **location–**Specifies the location notification interval. *interval-seconds*–Duration in seconds before the switch sends the MSE the location or attachment updates. The range is 1 to 30; the default is 30.
9.	**end**	Returns to privileged EXEC mode.

Monitoring and Maintaining LLDP, LLDP-MED, and Wired Location Service

Command	Description
clear lldp counters	Resets the traffic counters to zero.
clear lldp table	Deletes the LLDP neighbor information table.
clear nmsp statistics	Clears the NMSP statistic counters.
show lldp	Displays global information, such as frequency of transmissions, the holdtime for packets being sent, and the delay time before LLDP initializes on an interface.
show lldp entry *entry-name*	Displays information about a specific neighbor. You can enter an asterisk (*) to display all neighbors, or you can enter the neighbor name.
show lldp interface [*interface-id*]	Displays information about interfaces with LLDP enabled. You can limit the display to a specific interface.
show lldp neighbors [*interface-id*] [**detail**]	Displays information about neighbors, including device type, interface type and number, holdtime settings, capabilities, and port ID. You can limit the display to neighbors of a specific interface or expand the display for more detailed information.
show lldp traffic	Displays LLDP counters, including the number of packets sent and received, number of packets discarded, and number of unrecognized TLVs.
show location admin-tag *string*	Displays the location information for the specified administrative tag or site.
show location civic-location identifier *id*	Displays the location information for a specific global civic location.
show location elin-location identifier *id*	Displays the location information for an emergency location.
show network-policy profile	Displays the configured network-policy profiles.
show nmsp	Displays the NMSP information.

Configuration Examples for Configuring LLDP, LLDP-MED, and Wired Location Service

Enabling LLDP: Examples

This example shows how to globally enable LLDP:

```
Switch# configure terminal
Switch(config)# lldp run
Switch(config)# end
```

This example shows how to enable LLDP on an interface:

```
Switch# configure terminal
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# lldp transmit
Switch(config-if)# lldp receive
Switch(config-if)# end
```

Configuring LDP Parameters: Examples

This example shows how to configure LLDP parameters:

```
Switch# configure terminal
Switch(config)# lldp holdtime 120
Switch(config)# lldp reinit 2
Switch(config)# lldp timer 30
Switch(config)# end
```

Configuring TLV: Example

This example shows how to enable a TLV on an interface:

```
Switch# configure terminal
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# lldp med-tlv-select inventory-management
Switch(config-if)# end
```

Configuring Network Policy: Example

This example shows how to configure VLAN 100 for voice application with CoS and to enable the network-policy profile and network-policy TLV on an interface:

```
Switch# configure terminal
Switch(config)# network-policy profile 1
Switch(config-network-policy)# voice vlan 100 cos 4
Switch(config-network-policy)# exit
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# network-policy profile 1
Switch(config-if)# lldp med-tlv-select network-policy
```

Configuring Voice Application: Example

This example shows how to configure the voice application type for the native VLAN with priority tagging:

```
Switch(config-network-policy)# voice vlan dot1p cos 4
Switch(config-network-policy)# voice vlan dot1p dscp 34
```

Configuring Civic Location Information: Example

This example shows how to configure civic location information on the switch:

```
Switch(config)# location civic-location identifier 1
Switch(config-civic)# number 3550
Switch(config-civic)# primary-road-name "Cisco Way"
Switch(config-civic)# city "San Jose"
Switch(config-civic)# state CA
Switch(config-civic)# building 19
Switch(config-civic)# room C6
Switch(config-civic)# county "Santa Clara"
Switch(config-civic)# country US
Switch(config-civic)# end
```

Enabling NMSP: Example

This example shows how to enable NMSP on a switch and to set the location notification time to 10 seconds:

```
Switch(config)# nmsp enable
Switch(config)# nmsp notification interval location 10
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands Cisco IOS system management commands	*Cisco IOS Configuration Fundamentals Command Reference*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	–

MIBs

MIBs	MIBs Link
–	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	–

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Configuring Layer 2 NAT

One-to-one (1:1) Layer 2 Network Address Translation (NAT) is a service that allows the assignment of a unique public IP address to an existing private IP address (end device), so that the end device can communicate on both the private and public subnets. This service is configured in a NAT-enabled device and is the public “alias” of the IP address physically programmed on the end device. This is typically represented by a table in the NAT device.

Layer 2 NAT has two translation tables where private-to-public and public-to-private subnet translations can be defined. Layer 2 NAT is a hardware based implementation that provides the same high level of (bump-on-the-wire) wire-speed performance. This implementation also supports multiple VLANs through the NAT boundary for enhanced network segmentation.

For information about configuring Layer 2 NAT on a Cisco Industrial Ethernet Switch, see Layer 2 NAT Software Configuration Guide for Cisco Industrial Ethernet Switches.

Note - The IE 4010 and 5000 follow the same rules documented in the Layer 2 Nat guide.

Configuring CDP

Information About CDP

CDP

CDP is a device discovery protocol that runs over Layer 2 (the data link layer) on all Cisco-manufactured devices (routers, bridges, access servers, and switches) and allows network management applications to discover Cisco devices that are neighbors of already known devices. With CDP, network management applications can learn the device type and the Simple Network Management Protocol (SNMP) agent address of neighboring devices running lower-layer, transparent protocols. This feature enables applications to send SNMP queries to neighboring devices.

CDP runs on all media that support Subnetwork Access Protocol (SNAP). Because CDP runs over the data-link layer only, two systems that support different network-layer protocols can learn about each other.

Each CDP-configured device sends periodic messages to a multicast address, advertising at least one address at which it can receive SNMP messages. The advertisements also contain time-to-live, or holdtime information, which is the length of time a receiving device holds CDP information before discarding it. Each device also listens to the messages sent by other devices to learn about neighboring devices.

On the switch, CDP enables Network Assistant to display a graphical view of the network. The switch uses CDP to find cluster candidates and maintain information about cluster members and other devices up to three cluster-enabled devices away from the command switch by default.

For a switch and connected endpoint devices running Cisco Medianet, these events occur:

- CDP identifies connected endpoints that communicate directly with the switch.
- Only one wired switch reports the location information to prevent duplicate reports of neighboring devices.
- The wired switch and the endpoints both send and receive location information.

The switch supports CDP Version 2.

Default CDP Configuration

Feature	Default Setting
CDP global state	Enabled
CDP interface state	Enabled
CDP timer (packet update frequency)	60 seconds
CDP holdtime (before discarding)	180 seconds
CDP Version-2 advertisements	Enabled

How to Configure CDP

Configuring the CDP Parameters

You can configure the frequency of CDP updates, the amount of time to hold the information before discarding it, and whether or not to send Version-2 advertisements.

Note: Steps 2 through 4 are all optional and can be performed in any order.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**cdp timer** *seconds*	(Optional) Sets the transmission frequency of CDP updates in seconds. The range is 5 to 254; the default is 60 seconds.
3.	**cdp holdtime** *seconds*	(Optional) Specifies the amount of time a receiving device should hold the information sent by your device before discarding it. The range is 10 to 255 seconds; the default is 180 seconds.
4.	**cdp advertise-v2**	(Optional) Configures CDP to send Version-2 advertisements. This is the default state.
5.	**end**	Returns to privileged EXEC mode.

Disabling CDP

CDP is enabled by default.

Note: Switch clusters and other Cisco devices (such as Cisco IP Phones) regularly exchange CDP messages. Disabling CDP can interrupt cluster discovery and device connectivity.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**no cdp run**	Disables CDP globally.
3.	**interface** *interface-id*	Specifies the interface on which you are disabling CDP, and enters interface configuration mode.
4.	**no cdp enable**	Disables CDP on the interface.
5.	**end**	Returns to privileged EXEC mode.

Monitoring and Maintaining CDP

Command	Description
clear cdp counters	Resets the traffic counters to zero.
clear cdp table	Deletes the CDP table of information about neighbors.
show cdp	Displays global information, such as frequency of transmissions and the holdtime for packets being sent.
show cdp entry *entry-name* [**protocol** | **version**]	Displays information about a specific neighbor. You can enter an asterisk (*) to display all CDP neighbors, or you can enter the name of the neighbor about which you want information. You can also limit the display to information about the protocols enabled on the specified neighbor or information about the version of software running on the device.
show cdp interface [*interface-id*]	Displays information about interfaces where CDP is enabled. You can limit the display to the interface about which you want information.
show cdp neighbors [*interface-id*] [**detail**]	Displays information about neighbors, including device type, interface type and number, holdtime settings, capabilities, platform, and port ID. You can limit the display to neighbors of a specific interface or expand the display to provide more detailed information.
show cdp traffic	Displays CDP counters, including the number of packets sent and received and checksum errors.

Configuration Examples for CDP

Configuring CDP Parameters: Example

This example shows how to configure CDP parameters:

```
Switch# configure terminal
Switch(config)# cdp timer 50
Switch(config)# cdp holdtime 120
Switch(config)# cdp advertise-v2
Switch(config)# end
```

Enabling CDP: Examples

This example shows how to enable CDP on a port when it has been disabled:

```
Switch# configure terminal
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# cdp enable
Switch(config-if)# end
```

Note: Voice VLAN is not counted against port security when CDP is disabled on the switch interface.

This example shows how to enable CDP if it has been disabled:

```
Switch# configure terminal
Switch(config)# cdp run
Switch(config)# end
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands Cisco IOS system management commands	*Cisco IOS Configuration Fundamentals Command Reference*
Switch cluster configuration	Configuring Switch Clusters, page 91

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Configuring UDLD

Prerequisites for UDLD

- When configuring the mode (normal or aggressive), make sure that the same mode is configured on both sides of the link.

Restrictions for UDLD

- UDLD is not supported on ATM ports.
- A UDLD-capable port cannot detect a unidirectional link if it is connected to a UDLD-incapable port of another switch.
- Loop guard works only on point-to-point links. We recommend that each end of the link has a directly connected device that is running STP.

Information About UDLD

UDLD

UniDirectional Link Detection (UDLD) is a Layer 2 protocol that enables devices connected through fiber-optic or twisted-pair Ethernet cables to monitor the physical configuration of the cables and detect when a unidirectional link exists. All connected devices must support UDLD for the protocol to successfully identify and disable unidirectional links. When UDLD detects a unidirectional link, it disables the affected port and alerts you. Unidirectional links can cause a variety of problems, including spanning-tree topology loops.

Modes of Operation

UDLD supports two modes of operation: normal (the default) and aggressive. In normal mode, UDLD can detect unidirectional links due to misconnected ports on fiber-optic connections. In aggressive mode, UDLD can also detect unidirectional links due to one-way traffic on fiber-optic and twisted-pair links and to misconnected ports on fiber-optic links.

In normal and aggressive modes, UDLD works with the Layer 1 mechanisms to learn the physical status of a link. At Layer 1, autonegotiation takes care of physical signaling and fault detection. UDLD performs tasks that autonegotiation cannot perform, such as detecting the identities of neighbors and shutting down misconnected ports. When you enable both autonegotiation and UDLD, the Layer 1 and Layer 2 detections work together to prevent physical and logical unidirectional connections and the malfunctioning of other protocols.

A unidirectional link occurs whenever traffic sent by a local device is received by its neighbor but traffic from the neighbor is not received by the local device.

In normal mode, UDLD detects a unidirectional link when fiber strands in a fiber-optic port are misconnected and the Layer 1 mechanisms do not detect this misconnection. If the ports are connected correctly but the traffic is one way, UDLD does not detect the unidirectional link because the Layer 1 mechanism, which is supposed to detect this condition, does not do so. In this case, the logical link is considered undetermined, and UDLD does not disable the port.

When UDLD is in normal mode, if one of the fiber strands in a pair is disconnected, as long as autonegotiation is active, the link does not stay up because the Layer 1 mechanisms detects a physical problem with the link. In this case, UDLD does not take any action and the logical link is considered undetermined.

In aggressive mode, UDLD detects a unidirectional link by using the previous detection methods. UDLD in aggressive mode can also detect a unidirectional link on a point-to-point link on which no failure between the two devices is allowed. It can also detect a unidirectional link when one of these problems exists:

- On fiber-optic or twisted-pair links, one of the ports cannot send or receive traffic.
- On fiber-optic or twisted-pair links, one of the ports is down while the other is up.
- One of the fiber strands in the cable is disconnected.

In these cases, UDLD disables the affected port.

In a point-to-point link, UDLD hello packets can be considered as a heart beat whose presence guarantees the health of the link. Conversely, the loss of the heart beat means that the link must be shut down if it is not possible to reestablish a bidirectional link.

If both fiber strands in a cable are working normally from a Layer 1 perspective, UDLD in aggressive mode detects whether those fiber strands are connected correctly and whether traffic is flowing bidirectionally between the correct neighbors. This check cannot be performed by autonegotiation because autonegotiation operates at Layer 1.

Methods to Detect Unidirectional Links

UDLD operates by using two methods:

- Neighbor database maintenance

  UDLD learns about other UDLD-capable neighbors by periodically sending a hello packet (also called an advertisement or probe) on every active port to keep each device informed about its neighbors.

  When the switch receives a hello message, it caches the information until the age time (hold time or time-to-live) expires. If the switch receives a new hello message before an older cache entry ages, the switch replaces the older entry with the new one.

  Whenever a port is disabled and UDLD is running, whenever UDLD is disabled on a port, or whenever the switch is reset, UDLD clears all existing cache entries for the ports affected by the configuration change. UDLD sends at least one message to inform the neighbors to flush the part of their caches affected by the status change. The message is intended to keep the caches synchronized.

- Event-driven detection and echoing

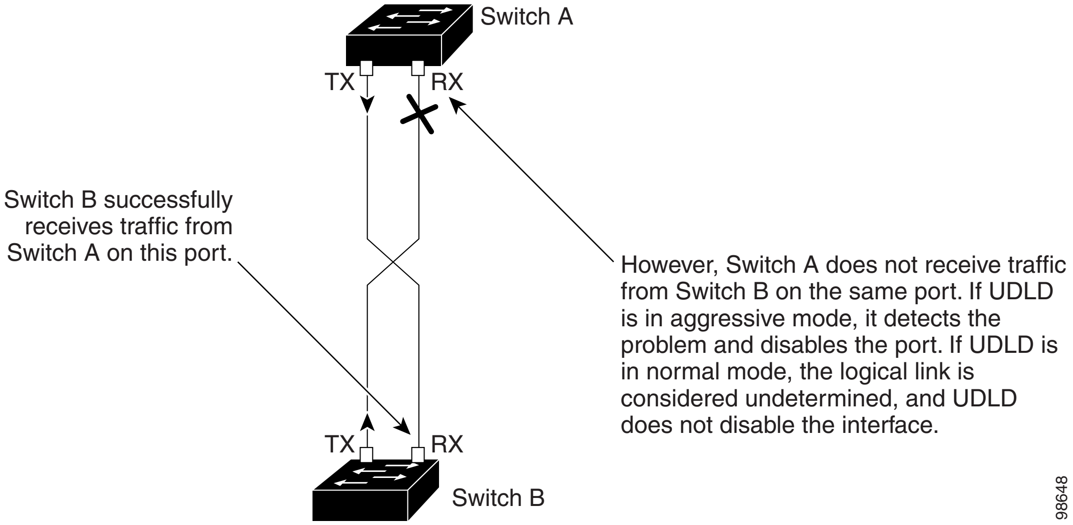
  UDLD relies on echoing as its detection mechanism. Whenever a UDLD device learns about a new neighbor or receives a resynchronization request from an out-of-sync neighbor, it restarts the detection window on its side of the connection and sends echo messages in reply. Because this behavior is the same on all UDLD neighbors, the sender of the echoes expects to receive an echo in reply.

  If the detection window ends and no valid reply message is received, the link might shut down, depending on the UDLD mode. When UDLD is in normal mode, the link might be considered undetermined and might not be shut down. When UDLD is in aggressive mode, the link is considered unidirectional, and the port is disabled.

If UDLD in normal mode is in the advertisement or in the detection phase and all the neighbor cache entries are aged out, UDLD restarts the link-up sequence to resynchronize with any potentially out-of-sync neighbors.

If you enable aggressive mode when all the neighbors of a port have aged out either in the advertisement or in the detection phase, UDLD restarts the link-up sequence to resynchronize with any potentially out-of-sync neighbor. UDLD shuts down the port if, after the fast train of messages, the link state is still undetermined.

Figure 71 UDLD Detection of a Unidirectional Link


Switch A
TX
RX
Switch B successfully receives traffic from Switch A on this port.
However, Switch A does not receive traffic from Switch B on the same port. If UDLD is in aggressive mode, it detects the problem and disables the port. If UDLD is in normal mode, the logical link is considered undetermined, and UDLD does not disable the interface.
TX
RX
Switch B
98648

Default UDLD Settings

Feature	Default Setting
UDLD global enable state	Globally disabled
UDLD per-port enable state for fiber-optic media	Disabled 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 aggressive mode	Disabled

How to Configure UDLD

Enabling UDLD Globally

Follow these steps to enable UDLD in the aggressive or normal mode and to set the configurable message timer on all fiber-optic ports on the switch:

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**udld** {**aggressive** | **enable** | **message time** *message-timer-interval*}	Specifies the UDLD mode of operation: ■ **aggressive**—Enables UDLD in aggressive mode on all fiber-optic ports. ■ **enable**—Enables UDLD in normal mode on all fiber-optic ports on the switch. UDLD is disabled by default. An individual interface configuration overrides the setting of the **udld enable** global configuration command. For more information about aggressive and normal modes, see Modes of Operation, page 533. ■ **message time** *message-timer-interval*—Configures the period of time between UDLD probe messages on ports that are in the advertisement phase and are detected to be bidirectional. The range is from 1 to 90 seconds. **Note:** This command affects fiber-optic ports only. Use the **udld** interface configuration command to enable UDLD on other port types. For more information, see Enabling UDLD on an Interface, page 536.
3.	**end**	Returns to privileged EXEC mode.

Enabling UDLD on an Interface

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the port to be enabled for UDLD, and enters interface configuration mode.
3.	**udld port** [**aggressive**]	UDLD is disabled by default. ■ **udld port**—Enables UDLD in normal mode on the specified port. ■ **udld port aggressive**—Enables UDLD in aggressive mode on the specified port. **Note:** Use the **no udld port** interface configuration command to disable UDLD on a specified fiber-optic port. For more information about aggressive and normal modes, see Modes of Operation, page 533.
4.	**end**	Returns to privileged EXEC mode.

Setting and Resetting UDLD Parameters

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**udld reset**	(Optional) Resets all ports disabled by UDLD.
3.	**no udld** {**aggressive | enable**}	(Optional) Disables the UDLD ports.
4.	**udld** {**aggressive | enable**}	(Optional) Reenables the disabled ports.
5.	**errdisable recovery cause udld**	(Optional) Enables the timer to automatically recover from the UDLD error-disabled state.
6.	**errdisable recovery interval** *interval*	(Optional) Specifies the time to recover from the UDLD error-disabled state.
7.	**interface** *interface-id*	Enters interface configuration mode.
8.	**no udld port**	(Optional) Disables the UDLD fiber-optic port.
9.	**udld port** [**aggressive**]	(Optional) Re-enables the disabled fiber-optic port.
10.	**shutdown**	(Optional) Disables an interface port.
11.	**no shutdown**	(Optional) Restarts a disabled port.
12.	**show udld**	(Optional) Verifies your entries.

Maintaining and Monitoring UDLD

Command	Purpose
show udld [*interface-id*]	Displays UDLD status.

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Configuring RMON

Prerequisites for RMON

- You must configure SNMP on the switch to access RMON MIB objects.
- We recommend that you use a generic RMON console application on the network management station (NMS) to take advantage of the RMON network management capabilities.

Restrictions for RMON

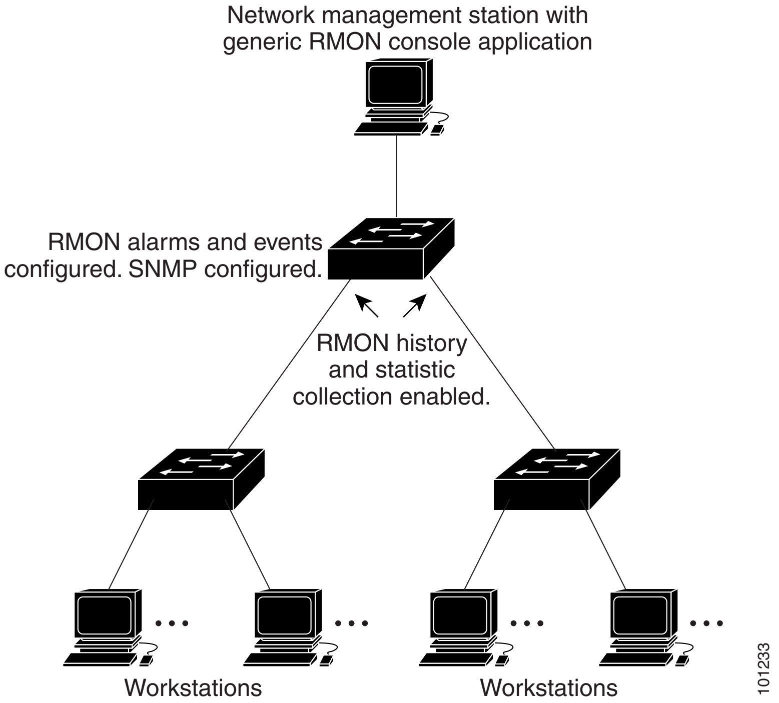
- 64-bit counters are not supported for RMON alarms.

Information About RMON

RMON

RMON is an Internet Engineering Task Force (IETF) standard monitoring specification that allows various network agents and console systems to exchange network monitoring data. You can use the RMON feature with the Simple Network Management Protocol (SNMP) agent in the switch to monitor all the traffic flowing among switches on all connected LAN segments as shown in Figure 72 on page 540.

Figure 72 Remote Monitoring Example



The switch supports these RMON groups (defined in RFC 1757):

- Statistics (RMON group 1)—Collects Ethernet statistics on an interface.
- History (RMON group 2)—Collects a history group of statistics on Ethernet ports for a specified polling interval.
- Alarm (RMON group 3)—Monitors a specific management information base (MIB) object for a specified interval, triggers an alarm at a specified value (rising threshold), and resets the alarm at another value (falling threshold). Alarms can be used with events; the alarm triggers an event, which can generate a log entry or an SNMP trap.
- Event (RMON group 9)—Specifies the action to take when an event is triggered by an alarm. The action can be to generate a log entry or an SNMP trap.

Because switches supported by this software release use hardware counters for RMON data processing, the monitoring is more efficient, and little processing power is required.

Note: 64-bit counters are not supported for RMON alarms.

RMON is disabled by default; no alarms or events are configured.

How to Configure RMON

Configuring RMON Alarms and Events

You can configure your switch for RMON by using the command-line interface (CLI) or an SNMP-compatible network management station.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**rmon alarm** *number variable interval* {**absolute** | **delta**} **rising-threshold** *value* [*event-number*] **falling-threshold** *value* [*event-number*] [**owner** *string*]	Sets an alarm on a MIB object. ■ *number*–Specifies the alarm number. The range is 1 to 65535. ■ *variable*–Specifies the MIB object to monitor. ■ *interval*–Specifies the time in seconds the alarm monitors the MIB variable. The range is 1 to 4294967295 seconds. ■ Specifies the **absolute** keyword to test each MIB variable directly. Specifies the **delta** keyword to test the change between samples of a MIB variable. ■ *value*–Specifies a number at which the alarm is triggered and one for when the alarm is reset. The range for the rising threshold and falling threshold values is -2147483648 to 2147483647. ■ (Optional) *event-number*–Specifies the event number to trigger when the rising or falling threshold exceeds its limit. ■ (Optional) **owner** *string*–Specifies the owner of the alarm.
3.	**rmon event** *number* [**description** *string*] [**log**] [**owner** *string*] [**trap** *community*]	Adds an event in the RMON event table that is associated with an RMON event number. ■ *number*–Assigns an event number. The range is 1 to 65535. ■ (Optional) **description** *string*–Specifies a description of the event. ■ (Optional) **log**–Generates an RMON log entry when the event is triggered. ■ (Optional) **owner** *string*–Specifies the owner of this event. ■ (Optional) **trap** *community*–Enters the SNMP community string used for this trap.
4.	**end**	Returns to privileged EXEC mode.

Collecting Group History Statistics on an Interface

You must first configure RMON alarms and events to display collection information.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the interface on which to collect history, and enters interface configuration mode.
3.	**rmon collection history** *index* [**buckets** *bucket-number*] [**interval** *seconds*] [**owner** *ownername*]	Enables history collection for the specified number of buckets and time period. ■ *index*–Identifies the RMON group of statistics. The range is 1 to 65535. ■ (Optional) **buckets** *bucket-number*–Specifies the maximum number of buckets desired for the RMON collection history group of statistics. The range is 1 to 65535. The default is 50 buckets. ■ (Optional) **interval** *seconds*–Specifies the number of seconds in each polling cycle. The range is 1 to 3600. The default is 1800 seconds. ■ (Optional) **owner** *ownername*–Enters the name of the owner of the RMON group of statistics.
4.	**end**	Returns to privileged EXEC mode.

Collecting Group Ethernet Statistics on an Interface

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the interface on which to collect statistics, and enters interface configuration mode.
3.	**rmon collection stats** *index* [**owner** *ownername*]	Enables RMON statistic collection on the interface. ■ *index*–Specifies the RMON group of statistics. The range is from 1 to 65535. ■ (Optional) **owner** *ownername*–Enters the name of the owner of the RMON group of statistics.
4.	**end**	Returns to privileged EXEC mode.

Monitoring and Maintaining RMON

Command	Purpose
show rmon	Displays general RMON statistics.
show rmon alarms	Displays the RMON alarm table.
show rmon events	Displays the RMON event table.
show rmon history	Displays the RMON history table.
show rmon statistics	Displays the RMON statistics table.

Configuration Examples for RMON

Configuring an RMON Alarm Number: Example

The following example shows how to configure an RMON alarm number:

```
Switch(config)# rmon alarm 10 ifEntry.20.1 20 delta rising-threshold 15 1 falling-threshold 0 owner
jjohnson
```

The alarm monitors the MIB variable *ifEntry.20.1* once every 20 seconds until the alarm is disabled and checks the change in the variable's rise or fall. If the *ifEntry.20.1* value shows a MIB counter increase of 15 or more, such as from 100000 to 100015, the alarm is triggered. The alarm in turn triggers event number 1, which is configured with the **rmon event** command. Possible events can include a log entry or an SNMP trap. If the *ifEntry.20.1* value changes by 0, the alarm is reset and can be triggered again.

Creating an RMON Event Number: Example

The following example creates RMON event number 1:

```
Switch(config)# rmon event 1 log trap eventtrap description "High ifOutErrors" owner jjones
```

The event is defined as *High ifOutErrors* and generates a log entry when the event is triggered by the alarm. The user *jjones* owns the row that is created in the event table by this command. This example also generates an SNMP trap when the event is triggered.

Configuring RMON Statistics: Example

This example shows how to collect RMON statistics for the owner *root*:

```
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# rmon collection stats 2 owner root
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands Cisco IOS system management commands	*Cisco IOS Configuration Fundamentals Command Reference*
SNMP configuration	Configuring SNMP, page 557
Alarm and event interaction	RFC 1757

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Configuring System Message Logging

Restrictions for System Message Logging

- Logging messages to the console at a high rate can result in high CPU utilization and adversely affect how the switch operates.

Information About System Message Logging

System Message Logging

By default, a switch sends the output from system messages and **debug** privileged EXEC commands to a logging process. The logging process controls the distribution of logging messages to various destinations, such as the logging buffer, terminal lines, or a UNIX syslog server, depending on your configuration. The process also sends messages to the console.

Note: The syslog format is compatible with 4.3 BSD UNIX.

When the logging process is disabled, messages are sent only to the console. The messages are sent as they are generated, so message and debug output are interspersed with prompts or output from other commands. Messages appear on the console after the process that generated them has finished.

You can set the severity level of the messages to control the type of messages displayed on the consoles and each of the destinations. You can time-stamp log messages or set the syslog source address to enhance real-time debugging and management.

You can access logged system messages by using the switch command-line interface (CLI) or by saving them to a properly configured syslog server. The switch software saves syslog messages in an internal buffer.

You can remotely monitor system messages by viewing the logs on a syslog server or by accessing the switch through Telnet or through the console port.

System Log Message Format

System log messages can contain up to 80 characters and a percent sign (%), which follows the optional sequence number or time-stamp information, if configured. Messages appear in this format:

seq no:timestamp: %facility-severity-MNEMONIC:description

The part of the message preceding the percent sign depends on the setting of the **service sequence-numbers**, **service timestamps log datetime**, **service timestamps log datetime** [**localtime**] [**msec**] [**show-timezone**], or **service timestamps log uptime** global configuration command.

Table 49 System Log Message Elements

Element	Description
seq no:	Stamps log messages with a sequence number only if the **service sequence-numbers** global configuration command is configured. For more information, see Enabling and Disabling Sequence Numbers in Log Messages, page 552.
timestamp formats: *mm/dd hh:mm:ss* or *hh:mm:ss* (short uptime) or *d h* (long uptime)	Date and time of the message or event. This information appears only if the **service timestamps log** [**datetime | log**] global configuration command is configured. For more information, see Enabling and Disabling Time Stamps on Log Messages, page 552.
facility	The facility to which the message refers (for example, SNMP, SYS, and so forth).
severity	Single-digit code from 0 to 7 that is the severity of the message.
MNEMONIC	Text string that uniquely describes the message.
description	Text string containing detailed information about the event being reported.

Log Messages

You can synchronize unsolicited messages and **debug** privileged EXEC command output with solicited device output and prompts for a specific console port line or virtual terminal line. You can identify the types of messages to be output asynchronously based on the level of severity. You can also configure the maximum number of buffers for storing asynchronous messages for the terminal after which messages are dropped.

When synchronous logging of unsolicited messages and **debug** command output is enabled, unsolicited device output appears on the console or printed after solicited device output appears or is printed. Unsolicited messages and **debug** command output appears on the console after the prompt for user input is returned. Therefore, unsolicited messages and **debug** command output are not interspersed with solicited device output and prompts. After the unsolicited messages appear, the console again displays the user prompt.

Message Severity Levels

Note: Specifying a *level* causes messages at that level and numerically lower levels to appear at the destination.

To disable logging to the console, use the **no logging console** global configuration command. To disable logging to a terminal other than the console, use the **no logging monitor** global configuration command. To disable logging to syslog servers, use the **no logging trap** global configuration command.

Table 50 on page 547 describes the *level* keywords. It also lists the corresponding UNIX syslog definitions from the most severe level to the least severe level.

Table 50 Level Keywords

Level Keyword	Level	Description	Syslog Definition
emergencies	0	System unstable	LOG_EMERG
alerts	1	Immediate action needed	LOG_ALERT
critical	2	Critical conditions	LOG_CRIT
errors	3	Error conditions	LOG_ERR
warnings	4	Warning conditions	LOG_WARNING
notifications	5	Normal but significant condition	LOG_NOTICE
informational	6	Informational messages only	LOG_INFO
debugging	7	Debugging messages	LOG_DEBUG

The software generates these categories of messages:

- Error messages about software or hardware malfunctions, displayed at levels **warnings** through **emergencies**. These types of messages mean that the functionality of the switch is affected.
- Output from the **debug** commands, displayed at the **debugging** level. Debug commands are typically used only by the Technical Assistance Center.
- Interface up or down transitions and system restart messages, displayed at the **notifications** level. This message is only for information; switch functionality is not affected.

Configuring UNIX Syslog Servers

The next sections describe how to configure the UNIX server syslog daemon and how to define the UNIX system logging facility.

Logging Messages to a UNIX Syslog Daemon

Before you can send system log messages to a UNIX syslog server, you must configure the syslog daemon on a UNIX server. This procedure is optional.

Note: Some recent versions of UNIX syslog daemons no longer accept by default syslog packets from the network. If this is the case with your system, use the UNIX **man syslogd** command to decide what options must be added to or removed from the syslog command line to enable logging of remote syslog messages.

Log in as root, and perform these steps:

1. Add a line such as the following to the file /etc/syslog.conf:

```
local7.debug /usr/adm/logs/cisco.log
```

The **local7** keyword specifies the logging facility to be used. The **debug** keyword specifies the syslog level. The syslog daemon sends messages at this level or at a more severe level to the file specified in the next field. The file must already exist, and the syslog daemon must have permission to write to it.

2. Create the log file by entering these commands at the UNIX shell prompt:

```
$ touch /var/log/cisco.log
$ chmod 666 /var/log/cisco.log
```

3. Make sure the syslog daemon reads the new changes:

```
$ kill -HUP `cat /etc/syslog.pid`
```

For more information, see the **man syslog.conf** and **man syslogd** commands on your UNIX system.

Table 51 on page 548 lists the UNIX system facilities supported by the software. For more information about these facilities, consult the operator’s manual for your UNIX operating system.

Table 51 UNIX System Facilities

Facility Type Keyword	Description
auth	Authorization system
cron	Cron facility
daemon	System daemon
kern	Kernel
local0-7	Locally defined messages
lpr	Line printer system
mail	Mail system
news	USENET news
sys9-14	System use
syslog	System log
user	User process
uucp	UNIX-to-UNIX copy system

Default System Message Logging Configuration

Feature	Default Setting
System message logging to the console	Enabled.
Console severity	Debugging (and numerically lower levels).
Logging file configuration	No filename specified.
Logging buffer size	4096 bytes.
Logging history size	1 message.
Time stamps	Disabled.
Synchronous logging	Disabled.
Logging server	Disabled.
Syslog server IP address	None configured.
Configuration change logger	Disabled.
Server facility	Local7.
Server severity	Informational (and numerically lower levels).

How to Configure System Message Logging

Disabling Message Logging

Message logging is enabled by default. It must be enabled to send messages to any destination other than the console. When enabled, log messages are sent to a logging process, which logs messages to designated locations asynchronously to the processes that generated the messages.

Disabling the logging process can slow down the switch because a process must wait until the messages are written to the console before continuing. When the logging process is disabled, messages appear on the console as soon as they are produced, often appearing in the middle of command output.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**no logging console**	Disables message logging.
3.	**end**	Returns to privileged EXEC mode.

Setting the Message Display Destination Device

If message logging is enabled, you can send messages to specific locations in addition to the console. Beginning in privileged EXEC mode, use one or more of the following commands to specify the locations that receive messages:

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**logging buffered** [*size*]	Logs messages to an internal buffer on the switch. The range is 4096 to 2147483647 bytes. The default buffer size is 4096 bytes. If the switch fails, the log file is lost unless you had previously saved it to flash memory. See Step 4. **Note:** Do not make the buffer size too large because the switch could run out of memory for other tasks. Use the **show memory** privileged EXEC command to view the free processor memory on the switch. However, this value is the maximum available, and the buffer size should *not* be set to this amount.
3.	**logging** *host*	Logs messages to a UNIX syslog server host. *host*–Specifies the name or IP address of the host to be used as the syslog server. To build a list of syslog servers that receive logging messages, enter this command more than once.
4.	**logging file flash:***filename* [*max-file-size* [*min-file-size*]] [*severity-level-number* | *type*]	Stores log messages in a file in flash memory. ■ *filename*–Enters the log message filename. ■ (Optional) *max-file-size*–Specifies the maximum logging file size. The range is 4096 to 2147483647. The default is 4096 bytes. ■ (Optional) *min-file-size*–Specifies the minimum logging file size. The range is 1024 to 2147483647. The default is 2048 bytes. ■ (Optional) *severity-level-number | type*–Specifies either the logging severity level or the logging type. The severity range is 0 to 7. By default, the log file receives debugging messages and numerically lower levels.
5.	**end**	Returns to privileged EXEC mode.
6.	**terminal monitor**	Logs messages to a nonconsole terminal during the current session. Terminal parameter-setting commands are set locally and do not remain in effect after the session has ended. You must perform this step for each session to see the debugging messages.

Synchronizing Log Messages

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**line** [**console** | **vty**] *line-number* [*ending-line-number*]	Specifies the line to be configured for synchronous logging of messages. ■ Use the **console** keyword for configurations that occur through the switch console port. ■ Use the **line vty** *line-number* command to specify which vty lines are to have synchronous logging enabled. You use a vty connection for configurations that occur through a Telnet session. The range of line numbers is from 0 to 15. You can change the setting of all 16 vty lines at once by entering: **line vty 0 15** Or you can change the setting of the single vty line being used for your current connection. For example, to change the setting for vty line 2, enter: **line vty 2** When you enter this command, the mode changes to line configuration.
3.	**logging synchronous** [**level** [*severity-level* | **all**] | **limit** *number-of-buffers*]	Enables synchronous logging of messages. ■ (Optional) **level** *severity-level*–Specifies the message severity level. Messages with a severity level equal to or higher than this value are printed asynchronously. Low numbers mean greater severity and high numbers mean lesser severity. The default is 2. ■ (Optional) **level all**–Specifies that all messages are printed asynchronously regardless of the severity level. ■ (Optional) **limit** *number-of-buffers*–Specifies the number of buffers to be queued for the terminal after which new messages are dropped. The range is 0 to 2147483647. The default is 20.
4.	**end**	Returns to privileged EXEC mode.

Enabling and Disabling Time Stamps on Log Messages

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**service timestamps log uptime** or **service timestamps log datetime** [**msec**] [**localtime**] [**show-timezone**]	Enables log time stamps. The first command enables time stamps on log messages, showing the time since the system was rebooted. The second command enables time stamps on log messages. Depending on the options selected, the time stamp can include the date, time in milliseconds relative to the local time-zone, and the time zone name.
3.	**end**	Returns to privileged EXEC mode.

Enabling and Disabling Sequence Numbers in Log Messages

Because there is a chance that more than one log message can have the same time stamp, you can display messages with sequence numbers so that you can unambiguously see a single message. By default, sequence numbers in log messages are not displayed.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**service sequence-numbers**	Enables sequence numbers.
3.	**end**	Returns to privileged EXEC mode.

Defining the Message Severity Level

You can limit messages displayed to the selected device by specifying the severity level of the message, which are described in Table 2.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**logging console** *level*	Limits messages logged to the console. By default, the console receives debugging messages and numerically lower levels.
3.	**logging monitor** *level*	Limits messages logged to the terminal lines. By default, the terminal receives debugging messages and numerically lower levels.
4.	**logging trap** *level*	Limits messages logged to the syslog servers. By default, syslog servers receive informational messages and numerically lower levels.
5.	**end**	Returns to privileged EXEC mode.

Limiting Syslog Messages Sent to the History Table and to SNMP

If you enabled syslog message traps to be sent to an SNMP network management station by using the **snmp-server enable trap** global configuration command, you can change the level of messages sent and stored in the switch history table. You also can change the number of messages that are stored in the history table.

Messages are stored in the history table because SNMP traps are not guaranteed to reach their destination. By default, one message of the level **warning** and numerically lower levels are stored in the history table even if syslog traps are not enabled.

When the history table is full (it contains the maximum number of message entries specified with the **logging history size** global configuration command), the oldest message entry is deleted from the table to allow the new message entry to be stored.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**logging history** *level*	Changes the default level of syslog messages stored in the history file and sent to the SNMP server. By default, **warnings**, **errors**, **critical**, **alerts**, and **emergencies** messages are sent.
3.	**logging history size** *number*	Specifies the number of syslog messages that can be stored in the history table. The default is to store one message. The range is 0 to 500 messages.
4.	**end**	Returns to privileged EXEC mode.

Enabling the Configuration-Change Logger

You can enable a configuration logger to keep track of configuration changes made with the command-line interface (CLI). When you enter the **logging enable** configuration-change logger configuration command, the log records the session, the user, and the command that was entered to change the configuration. You can configure the size of the configuration log from 1 to 1000 entries (the default is 100).

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**archive**	Enters archive configuration mode.
3.	**log config**	Enters configuration-change logger configuration mode.
4.	**logging enable**	Enables configuration change logging.
5.	**logging size** *entries*	(Optional) Configures the number of entries retained in the configuration log. The range is from 1 to 1000. The default is 100. **Note:** When the configuration log is full, the oldest log entry is removed each time a new entry is entered.
6.	**end**	Returns to privileged EXEC mode.

Configuring the UNIX System Logging Facility

When sending system log messages to an external device, you can cause the switch to identify its messages as originating from any of the UNIX syslog facilities.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**logging** *host*	Logs messages to a UNIX syslog server host by entering its IP address. To build a list of syslog servers that receive logging messages, enter this command more than once.
3.	**logging trap** *level*	Limits messages logged to the syslog servers. Be default, syslog servers receive informational messages and lower.
4.	**logging facility** *facility-type*	Configures the syslog facility. The default is **local7**.
5.	**end**	Returns to privileged EXEC mode.

Monitoring and Maintaining the System Message Log

Command	Purpose
show logging	Displays logging messages.
show archive log config	Displays the configuration log.

Configuration Examples for the System Message Log

System Message: Example

This example shows a partial switch system message:

```
00:00:46: %LINK-3-UPDOWN: Interface Port-channel1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet1/17, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet1/18, changed state to up
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet1/17, changed state to down 2
*Mar  1 18:46:11: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
18:47:02: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
*Mar  1 18:48:50.483 UTC: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
```

Logging Display: Examples

This example shows part of a logging display with the **service timestamps log datetime** global configuration command enabled:

```
*Mar  1 18:46:11: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
```

This example shows part of a logging display with the s**ervice timestamps log uptime** global configuration command enabled:

```
00:00:46: %LINK-3-UPDOWN: Interface Port-channel1, changed state to up
```

This example shows part of a logging display with sequence numbers enabled:

```
000019: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
```

Enabling the Logger: Example

This example shows how to enable the configuration-change logger and to set the number of entries in the log to 500.

```
Switch(config)# archive
Switch(config-archive)# log config
Switch(config-archive-log-cfg)# logging enable
Switch(config-archive-log-cfg)# logging size 500
Switch(config-archive-log-cfg)# end
```

Configuration Log Output: Example

This is an example of output for the configuration log:

```
Switch# show archive log config all
 idx   sess           user@line      Logged command
   38    11   unknown user@vty3     |no aaa authorization config-commands
   39    12   unknown user@vty3     |no aaa authorization network default group radius
   40    12   unknown user@vty3     |no aaa accounting dot1x default start-stop group radius
   41    13   unknown user@vty3     |no aaa accounting system default
   42    14           temi@vty4     |interface GigabitEthernet4/0/1
   43    14           temi@vty4     | switchport mode trunk
   44    14           temi@vty4     | exit
   45    16           temi@vty5     |interface FastEthernet5/0/1
   46    16           temi@vty5     | switchport mode trunk
   47    16           temi@vty5     | exit
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands Cisco IOS system management commands	*Cisco IOS Configuration Fundamentals Command Reference*
Syslog server configuration steps	Configuring the UNIX System Logging Facility, page 553

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Configuring SNMP

Prerequisites for SNMP

An SNMP *group* is a table that maps SNMP users to SNMP views. An SNMP *user* is a member of an SNMP group. An SNMP *host* is the recipient of an SNMP trap operation. An SNMP *engine ID* is a name for the local or remote SNMP engine.

- If the switch starts and the switch startup configuration has at least one **snmp-server** global configuration command, the SNMP agent is enabled.
- When configuring an SNMP group, do not specify a notify view. The **snmp-server host** global configuration command autogenerates a notify view for the user and then adds it to the group associated with that user. Modifying the group's notify view affects all users associated with that group. See the *Cisco IOS Network Management Command Reference* for information about when you should configure notify views.
- To configure a remote user, specify the IP address or port number for the remote SNMP agent of the device where the user resides.
- Before you configure remote users for a particular agent, configure the SNMP engine ID, using the **snmp-server engineID** global configuration with the **remote** option. The remote agent's SNMP engine ID and user password are used to compute the authentication and privacy digests. If you do not configure the remote engine ID first, the configuration command fails.

Restrictions for SNMP

- When configuring SNMP informs, you need to configure the SNMP engine ID for the remote agent in the SNMP database before you can send proxy requests or informs to it.
- If a local user is not associated with a remote host, the switch does not send informs for the **auth** (authNoPriv) and the **priv** (authPriv) authentication levels.
- Changing the value of the SNMP engine ID has important implications. A user's password (entered on the command line) is converted to an MD5 or SHA security digest based on the password and the local engine ID. The command-line password is then destroyed, as required by RFC 2274. Because of this deletion, if the value of the engine ID changes, the security digests of SNMPv3 users become invalid, and you need to reconfigure SNMP users by using the **snmp-server user** *username* global configuration command. Similar restrictions require the reconfiguration of community strings when the engine ID changes.

Information About SNMP

SNMP

The Simple Network Management Protocol (SNMP) is an application-layer protocol that provides a message format for communication between managers and agents. The SNMP system consists of an SNMP manager, an SNMP agent, and a MIB. The SNMP manager can be part of a network management system (NMS) such as CiscoWorks. The agent and MIB reside on the switch. To configure SNMP on the switch, you define the relationship between the manager and the agent.

The SNMP agent contains MIB variables whose values the SNMP manager can request or change. A manager can get a value from an agent or store a value into the agent. The agent gathers data from the MIB, the repository for information about device parameters and network data. The agent can also respond to a manager's requests to get or set data.

An agent can send unsolicited traps to the manager. Traps are messages alerting the SNMP manager to a condition on the network. Traps can mean improper user authentication, restarts, link status (up or down), MAC address tracking, closing of a TCP connection, loss of connection to a neighbor, or other significant events.

SNMP Versions

This software release supports these SNMP versions:

- SNMPv1—The Simple Network Management Protocol, a Full Internet Standard, defined in RFC 1157.
- SNMPv2C replaces the Party-based Administrative and Security Framework of SNMPv2Classic with the community-string-based Administrative Framework of SNMPv2C while retaining the bulk retrieval and improved error handling of SNMPv2Classic. It has these features:
  - SNMPv2—Version 2 of the Simple Network Management Protocol, a Draft Internet Standard, defined in RFCs 1902 through 1907.
  - SNMPv2C—The community-string-based Administrative Framework for SNMPv2, an Experimental Internet Protocol defined in RFC 1901.
- SNMPv3—Version 3 of the SNMP is an interoperable standards-based protocol defined in RFCs 2273 to 2275. SNMPv3 provides secure access to devices by authenticating and encrypting packets over the network and includes these security features:
  - Message integrity—Ensures that a packet was not tampered with in transit.
  - Authentication—Determines that the message is from a valid source.
  - Encryption—Mixes the contents of a package to prevent it from being read by an unauthorized source.

  To select encryption, enter the **priv** keyword. This keyword is available only when the cryptographic (encrypted) software image is installed.

Both SNMPv1 and SNMPv2C use a community-based form of security. The community of managers able to access the agent's MIB is defined by an IP address access control list and password.

SNMPv2C includes a bulk retrieval mechanism and more detailed error message reporting to management stations. The bulk retrieval mechanism retrieves tables and large quantities of information, minimizing the number of round-trips required. The SNMPv2C improved error-handling includes expanded error codes that distinguish different kinds of error conditions; these conditions are reported through a single error code in SNMPv1. Error return codes in SNMPv2C report the error type.

SNMPv3 provides for both security models and security levels. A security model is an authentication strategy set up for a user and the group within which the user resides. A security level is the permitted level of security within a security model. A combination of the security level and the security model determine which security mechanism is used when handling an SNMP packet. Available security models are SNMPv1, SNMPv2C, and SNMPv3.

Table 52 on page 559 identifies the characteristics of the different combinations of security models and levels.

Table 52 SNMP Security Models and Levels

Model	Level	Authentication	Encryption	Result
SNMPv1	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
SNMPv2C	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
SNMPv3	noAuthNoPriv	Username	No	Uses a username match for authentication.
SNMPv3	authNoPriv	Message Digest 5 (MD5) or Secure Hash Algorithm (SHA)	No	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms.
SNMPv3	authPriv	MD5 or SHA	Data Encryption Standard (DES) or Advanced Encryption Standard (AES)	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. Allows specifying the User-based Security Model (USM) with these encryption algorithms: ■ DES 56-bit encryption in addition to authentication based on the CBC-DES (DES-56) standard. ■ 3DES 168-bit encryption ■ AES 128-bit, 192-bit, or 256-bit encryption

You must configure the SNMP agent to use the SNMP version supported by the management station. Because an agent can communicate with multiple managers, you can configure the software to support communications using SNMPv1, SNMPv2C, or SNMPv3.

SNMP Manager Functions

The SNMP manager uses information in the MIB to perform the operations described in Table 53 on page 559.

Table 53 SNMP Operations

Operation	Description
get-request	Retrieves a value from a specific variable.
get-next-request	Retrieves a value from a variable within a table.[1]
get-bulk-request[2]	Retrieves large blocks of data, such as multiple rows in a table, that would otherwise require the transmission of many small blocks of data.
get-response	Replies to a get-request, get-next-request, and set-request sent by an NMS.
set-request	Stores a value in a specific variable.
trap	An unsolicited message sent by an SNMP agent to an SNMP manager when some event has occurred.

1. With this operation, an SNMP manager does not need to know the exact variable name. A sequential search is performed to find the needed variable from within a table.

2. The **get-bulk** command only works with SNMPv2 or later.

SNMP Agent Functions

The SNMP agent responds to SNMP manager requests as follows:

- Get a MIB variable—The SNMP agent begins this function 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.
- Set a MIB variable—The SNMP agent begins this function in response to a message from the NMS. The SNMP agent changes the value of the MIB variable to the value requested by the NMS.

The SNMP agent also sends unsolicited trap messages to notify an NMS that a significant event has occurred on the agent. Examples of trap conditions include, but are not limited to, when a port or module goes up or down, when spanning-tree topology changes occur, and when authentication failures occur.

SNMP Community Strings

SNMP community strings authenticate access to MIB objects and function as embedded passwords. In order for the NMS to access the switch, the community string definitions on the NMS must match at least one of the three community string definitions on the switch.

A community string can have one of these attributes:

- Read-only (RO)—Gives read access to authorized management stations to all objects in the MIB except the community strings, but does not allow write access.
- Read-write (RW)—Gives read and write access to authorized management stations to all objects in the MIB, but does not allow access to the community strings.

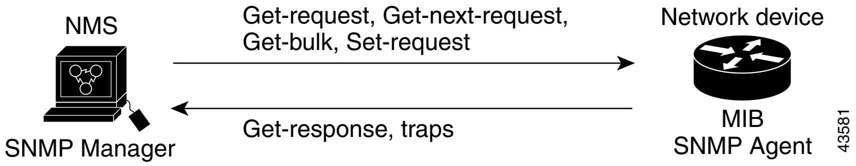
When a cluster is created, the command switch manages the exchange of messages among member switches and the SNMP application.

Using SNMP to Access MIB Variables

An example of an NMS is the CiscoWorks network management software. CiscoWorks 2000 software uses the switch MIB variables to set device variables and to poll devices on the network for specific information. The results of a poll can be displayed as a graph and analyzed to troubleshoot internetworking problems, increase network performance, verify the configuration of devices, monitor traffic loads, and more.

As shown in Figure 73 on page 561, the SNMP agent gathers data from the MIB. The agent can send traps, or notification of certain events, to the SNMP manager, which receives and processes the traps. Traps alert the SNMP manager to a condition on the network such as improper user authentication, restarts, link status (up or down), MAC address tracking, and so forth. The SNMP agent also responds to MIB-related queries sent by the SNMP manager in *get-request*, *get-next-request*, and *set-request* format.

Figure 73 SNMP Network



SNMP Notifications

SNMP allows the switch to send notifications to SNMP managers when particular events occur. SNMP notifications can be sent as traps or inform requests. In command syntax, unless there is an option in the command to select either traps or informs, the keyword **traps** refers to either traps or informs, or both. Use the **snmp-server host** command to specify whether to send SNMP notifications as traps or informs.

Note: SNMPv1 does not support informs.

Traps are unreliable because the receiver does not send an acknowledgment when it receives a trap, and the sender cannot determine if the trap was received. When an SNMP manager receives an inform request, it acknowledges the message with an SNMP response protocol data unit (PDU). If the sender does not receive a response, the inform request can be sent again. Because they can be resent, informs are more likely than traps to reach their intended destination.

The characteristics that make informs more reliable than traps also consume more resources in the switch and in the network. Unlike a trap, which is discarded as soon as it is sent, an inform request is held in memory until a response is received or the request times out. Traps are sent only once, but an inform might be resent or retried several times. The retries increase traffic and contribute to a higher overhead on the network. Therefore, traps and informs require a trade-off between reliability and resources. If it is important that the SNMP manager receive every notification, use inform requests. If traffic on the network or memory in the switch is a concern and notification is not required, use traps.

SNMP ifIndex MIB Object Values

In an NMS, the IF-MIB generates and assigns an interface index (ifIndex) object value that is a unique number greater than zero to identify a physical or a logical interface. When the switch reboots or the switch software is upgraded, the switch uses this same value for the interface. For example, if the switch assigns a port 2 an ifIndex value of 10003, this value is the same after the switch reboots.

The switch uses one of the values in Table 54 on page 561 to assign an ifIndex value to an interface.

Table 54 ifIndex MIB Object Values

Interface Type	ifIndex Range
SVI	1–4999
EtherChannel	5001–5048
Physical (such as Gigabit Ethernet or SFP-module interfaces) based on type and port numbers	10000–14500
Null	10501
Loopback and Tunnel	24567 +

Note: The switch might not use sequential values within a range.

Community Strings

You use the SNMP community string to define the relationship between the SNMP manager and the agent. The community string acts like a password to permit access to the agent on the switch. Optionally, you can specify one or more of these characteristics associated with the string:

- An access list of IP addresses of the SNMP managers that are permitted to use the community string to gain access to the agent
- A MIB view, which defines the subset of all MIB objects accessible to the given community
- Read and write or read-only permission for the MIB objects accessible to the community

SNMP Notifications

A trap manager is a management station that receives and processes traps. Traps are system alerts that the switch generates when certain events occur. By default, no trap manager is defined, and no traps are sent. Switches running this Cisco IOS release can have an unlimited number of trap managers.

Note: Many commands use the word *traps* in the command syntax. Unless there is an option in the command to select either traps or informs, the keyword ***traps*** refers to traps, informs, or both. Use the **snmp-server host** global configuration command to specify whether to send SNMP notifications as traps or informs.

This table describes the supported switch traps (notification types). You can enable any or all of these traps and configure a trap manager to receive them. To enable the sending of SNMP inform notifications, use the **snmp-server enable traps** global configuration command combined with the **snmp-server host** *host-addr* **informs** global configuration command.

Table 55 Switch Notification Types

Notification Type Keyword	Description
bridge	Generates STP bridge MIB traps.
config	Generates a trap for SNMP configuration changes.
copy-config	Generates a trap for SNMP copy configuration changes.
entity	Generates a trap for SNMP entity changes.
cpu threshold	Allows CPU-related traps.
envmon	Generates environmental monitor traps. You can enable any or all of these environmental traps: fan, shutdown, status, supply, temperature.
errdisable	Generates a trap for an error-disabled VLAN port. You can also set a maximum trap rate per minute. The range is from 0 to 10000; the default is 0, which means there is no rate limit.
flash	Generates SNMP FLASH notifications.
hsrp	Generates a trap for Hot Standby Router Protocol (HSRP) changes.
ipmulticast	Generates a trap for IP multicast routing changes.
mac-notification	Generates a trap for MAC address notifications.
msdp	Generates a trap for Multicast Source Discovery Protocol (MSDP) changes.
ospf	Generates a trap for Open Shortest Path First (OSPF) changes. You can enable any or all of these traps: Cisco specific, errors, link-state advertisement, rate limit, retransmit, and state changes.
pim	Generates a trap for Protocol-Independent Multicast (PIM) changes. You can enable any or all of these traps: invalid PIM messages, neighbor changes, and rendezvous point (RP)-mapping changes.

Table 55 Switch Notification Types (continued)

Notification Type Keyword	Description
port-security	Generates SNMP port security traps. You can also set a maximum trap rate per second. The range is from 0 to 1000; the default is 0, which means that there is no rate limit. **Note:** When you configure a trap by using the notification type **port-security**, configure the port security trap first, and then configure the port security trap rate: ■ **snmp-server enable traps port-security** ■ **snmp-server enable traps port-security trap-rate** *rate*
rtr	Generates a trap for the SNMP Response Time Reporter (RTR).
snmp	Generates a trap for SNMP-type notifications for authentication, cold start, warm start, link up or link down.
storm-control	Generates a trap for SNMP storm control. You can also set a maximum trap rate per minute. The range is from 0 to 1000; the default is 0 (no limit is imposed; a trap is sent at every occurrence).
stpx	Generates SNMP STP Extended MIB traps.
syslog	Generates SNMP syslog traps.
tty	Generates a trap for TCP connections. This trap is enabled by default.
vlan-membership	Generates a trap for SNMP VLAN membership changes.
vlancreate	Generates SNMP VLAN created traps.
vlandelete	Generates SNMP VLAN deleted traps.
vtp	Generates a trap for VLAN Trunking Protocol (VTP) changes.

Note: Though visible in the command-line help strings, the **fru-ctrl, insertion**, and **removal** keywords are not supported.

You can use the **snmp-server host** global configuration command to a specific host to receive the notification types listed in Table 55 on page 562.

Default SNMP Settings

Feature	Default Setting
SNMP agent	Disabled[1].
SNMP trap receiver	None configured.
SNMP traps	None enabled except the trap for TCP connections (**tty**).
SNMP version	If no **version** keyword is present, the default is Version 1.
SNMPv3 authentication	If no keyword is entered, the default is the **noauth** (noAuthNoPriv) security level.
SNMP notification type	If no type is specified, all notifications are sent.

[1] This is the default when the switch starts and the startup configuration does not have any **snmp-server** global configuration commands.

How to Configure SNMP

Disabling the SNMP Agent

The **no snmp-server** global configuration command disables all running versions (Version 1, Version 2C, and Version 3) on the device. No specific Cisco IOS command exists to enable SNMP. The first **snmp-server** global configuration command that you enter enables all versions of SNMP.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**no snmp-server**	Disables the SNMP agent operation.
3.	**end**	Returns to privileged EXEC mode.

Configuring Community Strings

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).

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**snmp-server community** *string* [**view** *view-name*] [**ro** | **rw**] [*access-list-number*]	Configures the community string. **Note:** The @ symbol is used for delimiting the context information. Avoid using the @ symbol as part of the SNMP community string when configuring this command. ■ *string*–Specifies a string that acts like a password and permits access to the SNMP protocol. You can configure one or more community strings of any length. ■ (Optional) **view**–Specifies the view record accessible to the community. ■ (Optional) Specifies either read-only (**ro**) if you want authorized management stations to retrieve MIB objects, or specifies read-write (**rw**) if you want authorized management stations to retrieve and modify MIB objects. By default, the community string permits read-only access to all objects. ■ (Optional) *access-list-number*–Specifies an IP standard access list numbered from 1 to 99 and 1300 to 1999.
3.	**access-list** *access-list-number* {**deny** / **permit**} *source* [*source-wildcard*]	(Optional) If you specified an IP standard access list number in Step 2, then create the list, repeating the command as many times as necessary. ■ *access-list-number*–Specifies the access list number specified in Step 2. ■ **deny** – Denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. ■ *source*–Specifies the IP address of the SNMP managers that are permitted to use the community string to gain access to the agent. ■ (Optional) *source-wildcard*–Specifies the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. The access list is always terminated by an implicit deny statement for everything.
4.	**end**	Returns to privileged EXEC mode.

Configuring SNMP Groups and Users

You can specify an identification name (engine ID) for the local or remote SNMP server engine on the switch. You can configure an SNMP server group that maps SNMP users to SNMP views, and you can add new users to the SNMP group.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**snmp-server engineID** {**local** *engineid-string* | **remote** *ip-address* [**udp-port** *port-number*] *engineid-string*}	Configures a name for either the local or remote copy of SNMP. ■ The *engineid-string* is a 24-character ID string with the name of the copy of SNMP. You need not specify the entire 24-character engine ID if it has trailing zeros. Specify only the portion of the engine ID up to the point where only zeros remain in the value. For example, to configure an engine ID of 123400000000000000000000, you can enter this: **snmp-server engineID local 1234** ■ If you select **remote**, specify the *ip-address* of the device that contains the remote copy of SNMP and the optional User Datagram Protocol (UDP) port on the remote device. The default is 162.

	Command	Purpose
3.	**snmp-server group** *groupname* {**v1** / **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*]	Configures a new SNMP group on the remote device. ■ *groupname*–Specifies the name of the group. ■ Specify a security model: – **v1** is the least secure of the possible security models. – **v2c** is the second least secure model. It allows transmission of informs and integers twice the normal width. – **v3,** the most secure, requires you to select an authentication level: **auth**–Enables the Message Digest 5 (MD5) and the Secure Hash Algorithm (SHA) packet authentication. **noauth**–Enables the noAuthNoPriv security level. This is the default if no keyword is specified. **priv**–Enables Data Encryption Standard (DES) packet encryption (also called *privacy*). **Note:** The **priv** keyword is available only when the cryptographic software image is installed. ■ (Optional) **read** *readview*–Specifies a string (not to exceed 64 characters) that is the name of the view in which you can only view the contents of the agent. ■ (Optional) **write** *writeview*–Specifies a string (not to exceed 64 characters) that is the name of the view in which you enter data and configure the contents of the agent. ■ (Optional) **notify** *notifyview*–Specifies a string (not to exceed 64 characters) that is the name of the view in which you specify a notify, inform, or trap. ■ (Optional) **access** *access-list*–Specifies a string (not to exceed 64 characters) that is the name of the access list.

	Command	Purpose
4.	**snmp-server user** *username groupname* {**remote** *host* [**udp-port** *port*]} {**v1** [**access** *access-list*] / **v2c** [**access** *access-list*] **| v3** [**encrypted**] [**access** *access-list*] [**auth** {**md5** | **sha**} *auth-password*]} [*priv* {**des** | **3des** | **aes** {**128** | **192** | **256**}} *priv-password*]	Adds a new user for an SNMP group. ■ *username*–Specifies a name of the user on the host that connects to the agent. ■ *groupname*–Specifies a name of the group to which the user is associated. ■ **remote**–Specifies a remote SNMP entity to which the user belongs and the hostname or IP address of that entity with the optional UDP port number. The default is 162. ■ Enters the SNMP version number (**v1**, **v2c**, or **v3**). If you enter **v3**, you have these additional options: – **encrypted**–Specifies that the password appears in encrypted format. This keyword is available only when the **v3** keyword is specified. – **auth**–Specifies an authentication level setting session that can be either the HMAC-MD5-96 (**md5**) or the HMAC-SHA-96 (**sha**) authentication level and requires a password string *auth-password* (not to exceed 64 characters). ■ If you enter **v3** and the switch is running the cryptographic software image, you can also configure a private (**priv**) encryption algorithm and password string *priv-password* (not to exceed 64 characters). – **priv**–Specifies the User-based Security Model (USM). – **des**–Specifies the use of the 56-bit DES algorithm. – **3des**–Specifies the use of the 168-bit DES algorithm. – **aes**–Specifies the use of the DES algorithm. You must select either 128-bit, 192-bit, or 256-bit encryption. ■ (Optional) Enters **access** *access-list* with a string (not to exceed 64 characters) that is the name of the access list.
5.	**end**	Returns to privileged EXEC mode.

Configuring SNMP Notifications

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**snmp-server engineID remote** *ip-address engineid-string*	Specifies the engine ID for the remote host.
3.	**snmp-server user** *username groupname* {**remote** *host* [**udp-port** *port*]} {**v1** [**access** *access-list*] / **v2c** [**access** *access-list*] **| v3** [**encrypted**] [**access** *access-list*] [**auth** {**md5** | **sha**} *auth-password*]}	Configures an SNMP user to be associated with the remote host created in Step 2. **Note:** You cannot configure a remote user for an address without first configuring the engine ID for the remote host. Otherwise, you receive an error message, and the command is not executed.

	Command	Purpose
4.	**snmp-server group** *groupname* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*]	Configures an SNMP group.
5.	**snmp-server host** *host-addr* [**informs** | **traps**] [**version** {**1** / **2c** | **3** {**auth** | **noauth** | **priv**}}] *community-string* [*notification-type*]	Specifies the recipient of an SNMP trap operation. ■ *host-addr*–Specifies the name or Internet address of the host (the targeted recipient). ■ (Optional) **informs**–Specifies SNMP informs to be sent to the host. ■ (Optional) **traps** (the default)–Specifies SNMP traps to be sent to the host. ■ (Optional) Specifies the SNMP **version** (**1**, **2c**, or **3**). SNMPv1 does not support informs. ■ (Optional) Version 3–Selects authentication level **auth, noauth**, or **priv**. **Note:** The **priv** keyword is available only when the cryptographic software image is installed. ■ *community-string*–When **version 1** or **version 2c** is specified, enters the password-like community string sent with the notification operation. When **version 3** is specified, enter the SNMPv3 username. **Note:** The @ symbol is used for delimiting the context information. Avoid using the @ symbol as part of the SNMP community string when configuring this command. ■ (Optional) *notification-type*–Specifies a notification type. Use the keywords listed in Table 55 on page 562. If no type is specified, all notifications are sent.
6.	**snmp-server enable traps** *notification-types*	Enables the switch to send traps or informs and specifies the type of notifications to be sent. For a list of notification types, see Table 55 on page 562, or enter **snmp-server enable traps ?** To enable multiple types of traps, you must enter a separate **snmp-server enable traps** command for each trap type. **Note:** When you configure a trap by using the notification type **port-security**, configure the port security trap first, and then configure the port security trap rate: ■ **snmp-server enable traps port-security** ■ **snmp-server enable traps port-security trap-rate** *rate*
7.	**snmp-server trap-source** *interface-id*	(Optional) Specifies the source interface, which provides the IP address for the trap message. This command also sets the source IP address for informs.

	Command	Purpose
8.	**snmp-server queue-length** *length*	(Optional) Establishes the message queue length for each trap host. The range is 1 to 1000; the default is 10.
9.	**snmp-server trap-timeout** *seconds*	(Optional) Defines how often to resend trap messages. The range is 1 to 1000; the default is 30 seconds.
10.	**end**	Returns to privileged EXEC mode.

Setting the CPU Threshold Notification Types and Values

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**process cpu threshold type** {**total** | **process** | **interrupt**} **rising** *percentage* **interval** *seconds* [**falling** *fall-percentage* **interval** *seconds*]	Sets the CPU threshold notification types and values: ■ **total**–Sets the notification type to total CPU utilization. ■ **process**–Sets the notification type to CPU process utilization. ■ **interrupt**–Sets the notification type to CPU interrupt utilization. ■ **rising** *percentage*–Specifies the percentage (1 to 100) of CPU resources that, when exceeded for the configured interval, sends a CPU threshold notification. ■ **interval** *seconds*–Specifies the duration of the CPU threshold violation in seconds (5 to 86400) that, when met, sends a CPU threshold notification. ■ **falling** *fall-percentage*–Specifies the percentage (1 to 100) of CPU resources that, when usage falls below this level for the configured interval, sends a CPU threshold notification. This value must be equal to or less than the **rising** *percentage* value. If not specified, the **falling** *fall-percentage* value is the same as the **rising** *percentage* value.
3.	**end**	Returns to privileged EXEC mode.

Setting the Agent Contact and Location Information

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**snmp-server contact** *text*	Sets the system contact string.
3.	**snmp-server location** *text*	Sets the system location string.
4.	**end**	Returns to privileged EXEC mode.

Limiting TFTP Servers Used Through SNMP

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**snmp-server tftp-server-list** *access-list-number*	Limits TFTP servers used for configuration file copies through SNMP to the servers in the access list. *access-list-number*–Enters an IP standard access list numbered from 1 to 99 and 1300 to 1999.
3.	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]	Creates a standard access list, repeating the command as many times as necessary. ■ *access-list-number*–Enters the access list number specified in Step 2. ■ **deny**–Denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. ■ *source*–Enters the IP address of the TFTP servers that can access the switch. ■ (Optional) *source-wildcard*–Enters the wildcard bits, in dotted decimal notation, to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
4.	**end**	Returns to privileged EXEC mode.

Monitoring and Maintaining SNMP

Command	Purpose
show snmp	Displays SNMP statistics.
show snmp engineID [**local | remote**]	Displays information on the local SNMP engine and all remote engines that have been configured on the device.
show snmp group	Displays information on each SNMP group on the network.
show snmp pending	Displays information on pending SNMP requests.
show snmp sessions	Displays information on the current SNMP sessions.
show snmp user	Displays information on each SNMP user name in the SNMP users table. **Note:** You must use this command to display SNMPv3 configuration information for **auth** | **noauth** | **priv** mode. This information is not displayed in the **show running-config** output.

Configuration Examples for SNMP

Enabling SNMP Versions: Example

This example shows how to enable all versions of SNMP. The configuration permits any SNMP manager to access all objects with read-only permissions using the community string *public*. This configuration does not cause the switch to send any traps.

```
Switch(config)# snmp-server community public
```

Permit SNMP Manager Access: Example

This example shows how to permit any SNMP manager to access all objects with read-only permission using the community string *public*. The switch also sends VTP traps to the hosts 192.180.1.111 and 192.180.1.33 using SNMPv1 and to the host 192.180.1.27 using SNMPv2C. The community string *public* is sent with the traps.

```
Switch(config)# snmp-server community public
Switch(config)# snmp-server enable traps vtp
Switch(config)# snmp-server host 192.180.1.27 version 2c public
Switch(config)# snmp-server host 192.180.1.111 version 1 public
Switch(config)# snmp-server host 192.180.1.33 public
```

Allow Read-Only Access: Example

This example shows how to allow read-only access for all objects to members of access list 4 that use the *comaccess* community string. No other SNMP managers have access to any objects. SNMP Authentication Failure traps are sent by SNMPv2C to the host *cisco.com* using the community string *public*.

```
Switch(config)# snmp-server community comaccess ro 4
Switch(config)# snmp-server enable traps snmp authentication
Switch(config)# snmp-server host cisco.com version 2c public
```

Configure SNMP Traps: Examples

This example shows how to send entity MIB traps to the host *cisco.com*. The community string is restricted. The first line enables the switch to send entity MIB traps in addition to any traps previously enabled. The second line specifies the destination of these traps and overwrites any previous **snmp-server host** commands for the host *cisco.com*.

```
Switch(config)# snmp-server enable traps entity
Switch(config)# snmp-server host cisco.com restricted entity
```

This example shows how to enable the switch to send all traps to the host *myhost.cisco.com* using the community string *public*:

```
Switch(config)# snmp-server enable traps
Switch(config)# snmp-server host myhost.cisco.com public
```

Associating a User with a Remote Host: Example

This example shows how to associate a user with a remote host and to send **auth** (authNoPriv) authentication-level informs when the user enters global configuration mode:

```
Switch(config)# snmp-server engineID remote 192.180.1.27 00000063000100a1c0b4011b
Switch(config)# snmp-server group authgroup v3 auth
Switch(config)# snmp-server user authuser authgroup remote 192.180.1.27 v3 auth md5 mypassword
Switch(config)# snmp-server user authuser authgroup v3 auth md5 mypassword
Switch(config)# snmp-server host 192.180.1.27 informs version 3 auth authuser config
Switch(config)# snmp-server enable traps
Switch(config)# snmp-server inform retries 0
```

Assigning a String to SNMP: Example

This example shows how to assign the string *comaccess* to SNMP, to allow read-only access, and to specify that IP access list 4 can use the community string to gain access to the switch SNMP agent:

```
Switch(config)# snmp-server community comaccess ro 4
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*
Cisco IOS SNMP syntax and usage	*Cisco IOS Network Management Command Reference*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Configuring Network Security with ACLs

This chapter describes how to configure network security on the IE 4000, IE 4010 and IE5000 switches by using access control lists (ACLs), also referred to as access lists. In this chapter, references to IP ACLs are specific to IP Version 4 (IPv4) ACLs, Port ACLs, VLAN ACLs and VLAN Maps.

Understanding ACLs

Packet filtering can help limit network traffic and restrict network use by certain users or devices. ACLs filter traffic as it passes through a router or switch and permit or deny packets crossing specified interfaces or VLANs. An ACL is a sequential collection of permit and deny conditions that apply to packets. When a packet is received on an interface, the switch compares the fields in the packet against any applied ACLs to verify that the packet has the required permissions to be forwarded, based on the criteria specified in the access lists. One by one, it tests packets against the conditions in an access list. The first match decides whether the switch accepts or rejects the packets. Because the switch stops testing after the first match, the order of conditions in the list is critical. If no conditions match, the switch rejects the packet. If there are no restrictions, the switch forwards the packet; otherwise, the switch drops the packet. The switch can use ACLs on all packets it forwards, including packets bridged within a VLAN.

You configure access lists on a router or Layer 3 switch to provide basic security for your network. If you do not configure ACLs, all packets passing through the switch could be allowed onto all parts of the network. You can use ACLs to control which hosts can access different parts of a network or to decide which types of traffic are forwarded or blocked at router interfaces. For example, you can allow e-mail traffic to be forwarded but not Telnet traffic. ACLs can be configured to block inbound traffic, outbound traffic, or both.

An ACL contains an ordered list of access control entries (ACEs). Each ACE specifies *permit* or *deny* and a set of conditions the packet must satisfy in order to match the ACE. The meaning of *permit* or *deny* depends on the context in which the ACL is used.

The switch supports IP ACLs and Ethernet (MAC) ACLs:

- IP ACLs filter IPv4 traffic, including TCP, User Datagram Protocol (UDP), Internet Group Management Protocol (IGMP), and Internet Control Message Protocol (ICMP).
- Ethernet ACLs filter non-IP traffic.

This switch also supports quality of service (QoS) classification ACLs. For more information, see Classification Based on QoS ACLs, page 622.

These sections contain this conceptual information:

- Supported ACLs, page 576
- Handling Fragmented and Unfragmented Traffic, page 578

Supported ACLs

Note: Router ACLs and VLAN maps are supported only on switches running the IP services image.

Port ACLs access-control traffic entering a Layer 2 interface. The switch does not support port ACLs in the outbound direction. You can apply only one IP access list and one MAC access list to a Layer 2 interface. For more information, see Port ACLs, page 576.

If IEEE 802.1Q tunneling is configured on an interface, any IEEE 802.1Q encapsulated IP packets received on the tunnel port can be filtered by MAC ACLs, but not by IP ACLs. This is because the switch does not recognize the protocol inside the IEEE 802.1Q header. This restriction applies to router ACLs and port ACLs.

Port ACLs

Port ACLs are ACLs that are applied to Layer 2 interfaces on a switch. Port ACLs are supported only on physical interfaces and not on EtherChannel interfaces and can be applied only on interfaces in the inbound direction. These access lists are supported:

- Standard IP access lists using source addresses
- Extended IP access lists using source and destination addresses and optional protocol type information
- MAC extended access lists using source and destination MAC addresses and optional protocol type information

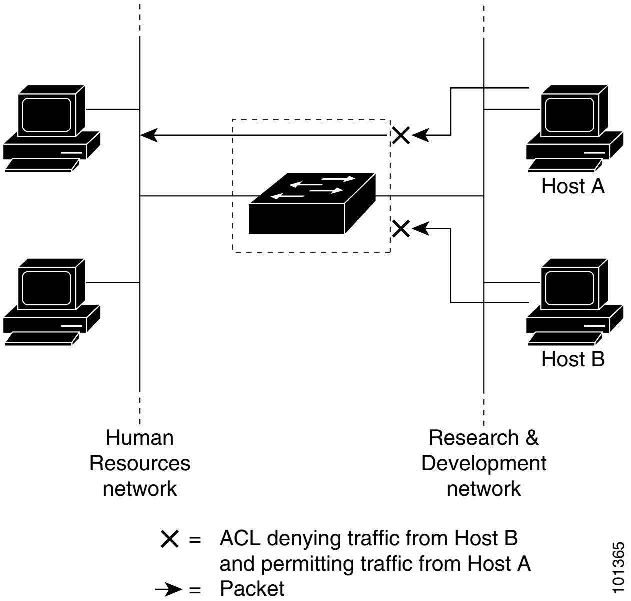
The switch examines ACLs associated with all inbound features configured on a given interface and permits or denies packet forwarding based on how the packet matches the entries in the ACL. In this way, ACLs control access to a network or to part of a network. Figure 74 on page 577 is an example of using port ACLs to control access to a network when all workstations are in the same VLAN. ACLs applied at the Layer 2 input would allow Host A to access the Human Resources network, but prevent Host B from accessing the same network. Port ACLs can only be applied to Layer 2 interfaces in the inbound direction.

When you apply a port ACL to a trunk port, the ACL filters traffic on all VLANs present on the trunk port. When you apply a port ACL to a port with voice VLAN, the ACL filters traffic on both data and voice VLANs.

With port ACLs, you can filter IP traffic by using IP access lists and non-IP traffic by using MAC addresses. You can filter both IP and non-IP traffic on the same Layer 2 interface by applying both an IP access list and a MAC access list to the interface.

Note: You cannot apply more than one IP access list and one MAC access list to a Layer 2 interface. If an IP access list or MAC access list is already configured on a Layer 2 interface and you apply a new IP access list or MAC access list to the interface, the new ACL replaces the previously configured one.

Figure 74 Using ACLs to Control Traffic to a Network


Host A
Host B
Human Resources network
Research & Development network
X = ACL denying traffic from Host B and permitting traffic from Host A
→ = Packet
101365

When you apply a port ACL to a trunk port, the ACL filters traffic on all VLANs present on the trunk port. When you apply a port ACL to a port with voice VLAN, the ACL filters traffic on both data and voice VLANs.

With port ACLs, you can filter IP traffic by using IP access lists and non-IP traffic by using MAC addresses. You can filter both IP and non-IP traffic on the same Layer 2 interface by applying both an IP access list and a MAC access list to the interface.

Note: You cannot apply more than one IP access list and one MAC access list to a Layer 2 interface. If an IP access list or MAC access list is already configured on a Layer 2 interface and you apply a new IP access list or MAC access list to the interface, the new ACL replaces the previously configured one.

Router ACLs

You can apply router ACLs on switch virtual interfaces (SVIs), which are Layer 3 interfaces to VLANs; on physical Layer 3 interfaces; and on Layer 3 EtherChannel interfaces. You apply router ACLs on interfaces for specific directions (inbound or outbound). You can apply one router ACL in each direction on an interface.

An ACL can be used with multiple features for a given interface, and one feature can use multiple ACLs. When a single router ACL is used by multiple features, it is examined multiple times.

Supported access lists for IPv4 traffic:

- Standard IP access lists use source addresses for matching operations.
- Extended IP access lists use source and destination addresses and optional protocol information for matching operations.

As with port ACLs, the switch examines ACLs associated with features configured on a given interface. However, you can apply only inbound port ACLs, while router ACLs are supported in both directions. As packets enter the switch on an interface, ACLs associated with all inbound features configured on that interface are examined. After packets are routed and before they are forwarded to the next hop, all ACLs associated with outbound features configured on the egress interface are examined.

ACLs permit or deny packet forwarding based on how the packet matches the entries in the ACL and can be used to control access to a network or to part of a network. In Figure 37-1, ACLs applied at the router input allow Host A to access the Human Resources network but prevent Host B from accessing the same network.

VLAN Maps

Use VLAN ACLs or VLAN maps to access-control all traffic. You can apply VLAN maps to all packets that are routed into or out of a VLAN or are bridged within a VLAN in the switch.

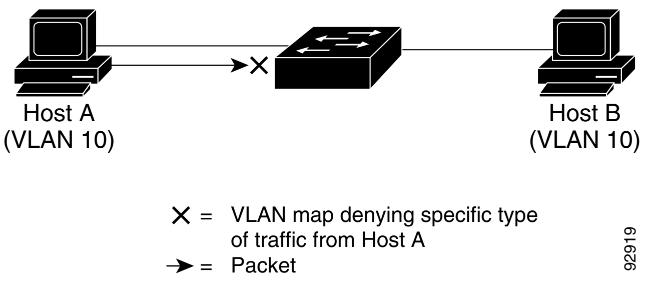
Use VLAN maps for security packet filtering. VLAN maps are not defined by direction (input or output).

You can configure VLAN maps to match Layer 3 addresses for IPv4 traffic.

All non-IP protocols are access-controlled through MAC addresses and Ethertype using MAC VLAN maps. (IP traffic is not access controlled by MAC VLAN maps.) You can enforce VLAN maps only on packets going through the switch; you cannot enforce VLAN maps on traffic between hosts on a hub or on another switch connected to this switch.

With VLAN maps, forwarding of packets is permitted or denied, based on the action specified in the map. Figure 37-2 shows how a VLAN map is applied to prevent a specific type of traffic from Host A in VLAN 10 from being forwarded. You can apply only one VLAN map to a VLAN.

Figure 75 Using VLAN Maps to Control Traffic



Handling Fragmented and Unfragmented Traffic

IP packets can be fragmented as they cross the network. When this happens, only the fragment containing the beginning of the packet contains the Layer 4 information, such as TCP or UDP port numbers, ICMP type and code, and so on. All other fragments are missing this information.

Some ACEs do not check Layer 4 information and therefore can be applied to all packet fragments. ACEs that do test Layer 4 information cannot be applied in the standard manner to most of the fragments in a fragmented IP packet. When the fragment contains no Layer 4 information and the ACE tests some Layer 4 information, the matching rules are modified:

- Permit ACEs that check the Layer 3 information in the fragment (including protocol type, such as TCP, UDP, and so on) are considered to match the fragment regardless of what the missing Layer 4 information might have been.
- Deny ACEs that check Layer 4 information never match a fragment unless the fragment contains Layer 4 information.

Consider access list 102, configured with these commands, applied to three fragmented packets:

```
Switch(config)# access-list 102 permit tcp any host 10.1.1.1 eq smtp
Switch(config)# access-list 102 deny tcp any host 10.1.1.2 eq telnet
Switch(config)# access-list 102 permit tcp any host 10.1.1.2
Switch(config)# access-list 102 deny tcp any any
```

Note: In the first and second ACEs in the examples, the *eq* keyword after the destination address means to test for the TCP-destination-port well-known numbers equaling Simple Mail Transfer Protocol (SMTP) and Telnet, respectively.

- Packet A is a TCP packet from host 10.2.2.2., port 65000, going to host 10.1.1.1 on the SMTP port. If this packet is fragmented, the first fragment matches the first ACE (a permit) as if it were a complete packet because all Layer 4 information is present. The remaining fragments also match the first ACE, even though they do not contain the SMTP port information, because the first ACE only checks Layer 3 information when applied to fragments. The information in this example is that the packet is TCP and that the destination is 10.1.1.1.
- Packet B is from host 10.2.2.2, port 65001, going to host 10.1.1.2 on the Telnet port. If this packet is fragmented, the first fragment matches the second ACE (a deny) because all Layer 3 and Layer 4 information is present. The remaining fragments in the packet do not match the second ACE because they are missing Layer 4 information. Instead, they match the third ACE (a permit).

  Because the first fragment was denied, host 10.1.1.2 cannot reassemble a complete packet, so packet B is effectively denied. However, the later fragments that are permitted will consume bandwidth on the network and resources of host 10.1.1.2 as it tries to reassemble the packet.
- Fragmented packet C is from host 10.2.2.2, port 65001, going to host 10.1.1.3, port ftp. If this packet is fragmented, the first fragment matches the fourth ACE (a deny). All other fragments also match the fourth ACE because that ACE does not check any Layer 4 information and because Layer 3 information in all fragments shows that they are being sent to host 10.1.1.3, and the earlier permit ACEs were checking different hosts.

Configuring IPv4 ACLs

Configuring IPv4 ACLs on the switch is the same as configuring IPv4 ACLs on other Cisco switches and routers.

1. Create an ACL by specifying an access list number or name and the access conditions.
2. Apply the ACL to interfaces or terminal lines.

Refer to the following sections for configuration information:

- Creating Standard and Extended IPv4 ACLs
- Applying an IPv4 ACL to a Terminal Line
- IPv4 ACL Application to an Interface Guidelines
- Hardware and Software Handling of IP ACLs
- Troubleshooting ACLs

Creating Standard and Extended IPv4 ACLs

This section describes IP ACLs. An ACL is a sequential collection of permit and deny conditions. One by one, the switch tests packets against the conditions in an access list. The first match determines whether the switch accepts or rejects the packet. Because the switch stops testing after the first match, the order of the conditions is critical. If no conditions match, the switch denies the packet.

The software supports these types of ACLs or access lists for IPv4:

- Standard IP access lists use source addresses for matching operations.
- Extended IP access lists use source and destination addresses for matching operations and optional protocol-type information for finer granularity of control.

The switch always rewrites the order of standard access lists so that entries with **host** matches and entries with matches having a *don't care* mask of 0.0.0.0 are moved to the top of the list, above any entries with non-zero *don't care* masks. Therefore, in **show** command output and in the configuration file, the ACEs do not necessarily appear in the order in which they were entered.

After creating a numbered standard IPv4 ACL, you can apply it to terminal lines (see Applying an IPv4 ACL to a Terminal Line, page 595), to interfaces (see Applying an IPv4 ACL to an Interface, page 596), or to VLANs (see Monitoring and Maintaining Network Security with ACLs, page 598).

Access List Numbers

The number you use to denote your ACL shows the type of access list that you are creating. Table 56 on page 580 lists the access-list number and corresponding access list type and shows whether or not they are supported in the switch. The switch supports IPv4 standard and extended access lists, numbers 1 to 199 and 1300 to 2699.

Table 56 Access List Number Support

Access List Number	Type	Supported
1-99	IP standard access list	Yes
100-199	IP extended access list	Yes
200-299	Protocol type-code access list	No
300-399	DECnet access list	No
400-499	XNS standard access list	No
500-599	XNS extended access list	No
600-699	AppleTalk access list	No
700-799	48-bit MAC address access list	No
800-899	IPX standard access list	No
900-999	IPX extended access list	No
1000-1099	IPX SAP access list	No
1100-1199	Extended 48-bit MAC address access list	No
1200-1299	IPX summary address access list	No
1300-1999	IP standard access list (expanded range)	Yes
2000-2699	IP extended access list (expanded range)	Yes

Note: In addition to numbered standard and extended ACLs, you can also create standard and extended named IP ACLs by using the supported numbers. That is, the name of a standard IP ACL can be 1 to 99; the name of an extended IP ACL can be 100 to 199. The advantage of using named ACLs instead of numbered lists is that you can delete individual entries from a named list.

ACL Logging

The switch software can provide logging messages about packets permitted or denied by a standard IP access list. That is, any packet that matches the ACL causes an informational logging message about the packet to be sent to the console. The level of messages logged to the console is controlled by the logging console commands controlling the syslog messages.

Note: Because routing is done in hardware and logging is done in software, if a large number of packets match a *permit* or *deny* ACE containing a **log** keyword, the software might not be able to match the hardware processing rate, and not all packets will be logged.

The first packet that triggers the ACL causes a logging message right away, and subsequent packets are collected over 5-minute intervals before they appear or logged. The logging message includes the access list number, whether the packet was permitted or denied, the source IP address of the packet, and the number of packets from that source permitted or denied in the prior 5-minute interval.

Creating a Numbered Standard ACL

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**access-list** access-list-number **{deny | permit}** source [source-wildcard] [log]	Define a standard IPv4 access list by using a source address and wildcard. The *access-list-number* is a decimal number from 1 to 99 or 1300 to 1999. Enter **deny** or **permit** to specify whether to deny or permit access if conditions are matched. The *source* is the source address of the network or host from which the packet is being sent specified as: The 32-bit quantity in dotted-decimal format. - The keyword any as an abbreviation for source and source-wildcard of 0.0.0.0 255.255.255.255. You do not need to enter a source-wildcard. - The keyword host as an abbreviation for source and source-wildcard of source 0.0.0.0. (Optional) The *source-wildcard* applies wildcard bits to the source. (Optional) Enter log to cause an informational logging message about the packet that matches the entry to be sent to the console. (Optional) Enter **log** to cause an informational logging message about the packet that matches the entry to be sent to the console.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show access-lists** [number | name]	Show the access list configuration.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Note: Use the **no access-list** *access-list-number* global configuration command to delete the entire ACL. You cannot delete individual ACEs from numbered access lists.

Use the no access-list access-list-number global configuration command to delete the entire ACL. You cannot delete individual ACEs from numbered access lists.

This example shows how to create a standard ACL to deny access to IP host 171.69.198.102, permit access to any others, and display the results.

```
Switch (config)# access-list 2 deny host 171.69.198.102
Switch (config)# access-list 2 permit any
Switch(config)# end
Switch# show access-lists
Standard IP access list 2
10 deny   171.69.198.102
20 permit any
```

The switch always rewrites the order of standard access lists so that entries with **host** matches and entries with matches having a don't care mask of 0.0.0.0 are moved to the top of the list, above any entries with non-zero don't care masks.

Therefore, in **show** command output and in the configuration file, the ACEs do not necessarily appear in the order in which they were entered.

Creating a Numbered Extended ACL

Although standard ACLs use only source addresses for matching, you can use extended ACL source and destination addresses for matching operations and optional protocol type information for finer granularity of control. When you are creating ACEs in numbered extended access lists, remember that after you create the ACL, any additions are placed at the end of the list. You cannot reorder the list or selectively add or remove ACEs from a numbered list.

Some protocols also have specific parameters and keywords that apply to that protocol.

These IP protocols are supported (protocol keywords are in parentheses in **bold**):

Authentication Header Protocol (**ahp**), Enhanced Interior Gateway Routing Protocol (**eigrp)**, Encapsulation Security Payload (**esp**), generic routing encapsulation (**gre**), Internet Control Message Protocol (**icmp)**, Internet Group Management Protocol (**igmp**), any Interior Protocol (**ip**), IP in IP tunneling (**ipinip**), KA9Q NOS-compatible IP over IP tunneling (**nos**), Open Shortest Path First routing (**ospf**), Payload Compression Protocol (**pcp**), Protocol Independent Multicast (**pim**), Transmission Control Protocol (**tcp**), User Datagram Protocol (**udp**),

Note: ICMP echo-reply cannot be filtered. All other ICMP codes or types can be filtered.

Note: The switch does not support dynamic or reflexive access lists. It also does not support filtering based on the type of service (ToS) minimize-monetary-cost bit.

Supported parameters can be grouped into these categories: TCP, UDP, ICMP, IGMP, or other IP.

After an ACL is created, any additions (possibly entered from the terminal) are placed at the end of the list. You cannot selectively add or remove access list entries from a numbered access list.

Note: When you are creating an ACL, remember that, by default, the end of the access list contains an implicit deny statement for all packets if it did not find a match before reaching the end.

After creating a numbered extended ACL, you can apply it to terminal lines (see Applying an IPv4 ACL to a Terminal Line, page 595), to interfaces (see Applying an IPv4 ACL to an Interface, page 596), or to VLANs (see Monitoring and Maintaining Network Security with ACLs, page 598)

.

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2a	**access-list** *access-list-number* **{deny | permit}** *protocol source source-wildcard destination* destination-wildcard **[precedence** *precedence*] **[tos** tos] **[fragments] [log] [log-input] [time-range** *time-range-name*] **[dscp** *dscp*] **Note:** If you enter a **dscp** value, you cannot enter **tos** or **precedence**. You can enter both a tos and a precedence value with no dscp.	Define an extended IPv4 access list and the access conditions. The access-list-number is a decimal number from 100 to 199 or 2000 to 2699. Enter **deny** or **permit** to specify whether to deny or permit the packet if conditions are matched.For protocol, enter the name or number of an IP protocol: **ahp, eigrp, esp, gre, icmp, igmp, igrp, ip, ipinip, nos, ospf, pcp, pim, tcp**, or **udp**, or an integer in the range 0 to 255 representing an IP protocol number. To match any Internet protocol (including ICMP, TCP, and UDP), use the keyword **ip**. **Note:** This step includes options for most IP protocols. For additional specific parameters for TCP, UDP, ICMP, and IGMP, see steps 2b through 2e. The *source* is the number of the network or host from which the packet is sent. The *source-wildcard* applies wildcard bits to the source. The *destination* is the network or host number to which the packet is sent. The *destination-wildcard* applies wildcard bits to the destination. Source, source-wildcard, destination, and destination-wildcard can be specified as: ■ The 32-bit quantity in dotted-decimal format. ■ The keyword **any** for 0.0.0.0 255.255.255.255 (any host). ■ The keyword **host** for a single host 0.0.0.0.

	Command	Purpose
Step 2a, continued		The other keywords are optional and have these meanings: ■ **precedence**–Enter to match packets with a precedence level specified as a number from 0 to 7 or by name: **routine (0)**, *priority* **(1)**, **immediate (2), flash** (3), **flash-override (4)**, **critical (5) internet (6)**, **network (7)**. ■ **fragments**–Enter to check non-initial fragments. ■ **tos**–Enter to match by type of service level, specified by a number from 0 to 15 or a name: normal (0), max-reliability (2), max-throughput (4), min-delay (8). ■ **log**–Enter to create an informational logging message to be sent to the console about the packet that matches the entry or log-input to include the input interface in the log entry. ■ If you enter a dscp value, you cannot enter tos or precedence. You can enter both a tos and a precedence value with no dscp ■ **time-range**–For an explanation of this keyword, see the "Using Time Ranges with ACLs" section on page 37-17. ■ **dscp**–Enter to match packets with the DSCP value specified by a number from 0 to 63, or use the question mark (?) to see a list of available values.
or	**access-list** *access-list-number* **{deny | permit}** *protocol* any any **[precedence** *precedence*] **[tos** *tos*] **[fragments] [log] [log-input] [time-range** *time-range-name*] **[dscp** *dscp*]	In access-list configuration mode, define an extended IP access list using an abbreviation for a source and source wildcard of 0.0.0.0 255.255.255.255 and an abbreviation for a destination and destination wildcard of 0.0.0.0 255.255.255.255. You can use the any keyword in place of source and destination address and wildcard.
or	**access-list** access-list-number **{deny | permit}** *protocol* **host** source **host** destination **[precedence** precedence] **[tos** ***tos*]** **[fragments] [log] [log-input] [time-range** *time-range-name*] **[dscp** *dscp*]	Define an extended IP access list by using an abbreviation for a source and a source wildcard of source 0.0.0.0 and an abbreviation for a destination and destination wildcard of destination 0.0.0.0. You can use the **host** keyword in place of the source and destination wildcard or mask.

	Command	Purpose
Step 2b	**access-list** *access-list-number* **{deny | permit} tcp** *source source-wildcard* [operator port] *destination destination-wildcard [operator port]* **[established] [precedence** *precedence*] **[tos** tos] **[fragments] [log] [log-input] [time-range time-range-name] [dscp** *dscp*] [*flag*]	(Optional) Define an extended TCP access list and the access conditions. Enter tcp for Transmission Control Protocol. The parameters are the same as those described in Step 2a, with these exceptions: (Optional) Enter an operator and port to compare source (if positioned after source source-wildcard) or destination (if positioned after destination destination-wildcard) port. Possible operators include eq (equal), gt (greater than), lt (less than), neq (not equal), and range (inclusive range). Operators require a port number (range requires two port numbers separated by a space). Enter the port number as a decimal number (from 0 to 65535) or the name of a TCP port. To see TCP port names, use the ? or see the " Configuring IP Services" section in the " IP Addressing and Services" chapter of the Cisco IOS IP Configuration Guide, Release 12.2. Use only TCP port numbers or names when filtering TCP. The other optional keywords have these meanings: ■ established–Enter to match an established connection. This has the same function as matching on the ack or rst flag. ■ flag–Enter one of these flags to match by the specified TCP header bits: ack (acknowledge), fin (finish), psh (push), rst (reset), syn (synchronize), or urg (urgent)
Step 2c	**access-list** *access-list-number* **{deny | permit} udp** *source source-wildcard [operator port] destination destination-wildcard [operator port]* **[precedence** *precedence*] **[tos** *tos*] **[fragments] [log] [log-input] [time-range** *time-range-name*] **[dscp** *dscp***]**	(Optional) Define an extended UDP access list and the access conditions. Enter udp for the User Datagram Protocol. The UDP parameters are the same as those described for TCP except that the *[operator [port]]* port number or name must be a UDP port number or name, and the **flag** and **established** parameters are not valid for UDP.

	Command	Purpose
Step 2d	**access-list** *access-list-number* **{deny | permit} icmp** *source source-wildcard destination destination-wildcard [icmp-type* | [[*icmp-type icmp-code*] | [*icmp-message*]] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log**] [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*]	(Optional) Define an extended ICMP access list and the access conditions. Enter **icmp** for Internet Control Message Protocol. The ICMP parameters are the same as those described for most IP protocols in Step 2a, with the addition of the ICMP message type and code parameters. These optional keywords have these meanings: ■ icmp-type—Enter to filter by ICMP message type, a number from 0 to 255. ■ icmp-code—Enter to filter ICMP packets that are filtered by the ICMP message code type, a number from 0 to 255. ■ icmp-message—Enter to filter ICMP packets by the ICMP message type name or the ICMP message type and code name. To see a list of ICMP message
Step 2e	**access-list** *access-list-number* **{deny | permit} igmp** *source source-wildcard destination destination-wildcard [igmp-type]* **[precedence** *precedence*] **[tos** *tos*] **[fragments] [log] [log-input] [time-range** *time-range-name*] **[dscp** *dscp*]	(Optional) Define an extended IGMP access list and the access conditions. Enter **igmp** for Internet Group Management Protocol. The IGMP parameters are the same as those described for most IP protocols in Step 2a, with this optional parameter. *igmp-type*—To match IGMP message type, enter a number from 0 to 15, or enter the message name (**dvmrp, host-query, host-report, pim,** or **trace**)
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show access-lists [number | name]**	Verify the access list configuration.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no access-list** *access-list-number* global configuration command to delete the entire access list. You cannot delete individual ACEs from numbered access lists.

This example shows how to create and display an extended access list to deny Telnet access from any host in network 171.69.198.0 to any host in network 172.20.52.0 and to permit any others. (The eq keyword after the destination address means to test for the TCP destination port number equaling Telnet.)

```
Switch(config)# access-list 102 deny tcp 171.69.198.0 0.0.0.255 172.20.52.0 0.0.0.255 eq telnet
Switch(config)# access-list 102 permit tcp any any
Switch(config)# end
Switch# show access-lists
Extended IP access list 102
    10 deny tcp 171.69.198.0 0.0.0.255 172.20.52.0 0.0.0.255 eq telnet
    20 permit tcp any any
```

After an ACL is created, any additions (possibly entered from the terminal) are placed at the end of the list. You cannot selectively add or remove access list entries from a numbered access list.

Note: When you are creating an ACL, remember that, by default, the end of the access list contains an implicit deny statement for all packets if it did not find a match before reaching the end.

After creating a numbered extended ACL, you can apply it to terminal lines (see the Applying an IPv4 ACL to a Terminal Line section), to interfaces (see the Applying an IPv4 ACL to an Interface section), or to VLANs (see the Configuring VLAN Maps section

Resequencing ACEs in an ACL

Sequence numbers for the entries in an access list are automatically generated when you create a new ACL. You can use the **ip access-list resequence** global configuration command to edit the sequence numbers in an ACL and change the order in which ACEs are applied. For example, if you add a new ACE to an ACL, it is placed at the bottom of the list. By changing the sequence number, you can move the ACE to a different position in the ACL.

Named Standard and Extended ACLs

You can identify IPv4 ACLs with an alphanumeric string (a name) rather than a number. You can use named ACLs to configure more IPv4 access lists in a router than if you were to use numbered access lists. If you identify your access list with a name rather than a number, the mode and command syntax are slightly different. However, not all commands that use IP access lists accept a named access list.

Note: The name you give to a standard or extended ACL can also be a number in the supported range of access list numbers. That is, the name of a standard IP ACL can be 1 to 99; the name of an extended IP ACL can be 100 to 199. The advantage of using named ACLs instead of numbered lists is that you can delete individual entries from a named list.

Consider these guidelines and limitations before configuring named ACLs:

- Not all commands that accept a numbered ACL accept a named ACL. ACLs for packet filters and route filters on interfaces can use a name.
- A standard ACL and an extended ACL cannot have the same name.
- Numbered ACLs are also available, as described in the Creating a Numbered Standard ACL, page 591.

When you are creating standard extended ACLs, remember that, by default, the end of the ACL contains an implicit deny statement for everything if it did not find a match before reaching the end. For standard ACLs, if you omit the mask from an associated IP host address access list specification, 0.0.0.0 is assumed to be the mask.

After you create an ACL, any additions are placed at the end of the list. You cannot selectively add ACL entries to a specific ACL. However, you can use **no permit** and **no deny** access-list configuration mode commands to remove entries from a named ACL. This example shows how you can delete individual ACEs from the named access list *border-list*:

```
Switch(config)# ip access-list extended border-list
Switch(config-ext-nacl)# no permit ip host 10.1.1.3 any
```

Being able to selectively remove lines from a named ACL is one reason you might use named ACLs instead of numbered ACLs.

Time Ranges with ACLs

You can selectively apply extended ACLs based on the time of day and the week by using the **time-range** global configuration command. First, define a time-range name and set the times and the dates or the days of the week in the time range. Then enter the time-range name when applying an ACL to set restrictions to the access list. You can use the time range to define when the permit or deny statements in the ACL are in effect, for example, during a specified time period or on specified days of the week.

These are some of the many possible benefits of using time ranges:

- You have more control over permitting or denying a user access to resources, such as an application (identified by an IP address/mask pair and a port number).

- You can control logging messages. ACL entries can be set to log traffic only at certain times of the day. Therefore, you can simply deny access without needing to analyze many logs generated during peak hours.

Time-based access lists trigger CPU activity because the new configuration of the access list must be merged with other features and the combined configuration loaded into the TCAM. For this reason, you should be careful not to have several access lists configured to take affect in close succession (within a small number of minutes of each other.)

Note: The time range relies on the switch system clock; therefore, you need a reliable clock source. We recommend that you use Network Time Protocol (NTP) to synchronize the switch clock.

Comments in ACLs

You can use the **remark** keyword to include comments (remarks) about entries in any IP standard or extended ACL. The remarks make the ACL easier for you to understand and scan. Each remark line is limited to 100 characters.

The remark can go before or after a permit or deny statement. You should be consistent about where you put the remark so that it is clear which remark describes which permit or deny statement. For example, it would be confusing to have some remarks before the associated permit or deny statements and some remarks after the associated statements.

To include a comment for IP numbered standard or extended ACLs, use the **access-list** *access-list number* **remark** *remark* global configuration command. To remove the remark, use the **no** form of this command.

Applying an IPv4 ACL to a Terminal Line

You can use numbered ACLs to control access to one or more terminal lines. You cannot apply named ACLs to lines. You must set identical restrictions on all the virtual terminal lines because a user can attempt to connect to any of them.

For procedures for applying ACLs to interfaces, see Applying an IPv4 ACL to an Interface, page 596. For applying ACLs to VLANs, see Monitoring and Maintaining Network Security with ACLs, page 598.

IPv4 ACL Application to an Interface Guidelines

- Apply an ACL only to inbound Layer 2 ports.
- Apply an ACL to either outbound or inbound Layer 3 interfaces.
- When controlling access to an interface, you can use a named or numbered ACL.
- If you apply an ACL to a port that is a member of a VLAN, the port ACL takes precedence over an ACL applied to the VLAN interface.
- If you apply an ACL to a Layer 2 interface that is a member of a VLAN, the Layer 2 (port) ACL takes precedence over an input Layer 3 ACL applied to the VLAN interface. The port ACL always filters incoming packets received on the Layer 2 port.
- If you apply an ACL to a Layer 3 interface and routing is not enabled, the ACL only filters packets that are intended for the CPU, such as SNMP, Telnet, or web traffic. You do not have to enable routing to apply ACLs to Layer 2 interfaces.
- When private VLANs are configured, you can apply router ACLs only on the primary-VLAN SVIs. The ACL is applied to both primary and secondary VLAN Layer 3 traffic.

Note: By default, the router sends Internet Control Message Protocol (ICMP) unreachable messages when a packet is denied by an access group. These access-group denied packets are not dropped in hardware but are bridged to the switch CPU so that it can generate the ICMP-unreachable message. Port ACLs are an exception. They do not generate

ICMP unreachable messages.

ICMP unreachable messages can be disabled on router ACLs with the **no ip unreachables** interface command.

For inbound ACLs, after receiving a packet, the switch checks the packet against the ACL. If the ACL permits the packet, the switch continues to process the packet. If the ACL rejects the packet, the switch discards the packet.

For outbound ACLs, after receiving and sending a packet to a controlled interface, the switch checks the packet against the ACL. If the ACL permits the packet, the switch sends the packet. If the ACL rejects the packet, the switch discards the packet.

By default, the input interface sends ICMP Unreachable messages whenever a packet is discarded, regardless of whether the packet was discarded because of an ACL on the input interface or because of an ACL on the output interface. ICMP Unreachables are normally limited to no more than one every one-half second per input interface, but this can be changed by using the **ip icmp rate-limit unreachable** global configuration command.

When you apply an undefined ACL to an interface, the switch acts as if the ACL has not been applied to the interface and permits all packets. Remember this behavior if you use undefined ACLs for network security.

Hardware and Software Handling of IP ACLs

ACL processing is primarily accomplished in hardware, but requires forwarding of some traffic flows to the CPU for software processing. If the hardware reaches its capacity to store ACL configurations, packets are sent to the CPU for forwarding. The forwarding rate for software-forwarded traffic is substantially less than for hardware-forwarded traffic.

Note: If an ACL configuration cannot be implemented in hardware due to an out-of-resource condition on a switch, then only the traffic in that VLAN arriving on that switch is affected (forwarded in software). Software forwarding of packets might adversely impact the performance of the switch, depending on the number of CPU cycles that this consumes.

For router ACLs, other factors can cause packets to be sent to the CPU:

- Using the **log** keyword
- Generating ICMP unreachable messages

When traffic flows are both logged and forwarded, forwarding is done by hardware, but logging must be done by software. Because of the difference in packet handling capacity between hardware and software, if the sum of all flows being logged (both permitted flows and denied flows) is of significant bandwidth, not all of the packets that are forwarded can be logged.

If router ACL configuration cannot be applied in hardware, packets arriving in a VLAN that must be routed are routed in software, but are bridged in hardware. If ACLs cause large numbers of packets to be sent to the CPU, the switch performance can be negatively affected.

When you enter the **show ip access-lists** privileged EXEC command, the match count displayed does not account for packets that are access controlled in hardware. Use the **show access-lists hardware counters** privileged EXEC command to obtain some basic hardware ACL statistics for switched and routed packets.

Troubleshooting ACLs

If this ACL manager message appears, where [chars] is the access-list name, the switch then has insufficient resources to create a hardware representation of the ACL.

```
ACLMGR-2-NOVMR: Cannot generate hardware representation of access list [chars]
```

The resources include hardware memory and label space but not CPU memory. A lack of available logical operation units or specialized hardware resources causes this problem. Logical operation units are needed for a TCP flag match or a test other than **eq** (**ne**, **gt**, **lt**, or **range**) on TCP, UDP, or SCTP port numbers.

Use one of these workarounds:

- Modify the ACL configuration to use fewer resources.
- Rename the ACL with a name or number that alphanumerically precedes the ACL names or numbers.

To determine the specialized hardware resources, enter the **show platform layer4 acl map** privileged EXEC command. If the switch does not have available resources, the output shows that index 0 to index 15 are not available.

For more information about configuring ACLs with insufficient resources, see CSCsq63926 in the Bug Toolkit.

Named MAC Extended ACLs

You can filter non-IPv4 traffic on a VLAN or on a Layer 2 interface by using MAC addresses and named MAC extended ACLs. The procedure is similar to that of configuring other extended named ACLs.

Note: You cannot apply named MAC extended ACLs to Layer 3 interfaces.

Note: Though visible in the command-line help strings, **appletalk** is not supported as a matching condition for the **deny** and **permit** MAC access-list configuration mode commands.

MAC ACL to a Layer 2 Interface

After you create a MAC ACL, you can apply it to a Layer 2 interface to filter non-IP traffic coming in that interface. When you apply the MAC ACL, consider these guidelines:

- If you apply an ACL to a Layer 2 interface that is a member of a VLAN, the Layer 2 (port) ACL takes precedence over an input Layer 3 ACL applied to the VLAN interface. Incoming packets received on the Layer 2 port are always filtered by the port ACL.
- You can apply no more than one IP access list and one MAC access list to the same Layer 2 interface. The IP access list filters only IP packets, and the MAC access list filters non-IP packets.
- A Layer 2 interface can have only one MAC access list. If you apply a MAC access list to a Layer 2 interface that has a MAC ACL configured, the new ACL replaces the previously configured one.

How to Configure Network Security with ACLs

Creating a Numbered Standard ACL

Note: When creating an ACL, remember that, by default, the end of the ACL contains an implicit deny statement for all packets that it did not find a match for before reaching the end. With standard access lists, if you omit the mask from an associated IP host address ACL specification, 0.0.0.0 is assumed to be the mask.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] [**log**] **Note:** Use the no access-list access-list-number global configuration command to delete the entire ACL.	Defines a standard IPv4 access list by using a source address and wildcard. *access-list-number*–Specifies a decimal number from 1 to 99 or 1300 to 1999. **deny** or **permit**–Specifies whether to deny or permit access if conditions are matched. *source*–Specifies the source address of the network or host from which the packet is being sent specified as: ■ The 32-bit quantity in dotted-decimal format. ■ The keyword **any** as an abbreviation for *source* and *source-wildcard* of 0.0.0.0 255.255.255.255. You do not need to enter a source-wildcard. ■ The keyword **host** as an abbreviation for source and source-wildcard of *source* 0.0.0.0. (Optional) *source-wildcard*–Applies wildcard bits to the source. (Optional) **log**–Causes an informational logging message about the packet that matches the entry to be sent to the console.
3.	**end**	Returns to privileged EXEC mode.
4.	**show access-lists** [number | name]	Show the access list configuration.
5.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Creating a Numbered Extended ACL

Although standard ACLs use only source addresses for matching, you can use extended ACL source and destination addresses for matching operations and optional protocol type information for finer granularity of control. When you are creating ACEs in numbered extended access lists, remember that after you create the ACL, any additions are placed at the end of the list. You cannot reorder the list or selectively add or remove ACEs from a numbered list.

Some protocols also have specific parameters and keywords that apply to that protocol.

These IP protocols are supported (protocol keywords are in parentheses in bold):

Authentication Header Protocol (ahp), Enhanced Interior Gateway Routing Protocol (eigrp), Encapsulation Security Payload (esp), generic routing encapsulation (gre), Internet Control Message Protocol (icmp), Internet Group Management Protocol (igmp), any Interior Protocol (ip), IP in IP tunneling (ipinip), KA9Q NOS-compatible IP over IP tunneling (nos), Open Shortest Path First routing (ospf), Payload Compression Protocol (pcp), Protocol Independent.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2. a	**access-list** *access-list-number* { **deny** | **permit** } *protocol source source-wildcard destination destination-wildcard* [**precedence** *precedence*] [**tos** tos] [**fragments**] [**log**] [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*] **Note:** If you enter a dscp value, you cannot enter tos or precedence. You can enter both a tos and a precedence value with no dscp	Defines an extended IPv4 access list and the access conditions. *access-list-number* – Specifies a decimal number from 100 to 199 or 2000 to 2699. *deny or permit* –Specifies whether to deny or permit the packet if conditions are matched. *protocol* –Specifies the name or number of an IP protocol: ahp, eigrp, esp, gre, icmp, igmp, igrp, ip, ipinip, nos, ospf, pcp, pim, tcp, or udp, or an integer in the range 0 to 255 representing an IP protocol number. To match any Internet protocol (including ICMP, TCP, and UDP), use the keyword ip. **Note:** This step includes options for most IP protocols. For additional specific parameters for TCP, UDP, ICMP, and IGMP, see steps 2b through 2e. source –The number of the network or host from which the packet is sent. source-wildcard– Applies wildcard bits to the source. destination– The network or host number to which the packet is sent. destination-wildcard– Applies wildcard bits to the destination. source, source-wildcard, destination, and destination-wildcard can be specified as: ■The 32-bit quantity in dotted-decimal format. ■The keyword any for 0.0.0.0 255.255.255.255 (any host). ■The keyword host for a single host 0.0.0.0. The other keywords are optional and have these meanings: ■ precedence –Matches packets with a precedence level specified as a number from 0 to 7 or by name: routine (0), priority (1), immediate (2), flash (3), flash-override (4), critical (5), internet (6), network (7). ■ fragments –Checks non-initial fragments. ■ tos –Matches by type of service level, specified by a number from 0 to 15 or a name: normal (0), max-reliability (2), max-throughput (4), min-delay (8). ■ log –Creates an informational logging message to be sent to the console about the packet that matches the entry or log-input to include the input interface in the log entry. ■ time-range –For an explanation of this keyword, see Using Time Ranges with ACLs. ■ dscp –Matches packets with the DSCP value specified by a number from 0 to 63, or use the question mark (?) to see a list of available values.

	Command	Purpose
or	**access-list** *access-list-number* **{ deny | permit }** *protocol* **any any [precedence** *precedence* **] [tos** *tos* **] [fragments] [log]** [*log-input*] **[time-range** *time-range-name*] **[dscp** *dscp*]	In access-list configuration mode, defines an extended IP access list using an abbreviation for a source and source wildcard of 0.0.0.0 255.255.255.255 and an abbreviation for a destination and destination wildcard of 0.0.0.0 255.255.255.255. You can use the any keyword in place of source and destination address and wildcard.

Creating Named Standard and Extended ACLs

Note: When creating an ACL, remember that, by default, the end of the ACL contains an implicit deny statement for all packets that it did not find a match for before reaching the end. With standard access lists, if you omit the mask from an associated IP host address ACL specification, 0.0.0.0 is assumed to be the mask.

<insert tables/steps>

Using Time Ranges with ACLs

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**time-range** *time-range-name*	Assigns a meaningful name (for example, *workhours*) to the time range to be created, and enters time-range configuration mode. The name cannot contain a space or quotation mark and must begin with a letter.
3.	**absolute** [**start** *time date*] [**end** *time date*] or **periodic** *day-of-the-week hh:mm to* [*day-of-the-week*] *hh:mm* or **periodic** {**weekdays** | **weekend** | **daily**} *hh:mm to hh:mm*	Specifies when the function it will be applied to is operational. ■ You can use only one **absolute** statement in the time range. If you configure more than one absolute statement, only the one configured last is executed. ■ You can enter multiple **periodic** statements. For example, you could configure different hours for weekdays and weekends. See the example configurations.
4.	**end**	Returns to privileged EXEC mode.

Applying an IPv4 ACL to a Terminal Line

This task restricts incoming and outgoing connections between a virtual terminal line and the addresses in an ACL:

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**line** [**console** | **vty**] *line-number*	Identifies a specific line to configure, and enters in-line configuration mode. - **console**–Specifies the console terminal line. The console port is DCE. - **vty**–Specifies a virtual terminal for remote console access. The *line-number* is the first line number in a contiguous group that you want to configure when the line type is specified. The range is from 0 to 16.
3.	**access-class** *access-list-number* {**in** | **out**}	Restricts incoming and outgoing connections between a particular virtual terminal line (into a device) and the addresses in an access list.
4.	**end**	Returns to privileged EXEC mode.

Applying an IPv4 ACL to an Interface

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Identifies a specific interface for configuration, and enters interface configuration mode. The interface is a Layer 2 interface (port ACL).
3.	**ip access-group** {*access-list-number | name*} {**in** | **out**}	Controls access to the specified interface. The **out** keyword is not supported for Layer 2 interfaces (port ACLs).
4.	**end**	Returns to privileged EXEC mode.

Creating Named MAC Extended ACLs

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**mac access-list extended** *name*	Defines an extended MAC access list using a name.
3.	{**deny** | **permit**} {**any** | **host** *source MAC address* | *source MAC address mask*} {**any** | **host** *destination MAC address* | *destination MAC address mask*} [*type mask* | **lsap** *lsap mask* | **aarp** | **amber** | **dec-spanning** | **decnet-iv** | **diagnostic** | **dsm** | **etype-6000** | **etype-8042** | **lat** | **lavc-sca** | **mop-console** | **mop-dump** | **msdos** | **mumps** | **netbios** | **vines-echo** |**vines-ip** | **xns-idp** **|** *0-65535*] [**cos** *cos*]	In extended MAC access-list configuration mode, specifies to **permit** or **deny any** source MAC address, a source MAC address with a mask, or a specific **host** source MAC address and **any** destination MAC address, destination MAC address with a mask, or a specific destination MAC address. (Optional) You can also enter these options: ■ *type mask*–Specifies an arbitrary EtherType number of a packet with Ethernet II or SNAP encapsulation in decimal, hexadecimal, or octal with optional mask of *don't care* bits applied to the EtherType before testing for a match. ■ **lsap** *lsap mask*–Specifies an LSAP number of a packet with IEEE 802.2 encapsulation in decimal, hexadecimal, or octal with optional mask of *don't care* bits. ■ **aarp** | **amber** | **dec-spanning** | **decnet-iv** | **diagnostic** | **dsm** | **etype-6000** | **etype-8042** | **lat** | **lavc-sca** | **mop-console** | **mop-dump** | **msdos** | **mumps** | **netbios** | **vines-echo** |**vines-ip** | **xns-idp**–Specifies a non-IP protocol. ■ **cos** *cos*–Specifies an IEEE 802.1Q cost of service number from 0 to 7 used to set priority.
4.	**end**	Returns to privileged EXEC mode.

Applying a MAC ACL to a Layer 2 Interface

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Identifies a specific interface, and enters interface configuration mode. The interface must be a physical Layer 2 interface (port ACL).
3.	**mac access-group** {*name*} {**in**}	Controls access to the specified interface by using the MAC access list. Port ACLs are supported only in the inbound direction.
4.	**end**	Returns to privileged EXEC mode.

Monitoring and Maintaining Network Security with ACLs

Command	Purpose
show access-lists [*number* | *name*]	Displays the contents of one or all current IP and MAC address access lists or a specific access list (numbered or named).
show ip access-lists [*number* | *name*]	Displays the contents of all current IP access lists or a specific IP access list (numbered or named).
show ip interface *interface-id*	Displays detailed configuration and status of an interface. If IP is enabled on the interface and ACLs have been applied by using the **ip access-group** interface configuration command, the access groups are included in the display.
show running-config [**interface** *interface-id*]	Displays the contents of the configuration file for the switch or the specified interface, including all configured MAC and IP access lists and which access groups are applied to an interface.
show mac access-group [**interface** *interface-i*d]	Displays MAC access lists applied to all Layer 2 interfaces or the specified Layer 2 interface.
show access-lists [*number* | *name*]	Displays the access list configuration.
show time-range	Verifies the time-range configuration.
show mac access-group [**interface** *interface-i*d]	Displays the MAC access list applied to the interface or all Layer 2 interfaces.

Configuration Examples for Network Security with ACLs

Creating a Standard ACL: Example

This example shows how to create a standard ACL to deny access to IP host 171.69.198.102, permit access to any others, and display the results.

```
Switch (config)# access-list 2 deny host 171.69.198.102
Switch (config)# access-list 2 permit any
Switch(config)# end
Switch# show access-lists
Standard IP access list 2
    10 deny   171.69.198.102
    20 permit any
```

Creating an Extended ACL: Example

This example shows how to create and display an extended access list to deny Telnet access from any host in network 171.69.198.0 to any host in network 172.20.52.0 and to permit any others. (The **eq** keyword after the destination address means to test for the TCP destination port number equaling Telnet.)

```
Switch(config)# access-list 102 deny tcp 171.69.198.0 0.0.0.255 172.20.52.0 0.0.0.255 eq telnet
Switch(config)# access-list 102 permit tcp any any
Switch(config)# end
Switch# show access-lists
Extended IP access list 102
    10 deny tcp 171.69.198.0 0.0.0.255 172.20.52.0 0.0.0.255 eq telnet
    20 permit tcp any any
```

Configuring Time Ranges: Examples

This example shows how to configure time ranges for *workhours* and to configure January 1, 2006, as a company holiday and to verify your configuration.

```
Switch(config)# time-range workhours
Switch(config-time-range)# periodic weekdays 8:00 to 12:00
Switch(config-time-range)# periodic weekdays 13:00 to 17:00
Switch(config-time-range)# exit
Switch(config)# time-range new_year_day_2006
Switch(config-time-range)# absolute start 00:00 1 Jan 2006 end 23:59 1 Jan 2006
Switch(config-time-range)# end
Switch# show time-range
time-range entry: new_year_day_2003 (inactive)
   absolute start 00:00 01 January 2006 end 23:59 01 January 2006
time-range entry: workhours (inactive)
   periodic weekdays 8:00 to 12:00
   periodic weekdays 13:00 to 17:00
```

To apply a time range, enter the time-range name in an extended ACL that can implement time ranges. This example shows how to create and verify extended access list 188 that denies TCP traffic from any source to any destination during the defined holiday times and permits all TCP traffic during work hours.

```
Switch(config)# access-list 188 deny tcp any any time-range new_year_day_2006
Switch(config)# access-list 188 permit tcp any any time-range workhours
Switch(config)# end
Switch# show access-lists
Extended IP access list 188
   10 deny tcp any any time-range new_year_day_2006 (inactive)
   20 permit tcp any any time-range workhours (inactive)
```

Using Named ACLs: Example

This example uses named ACLs to permit and deny the same traffic.

```
Switch(config)# ip access-list extended deny_access
Switch(config-ext-nacl)# deny tcp any any time-range new_year_day_2006
Switch(config-ext-nacl)# exit
Switch(config)# ip access-list extended may_access
Switch(config-ext-nacl)# permit tcp any any time-range workhours
Switch(config-ext-nacl)# end
Switch# show ip access-lists
Extended IP access list lpip_default
    10 permit ip any any
Extended IP access list deny_access
    10 deny tcp any any time-range new_year_day_2006 (inactive)
Extended IP access list may_access
    10 permit tcp any any time-range workhours (inactive)
```

Including Comments in ACLs: Examples

In this example, the workstation that belongs to Jones is allowed access, and the workstation that belongs to Smith is not allowed access:

```
Switch(config)# access-list 1 remark Permit only Jones workstation through
Switch(config)# access-list 1 permit 171.69.2.88
Switch(config)# access-list 1 remark Do not allow Smith through
Switch(config)# access-list 1 deny 171.69.3.13
```

For an entry in a named IP ACL, use the **remark** access-list configuration command. To remove the remark, use the **no** form of this command.

In this example, the Jones subnet is not allowed to use outbound Telnet:

```
Switch(config)# ip access-list extended telnetting
Switch(config-ext-nacl)# remark Do not allow Jones subnet to telnet out
Switch(config-ext-nacl)# deny tcp host 171.69.2.88 any eq telnet
```

Applying ACL to a Port: Example

This example shows how to apply access list 2 to a port to filter packets entering the port:

```
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# ip access-group 2 in
```

Applying an ACL to an Interface: Example

For example, if you apply this ACL to an interface:

```
permit tcp source source-wildcard destination destination-wildcard range 5 60
permit tcp source source-wildcard destination destination-wildcard range 15 160
permit tcp source source-wildcard destination destination-wildcard range 115 1660
permit tcp source source-wildcard destination destination-wildcard
```

And if this message appears:

```
ACLMGR-2-NOVMR: Cannot generate hardware representation of access list [chars]
```

The flag-related operators are not available. To avoid this issue,

- Move the fourth ACE before the first ACE by using **ip access-list resequence** global configuration command:

  ```
  permit tcp source source-wildcard destination destination-wildcard
  permit tcp source source-wildcard destination destination-wildcard range 5 60
  permit tcp source source-wildcard destination destination-wildcard range 15 160
  permit tcp source source-wildcard destination destination-wildcard range 115 1660
  ```

  or

- Rename the ACL with a name or number that alphanumerically precedes the other ACLs (for example, rename ACL *79* to ACL *1*).

You can now apply the first ACE in the ACL to the interface. The switch allocates the ACE to available mapping bits in the Opselect index and then allocates flag-related operators to use the same bits in the TCAM.

Router ACLs function as follows:

- The hardware controls permit and deny actions of standard and extended ACLs (input and output) for security access control.
- If **log** has not been specified, 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 hardware.
- Adding the **log** keyword to an ACE in a router ACL causes a copy of the packet to be sent to the CPU for logging only. If the ACE is a *permit* statement, the packet is still switched and routed in hardware.

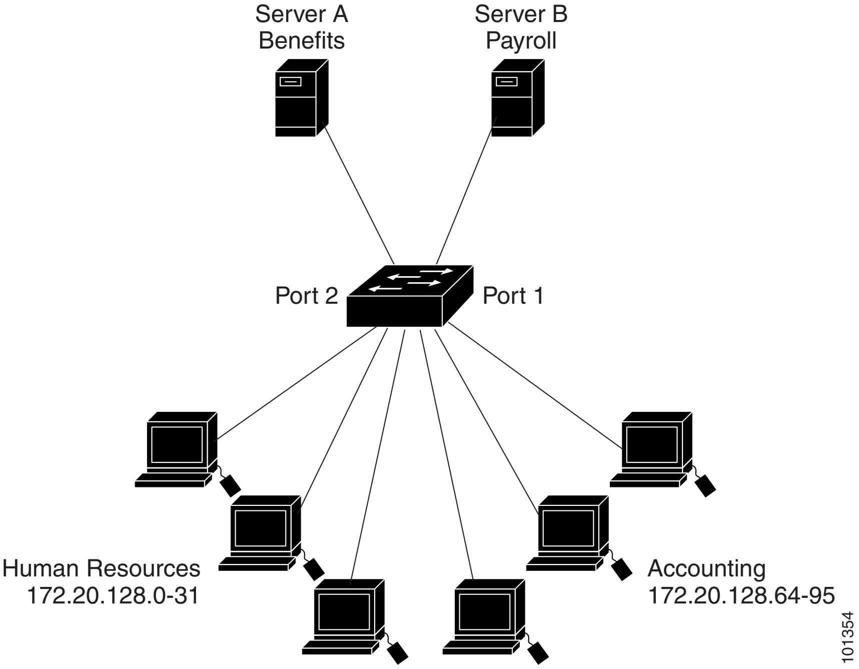
Routed ACLs: Examples

Figure 76 on page 601 shows a small networked office environment with routed Port 2 connected to Server A, containing benefits and other information that all employees can access, and routed Port 1 connected to Server B, containing confidential payroll data. All users can access Server A, but Server B has restricted access.

Use router ACLs to do this in one of two ways:

- Create a standard ACL, and filter traffic coming to the server from Port 1.
- Create an extended ACL, and filter traffic coming from the server into Port 1.

Figure 76 Using Router ACLs to Control Traffic



This example uses a standard ACL to filter traffic coming into Server B from a port, permitting traffic only from Accounting’s source addresses 172.20.128.64 to 172.20.128.95. The ACL is applied to traffic coming out of routed Port 1 from the specified source address.

```
Switch(config)# access-list 6 permit 172.20.128.64 0.0.0.31
Switch(config)# end
Switch# show access-lists
Standard IP access list 6
 permit 172.20.128.64, wildcard bits 0.0.0.31
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# ip access-group 6 out
```

This example uses an extended ACL to filter traffic coming from Server B into a port, permitting traffic from any source address (in this case Server B) to only the Accounting destination addresses 172.20.128.64 to 172.20.128.95. The ACL is applied to traffic going into routed Port 1, permitting it to go only to the specified destination addresses. Note that with extended ACLs, you must enter the protocol (IP) before the source and destination information.

```
Switch(config)# access-list 106 permit ip any 172.20.128.64 0.0.0.31
Switch(config)# end
Switch# show access-lists
Extended IP access list 106
```

```
 permit ip any 172.20.128.64 0.0.0.31
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# ip access-group 106 in
```

Configuring Numbered ACLs: Example

In this example, network 36.0.0.0 is a Class A network whose second octet specifies a subnet; that is, its subnet mask is 255.255.0.0. The third and fourth octets of a network 36.0.0.0 address specify a particular host. Using access list 2, the switch accepts one address on subnet 48 and reject all others on that subnet. The last line of the list shows that the switch accepts addresses on all other network 36.0.0.0 subnets. The ACL is applied to packets entering a port.

```
Switch(config)# access-list 2 permit 36.48.0.3
Switch(config)# access-list 2 deny 36.48.0.0 0.0.255.255
Switch(config)# access-list 2 permit 36.0.0.0 0.255.255.255
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# ip access-group 2 in
```

Configuring Extended ACLs: Examples

In this example, the first line permits any incoming TCP connections with destination ports greater than 1023. The second line permits incoming TCP connections to the Simple Mail Transfer Protocol (SMTP) port of host 128.88.1.2. The third line permits incoming ICMP messages for error feedback.

```
Switch(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 gt 1023
Switch(config)# access-list 102 permit tcp any host 128.88.1.2 eq 25
Switch(config)# access-list 102 permit icmp any any
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# ip access-group 102 in
```

In this example, suppose that you have a network connected to the Internet, and you want any host on the network to be able to form TCP connections to any host on the Internet. However, you do not want IP hosts to be able to form TCP connections to hosts on your network, except to the mail (SMTP) port of a dedicated mail host.

SMTP uses TCP port 25 on one end of the connection and a random port number on the other end. The same port numbers are used throughout the life of the connection. Mail packets coming in from the Internet have a destination port of 25. Outbound packets have the port numbers reversed. Because the secure system of the network always accepts mail connections on port 25, the incoming and outgoing services are separately controlled. The ACL must be configured as an input ACL on the outbound interface and an output ACL on the inbound interface.

In this example, the network is a Class B network with the address 128.88.0.0, and the mail host address is 128.88.1.2. The **established** keyword is used only for the TCP to show an established connection. A match occurs if the TCP datagram has the ACK or RST bits set, which show that the packet belongs to an existing connection. Gigabit Ethernet interface 1 is the interface that connects the router to the Internet.

```
Switch(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 established
Switch(config)# access-list 102 permit tcp any host 128.88.1.2 eq 25
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# ip access-group 102 in
```

Creating Named ACLs: Example

This example creates a standard ACL named *Internet_filter* and an extended ACL named *marketing_group*. The *Internet_filter* ACL allows all traffic from the source address 1.2.3.4.

```
Switch(config)# ip access-list standard Internet_filter
Switch(config-ext-nacl)# permit 1.2.3.4
Switch(config-ext-nacl)# exit
```

The *marketing_group* ACL allows any TCP Telnet traffic to the destination address and wildcard 171.69.0.0 0.0.255.255 and denies any other TCP traffic. It permits ICMP traffic, denies UDP traffic from any source to the destination address range 171.69.0.0 through 179.69.255.255 with a destination port less than 1024, denies any other IP traffic, and provides a log of the result.

```
Switch(config)# ip access-list extended marketing_group
Switch(config-ext-nacl)# permit tcp any 171.69.0.0 0.0.255.255 eq telnet
Switch(config-ext-nacl)# deny tcp any any
Switch(config-ext-nacl)# permit icmp any any
Switch(config-ext-nacl)# deny udp any 171.69.0.0 0.0.255.255 lt 1024
Switch(config-ext-nacl)# deny ip any any log
Switch(config-ext-nacl)# exit
```

The *Internet_filter* ACL is applied to outgoing traffic and the *marketing_group* ACL is applied to incoming traffic on a Layer 3 port.

```
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# no switchport
Switch(config-if)# ip address 2.0.5.1 255.255.255.0
Switch(config-if)# ip access-group Internet_filter out
Switch(config-if)# ip access-group marketing_group in
```

Applying Time Range to an IP ACL: Example

This example denies HTTP traffic on IP on Monday through Friday between the hours of 8:00 a.m. and 6:00 p.m (18:00). The example allows UDP traffic only on Saturday and Sunday from noon to 8:00 p.m. (20:00).

```
Switch(config)# time-range no-http
Switch(config)# periodic weekdays 8:00 to 18:00
!
Switch(config)# time-range udp-yes
Switch(config)# periodic weekend 12:00 to 20:00
!
Switch(config)# ip access-list extended strict
Switch(config-ext-nacl)# deny tcp any any eq www time-range no-http
Switch(config-ext-nacl)# permit udp any any time-range udp-yes
!
Switch(config-ext-nacl)# exit
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# ip access-group strict in
```

Creating Commented IP ACL Entries: Examples

In this example of a numbered ACL, the workstation that belongs to Jones is allowed access, and the workstation that belongs to Smith is not allowed access:

```
Switch(config)# access-list 1 remark Permit only Jones workstation through
Switch(config)# access-list 1 permit 171.69.2.88
Switch(config)# access-list 1 remark Do not allow Smith workstation through
Switch(config)# access-list 1 deny 171.69.3.13
```

In this example of a numbered ACL, the Winter and Smith workstations are not allowed to browse the web:

```
Switch(config)# access-list 100 remark Do not allow Winter to browse the web
Switch(config)# access-list 100 deny host 171.69.3.85 any eq www
Switch(config)# access-list 100 remark Do not allow Smith to browse the web
Switch(config)# access-list 100 deny host 171.69.3.13 any eq www
```

In this example of a named ACL, the Jones subnet is not allowed access:

```
Switch(config)# ip access-list standard prevention
Switch(config-std-nacl)# remark Do not allow Jones subnet through
Switch(config-std-nacl)# deny 171.69.0.0 0.0.255.255
```

In this example of a named ACL, the Jones subnet is not allowed to use outbound Telnet:

```
Switch(config)# ip access-list extended telnetting
Switch(config-ext-nacl)# remark Do not allow Jones subnet to telnet out
Switch(config-ext-nacl)# deny tcp 171.69.0.0 0.0.255.255 any eq telnet
```

Configuring ACL Logging: Examples

Two variations of logging are supported on router ACLs. The **log** keyword sends an informational logging message to the console about the packet that matches the entry; the **log-input** keyword includes the input interface in the log entry.

In this example, standard named access list *stan1* denies traffic from 10.1.1.0 0.0.0.255, allows traffic from all other sources, and includes the **log** keyword.

```
Switch(config)# ip access-list standard stan1
Switch(config-std-nacl)# deny 10.1.1.0 0.0.0.255 log
Switch(config-std-nacl)# permit any log
Switch(config-std-nacl)# exit
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# ip access-group stan1 in
Switch(config-if)# end
Switch# show logging
Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
    Console logging: level debugging, 37 messages logged
    Monitor logging: level debugging, 0 messages logged
    Buffer logging: level debugging, 37 messages logged
    File logging: disabled
    Trap logging: level debugging, 39 message lines logged

Log Buffer (4096 bytes):

00:00:48: NTP: authentication delay calculation problems

<output truncated>

00:09:34:%SEC-6-IPACCESSLOGS:list stan1 permitted 0.0.0.0 1 packet
00:09:59:%SEC-6-IPACCESSLOGS:list stan1 denied 10.1.1.15 1 packet
00:10:11:%SEC-6-IPACCESSLOGS:list stan1 permitted 0.0.0.0 1 packet
```

This example is a named extended access list *ext1* that permits ICMP packets from any source to 10.1.1.0 0.0.0.255 and denies all UDP packets.

```
Switch(config)# ip access-list extended ext1
Switch(config-ext-nacl)# permit icmp any 10.1.1.0 0.0.0.255 log
Switch(config-ext-nacl)# deny udp any any log
Switch(config-std-nacl)# exit
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# ip access-group ext1 in
```

Applying a MAC ACL to a Layer 2 Interface: Examples

This example shows how to create and display an access list named *mac1*, denying only EtherType DECnet Phase IV traffic, but permitting all other types of traffic.

```
Switch(config)# mac access-list extended mac1
Switch(config-ext-macl)# deny any any decnet-iv
Switch(config-ext-macl)# permit any any
Switch(config-ext-macl)# end
```

```
Switch # show access-lists
Extended MAC access list mac1
    10 deny   any any decnet-iv
    20 permit any any
```

This example shows how to apply MAC access list *mac1* to a port to filter packets entering the port:

```
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# mac access-group mac1 in
```

Note: The **mac access-group** interface configuration command is only valid when applied to a physical Layer 2 interface.You cannot use the command on EtherChannel port channels.

After receiving a packet, the switch checks it against the inbound ACL. If the ACL permits it, the switch continues to process the packet. If the ACL rejects the packet, the switch discards it. When you apply an undefined ACL to an interface, the switch acts as if the ACL has not been applied and permits all packets. Remember this behavior if you use undefined ACLs for network security.

Configuring VLAN Maps with ACLs

This section describes how to configure VLAN maps, which is the only way to control filtering within a VLAN. VLAN maps have no direction.

To filter traffic in a specific direction by using a VLAN map, you need to include an ACL with specific source or destination addresses. If there is a match clause for that type of packet (IP or MAC) in the VLAN map, the default action is to drop the packet if the packet does not match any of the entries within the map. If there is no match clause for that type of packet, the default is to forward the packet.

For complete syntax and usage information for the commands used in this section, see the command reference for this release.

To create a VLAN map and apply it to one or more VLANs, perform these steps:

1. Create the standard or extended IPv4 ACLs or named MAC extended ACLs that you want to apply to the VLAN. See the " Creating Standard and Extended IPv4 ACLs" section and the " Creating a VLAN Map" section.

2. Enter the vlan access-map global configuration command to create a VLAN ACL map entry.

3. In access-map configuration mode, optionally enter an **action–forward** (the default) or **drop**–and enter the **match** command to specify an IP packet or a non-IP packet (with only a known MAC address) and to match the packet against one or more ACLs (standard or extended).

   Note: If the VLAN map is configured with a match clause for a type of packet (IP or MAC) and the map action is drop, all packets that match the type are dropped. If the VLAN map has no match clause, and the configured action is drop, all IP and Layer 2 packets are dropped.

4. Use the vlan filter global configuration command to apply a VLAN map to one or more VLANs.

VLAN Map Configuration Guidelines

Follow these guidelines when configuring VLAN maps:

- If there is no ACL configured to deny traffic on an interface and no VLAN map is configured, all traffic is permitted.

- Each VLAN map consists of a series of entries. The order of entries in an VLAN map is important. A packet that comes into the switch is tested against the first entry in the VLAN map. If it matches, the action specified for that part of the VLAN map is taken. If there is no match, the packet is tested against the next entry in the map.

- If the VLAN map has at least one match clause for the type of packet (IP or MAC) and the packet does not match any of these match clauses, the default is to drop the packet. If there is no match clause for that type of packet in the VLAN map, the default is to forward the packet.
- The system might take longer to boot up if you have configured a very large number of ACLs.
- Logging is not supported for VLAN maps.
- When a switch has an IP access list or MAC access list applied to a Layer 2 interface, and you apply a VLAN map to a VLAN that the port belongs to, the port ACL takes precedence over the VLAN map.
- If VLAN map configuration cannot be applied in hardware, all packets in that VLAN must be bridged and routed by software.
- You can configure VLAN maps on primary and secondary VLANs. However, we recommend that you configure the same VLAN maps on private-VLAN primary and secondary VLANs.
- When a frame is Layer-2 forwarded within a private VLAN, the same VLAN map is applied at the ingress side and at the egress side. When a frame is routed from inside a private VLAN to an external port, the private-VLAN map is applied at the ingress side.
  - For frames going upstream from a host port to a promiscuous port, the VLAN map configured on the secondary VLAN is applied.
  - For frames going downstream from a promiscuous port to a host port, the VLAN map configured on the primary VLAN is applied.

  To filter out specific IP traffic for a private VLAN, you should apply the VLAN map to both the primary and secondary VLANs. For more information about private VLANs, see Chapter 19 " Configuring Private VLANs."

For configuration examples, see the " Using VLAN Maps in Your Network" section.

For information about using both router ACLs and VLAN maps, see the " VLAN Maps and Router ACL Configuration Guidelines" section.

Creating a VLAN Map

Each VLAN map consists of an ordered series of entries. Beginning in privileged EXEC mode, follow these steps to create, add to, or delete a VLAN map entry:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**vlan access-map** name [number]	Create a VLAN map, and give it a name and (optionally) a number. The number is the sequence number of the entry within the map. When you create VLAN maps with the same name, numbers are assigned sequentially in increments of 10. When modifying or deleting maps, you can enter the number of the map entry that you want to modify or delete. Entering this command changes to access-map configuration mode.
Step 3	**action {drop | forward}**	(Optional) Set the action for the map entry. The default is to forward.
Step 4	**match {ip | mac} address** {name | number} [name | number]	Match the packet (using either the IP or MAC address) against one or more standard or extended access lists. Note that packets are only matched against access lists of the correct protocol type. IP packets are matched against standard or extended IP access lists. Non-IP packets are only matched against named MAC extended access lists.

Step 5	**end**	Return to global configuration mode.
Step 6	**show running-config**	Display the access list configuration
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the no vlan access-map name global configuration command to delete a map. Use the no vlan access-map name number global configuration command to delete a single sequence entry from within the map.

Use the no action access-map configuration command to enforce the default action, which is to forward.

VLAN maps do not use the specific permit or deny keywords. To deny a packet by using VLAN maps, create an ACL that would match the packet, and set the action to drop. A permit in the ACL counts as a match. A deny in the ACL means no match.

Examples of ACLs and VLAN Maps

These examples show how to create ACLs and VLAN maps for specific purposes.

Example 1

This example shows how to create an ACL and a VLAN map to deny a packet. In the first map, any packets that match the ip1 ACL (TCP packets) would be dropped. You first create the ip1ACL to permit any TCP packet and no other packets. Because there is a match clause for IP packets in the VLAN map, the default action is to drop any IP packet that does not match any of the match clauses.

```
Switch(config)# ip access-list extended ip1
Switch(config-ext-nacl)# permit tcp any any
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map map_1 10
Switch(config-access-map)# match ip address ip1
Switch(config-access-map)# action drop
```

This example shows how to create a VLAN map to permit a packet. ACL *ip2* permits UDP packets and any packets that match the ip2 ACL are forwarded. In this map, any IP packets that did not match any of the previous ACLs (that is, packets that are not TCP packets or UDP packets) would get dropped.

```
Switch(config)# ip access-list extended ip2
Switch(config-ext-nacl)# permit udp any any
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map map_1 20
Switch(config-access-map)# match ip address ip2
Switch(config-access-map)# action forward
```

Example 2

In this example, the VLAN map has a default action of drop for IP packets and a default action of forward for MAC packets. Used with standard ACL 101 and extended named access lists igmp-match and tcp-match, the map will have the following results:

- Forward all UDP packets
- Drop all IGMP packets
- Forward all TCP packets
- Drop all other IP packets

- Forward all non-IP packets

```
Switch(config)# access-list 101 permit udp any any
Switch(config)# ip access-list extended igmp-match
Switch(config-ext-nacl)# permit igmp any any
Switch(config)# ip access-list extended tcp-match
Switch(config-ext-nacl)# permit tcp any any
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map drop-ip-default 10
Switch(config-access-map)# match ip address 101
Switch(config-access-map)# action forward
Switch(config-access-map)# exit
Switch(config)# vlan access-map drop-ip-default 20
Switch(config-access-map)# match ip address igmp-match
Switch(config-access-map)# action drop
Switch(config-access-map)# exit
Switch(config)# vlan access-map drop-ip-default 30
Switch(config-access-map)# match ip address tcp-match
Switch(config-access-map)# action forward
```

Example 3

In this example, the VLAN map has a default action of drop for MAC packets and a default action of forward for IP packets. Used with MAC extended access lists good-hosts and good-protocols, the map will have the following results:

- Forward MAC packets from hosts 0000.0c00.0111 and 0000.0c00.0211
- Forward MAC packets with decnet-iv or vines-ip protocols
- Drop all other non-IP packets
- Forward all IP packets

```
Switch(config)# mac access-list extended good-hosts
Switch(config-ext-macl)# permit host 000.0c00.0111 any
Switch(config-ext-macl)# permit host 000.0c00.0211 any
Switch(config-ext-nacl)# exit
Switch(config)# mac access-list extended good-protocols
Switch(config-ext-macl)# permit any any decnet-ip
Switch(config-ext-macl)# permit any any vines-ip
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map drop-mac-default 10
Switch(config-access-map)# match mac address good-hosts
Switch(config-access-map)# action forward
Switch(config-access-map)# exit
Switch(config)# vlan access-map drop-mac-default 20
Switch(config-access-map)# match mac address good-protocols
Switch(config-access-map)# action forward
```

Example 4

In this example, the VLAN map has a default action of drop for all packets (IP and non-IP). Used with access lists tcp-match and good-hosts from Examples 2 and 3, the map will have the following results:

- Forward all TCP packets
- Forward MAC packets from hosts 0000.0c00.0111 and 0000.0c00.0211
- Drop all other IP packets
- Drop all other MAC packets

```
Switch(config)# vlan access-map drop-all-default 10
Switch(config-access-map)# match ip address tcp-match
Switch(config-access-map)# action forward
Switch(config-access-map)# exit
Switch(config)# vlan access-map drop-all-default 20
Switch(config-access-map)# match mac address good-hosts
Switch(config-access-map)# action forward
```

Applying a VLAN Map to a VLAN

Beginning in privileged EXEC mode, follow these steps to apply a VLAN map to one or more VLANs:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**vlan filter** mapname **vlan-list** list	Apply the VLAN map to one or more VLAN IDs. The list can be a single VLAN ID (22), a consecutive list (10-22), or a string of VLAN IDs (12, 22, 30). **Note:** Spaces around the comma and hyphen are optional.
Step 3	**show running-config**	Display the access list.
Step 4	**copy running-config startup config**	(Optional) Save your entries in the configuration file.

To remove the VLAN map, use the **no vlan filter** *mapname* **vlan-list** *li*st global configuration command.

Example below shows how to apply VLAN map 1 to VLANs 20 thru 22:

Switch(config)# **vlan filter map 1 vlan-list 20-22**

Using VLAN Maps in Your Network

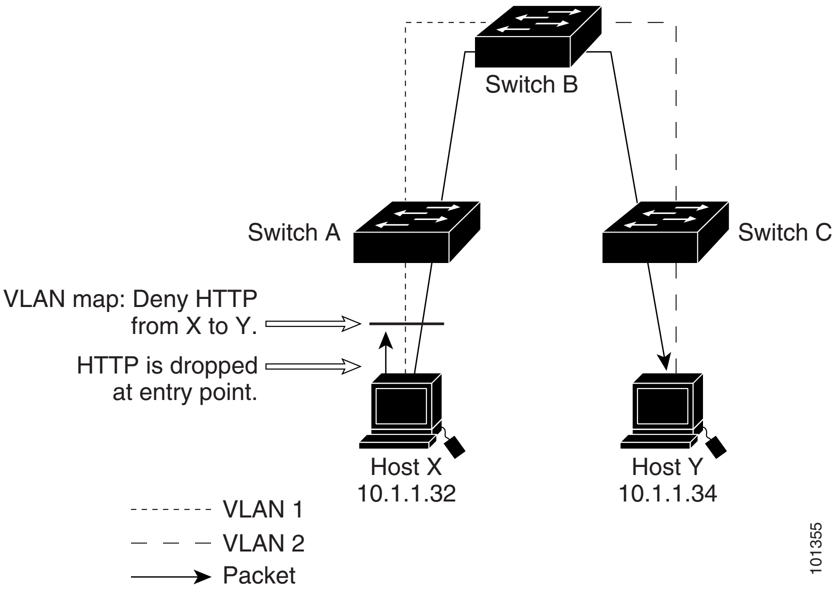
These sections describe some typical uses for VLAN maps:

- Wiring Closet Configuration
- Denying Access to a Server on Another

Wiring Closet Configuration

In a wiring closet configuration, routing might not be enabled on the switch. In this configuration, the switch can still support a VLAN map and a QoS classification ACL. In Figure 37-4, assume that Host X and Host Y are in different VLANs and are connected to wiring closet switches A and C. Traffic from Host X to Host Y is eventually being routed by Switch B, a Layer 3 switch with routing enabled. Traffic from Host X to Host Y can be access-controlled at the traffic entry point, Switch A.

Figure 77 Wiring Closet Configuration



If you do not want HTTP traffic switched from Host X to Host Y, you can configure a VLAN map on Switch A to drop all HTTP traffic from Host X (IP address 10.1.1.32) to Host Y (IP address 10.1.1.34) at Switch A and not bridge it to Switch B.

First, define the IP access list http that permits (matches) any TCP traffic on the HTTP port.

```
Switch(config)# ip access-list extended http
Switch(config-ext-nacl)# permit tcp host 10.1.1.32 host 10.1.1.34 eq www
Switch(config-ext-nacl)# exit
```

Next, create VLAN access map ‘map2’ so that traffic that matches the http access list is dropped and all other IP traffic is forwarded.

```
Switch(config)# vlan access-map map2 10
Switch(config-access-map)# match ip address http
Switch(config-access-map)# action drop
Switch(config-access-map)# exit
Switch(config)# ip access-list extended match_all
Switch(config-ext-nacl)# permit ip any any
Switch(config-ext-nacl)# exit
Switch(config)# vlan access-map map2 20
Switch(config-access-map)# match ip address match_all
Switch(config-access-map)# action forward
```

Then, apply VLAN access map map2 to VLAN 1.

```
Switch(config)# vlan filter map2 vlan 1
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*
Cisco IOS multicast commands	*Cisco IOS IP Command Reference, Volume 3 of 3:Multicast*
Cisco IOS IP Addressing and Services configuration Cisco IOS ACL configuration	*Cisco IOS IP Configuration Guide* *Cisco IOS IP Command Reference, Volume 1 of 3: Addressing and Services* *Cisco IOS Security Configuration Guide*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	–

MIBs

MIBs	MIBs Link
–	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	–

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Configuring QoS

This chapter describes how to configure quality of service (QoS) by using the modular QoS command-line interface (CLI), or MQC, commands on the Cisco IE switch. With QoS, you can provide preferential treatment to certain types of traffic at the expense of others. When QoS is not configured, the switch offers best-effort service to each packet, regardless of the packet contents or size. It sends the packets without any assurance of reliability, delay bounds, or throughput. MQC provides a comprehensive hierarchical configuration framework for prioritizing or limiting specific streams of traffic.

Note: IPv6 QoS is not supported.

For more information about Cisco IOS MQC commands, see the "Cisco IOS Quality of Service Solutions Command Reference" at this site:

http://www.cisco.com/en/US/docs/ios/12_2/qos/command/reference/fqos_r.html

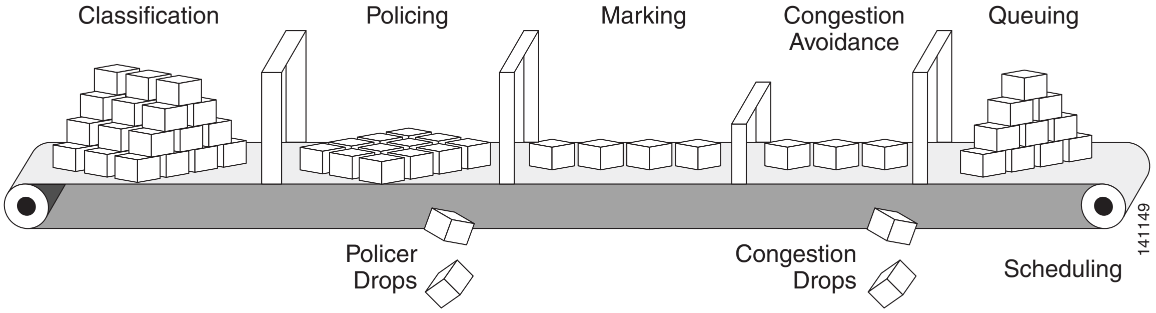
Understanding QoS

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.

When you configure the QoS feature, you can select specific network traffic, prioritize it according to its relative importance, and use traffic-management techniques to provide preferential treatment. Implementing QoS in your network makes network performance more predictable and bandwidth utilization more effective.

Figure 78 on page 613 shows the MQC model.

Figure 78 Modular QoS CLI Model



Basic QoS includes these actions.

- Packet classification organizes traffic on the basis of whether or not the traffic matches a specific criteria. When a packet is received, the switch identifies all key packet fields: class of service (CoS), Differentiated Services Code Point (DSCP), or IP precedence. The switch classifies the packet based on this content or based on an access-control list lookup. For more information, see Classification, page 617.

- Packet policing determines whether a packet is in or out of profile by comparing the rate of the incoming traffic to the configured policer. You can control the traffic flow for packets that conform to or exceed the configured policer. You can configure a committed information rate (CIR) and peak information rate (PIR) and set actions to perform on packets that conform to the CIR and PIR (conform-action), packets that conform to the PIR, but not the CIR (exceed-action), and packets that exceed the PIR value (violate-action). For more information, see Policing, page 626.

- Packet prioritization or marking evaluates the classification and policer information to determine the action to take. All packets that belong to a classification can be remarked. When you configure a policer, packets that meet or exceed the permitted bandwidth requirements (bits per second) can be conditionally passed through, dropped, or reclassified. For more information, see Marking, page 631.

- Congestion management uses queuing and scheduling algorithms to queue and sort traffic that is leaving a port. The switch supports these scheduling and traffic-limiting features: class-based weighted fair queuing (CBWFQ), class-based traffic shaping, port shaping, and class-based priority queuing. You can provide guaranteed bandwidth to a particular class of traffic while still servicing other traffic queues. For more information, see Congestion Management and Scheduling, page 635.

- Queuing on the switch is enhanced with the weighted tail-drop (WTD) algorithm, a congestion-avoidance mechanism. WTD differentiates traffic classes and regulates the queue size (in number of packets) based on the classification. For more information, see Congestion Avoidance and Queuing, page 639.

This section includes information about these topics:

Modular QoS CLI

Modular QoS CLI (MQC) allows users to create traffic policies and attach these policies to interfaces. A traffic policy contains a traffic class and one or more QoS features. You use a traffic class to classify traffic, and the QoS features in the traffic policy determine how to treat the classified traffic.

Modular QoS CLI configuration includes these steps:

1. Define a traffic class.

Use the **class-map** [**match-all** | **match-any**] *class-map-name* global configuration command to define a traffic class and to enter class-map configuration mode. A traffic class contains three elements: a name, an instruction on how to evaluate the configured **match** commands (if more than one match command is configured in the class map), and a series of **match** commands

- You name the traffic class in the **class-map** command line to enter class-map configuration mode.

- You can optionally include keywords to evaluate these match commands by entering **class-map match-any** or **class-map match-all**. If you specify **match-any**, the traffic being evaluated must match *one* of the specified criteria. If you specify **match-all**, the traffic being evaluated must match *all* of the specified criteria. A **match-all** class map can contain only one match statement, but a **match-any** class map can contain multiple match statements.

  If you do not enter **match-all** or **match-any**, the default is to match all.

- You use the **match** class-map configuration commands to specify criteria for classifying packets. If a packet matches the specified criteria, that packet is considered a member of the class and is forwarded according to the QoS specifications set in the traffic policy. Packets that fail to meet any of the matching criteria are classified as members of the default traffic class.

2. Create a traffic policy to associate the traffic class with one or more QoS features.

You use the **policy-map** *policy-map-name* global configuration command to create a traffic policy and to enter policy-map configuration mode. A traffic policy defines the QoS features to associate with the specified traffic class. A traffic policy contains three elements: a name, a traffic class (specified with the **class** policy-map configuration command), and the QoS policies configured in the class.

- You name the traffic policy in the **policy-map** command line to enter policy-map configuration mode.
- In policy-map configuration mode, enter the name of the traffic class used to classify traffic to the specified policy, and enter policy-map class configuration mode.
- In policy-map class configuration mode, you can enter the QoS features to apply to the classified traffic. These include using the **set**, **police**, or **police aggregate** commands for input policy maps or the **bandwidth**, **priority**, **queue-limit** or **shape average** commands for output policy maps.

Note: A packet can match only one traffic class within a traffic policy. If a packet matches more than one traffic class in the traffic policy, the first traffic class defined in the policy is used. To configure more than one match criterion for packets, you can associate multiple traffic classes with a single traffic policy.

3. Attach the traffic policy to an interface.

You use the **service-policy** interface configuration command to attach the policy map to an interface for packets entering or leaving the interface. You must specify whether the traffic policy characteristics should be applied to incoming or outgoing packets. For example, entering the **service-policy output class1** interface configuration command attaches all the characteristics of the traffic policy named *class1* to the specified interface. All packets leaving the specified interface are evaluated according to the criteria specified in the traffic policy named *class1*.

Input and Output Policies

Policy maps are either input policy maps or output policy maps, attached to packets as they enter or leave the switch by service policies applied to interfaces. Input policy maps perform policing and marking on received traffic. Policed packets can be dropped or reduced in priority (marked down) if they exceed the maximum permitted rates. Output policy maps perform scheduling and queuing on traffic as it leaves the switch.

Input policies and output policies have the same basic structure; the difference is in the characteristics that they regulate. Figure 79 on page 616 shows the relationship of input and output policies.

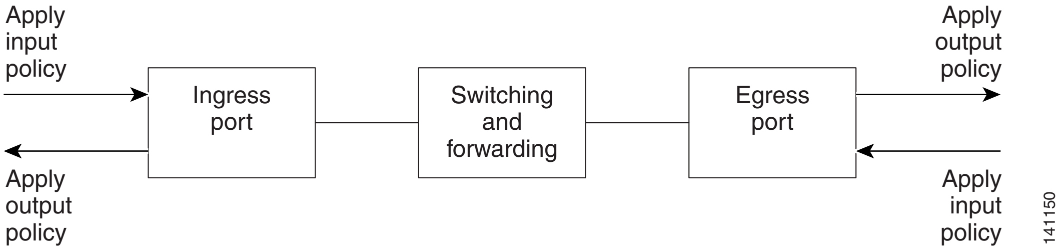
You can configure a maximum of 256 policy maps.

The number of configurable policer profiles on the switch is 256. The number of supported policer instances on the switch is 1024 minus 1 more than the number of interfaces on the switch. On a 24-port switch, the number of available policer instances is 999. You can use a policer profile in multiple instances.

You can apply one input policy map and one output policy map to an interface.

When CPU protection is enabled (the default), you can configure 45 ingress policers per port. If you disable CPU protection by entering the **no policer cpu uni all** global configuration command and reloading the switch, you can configure a maximum of 63 policers per port (62 on every 4th port) for user-defined classes and one for class-default. You can enter the **show policer cpu uni-eni** {**drop** | **rate**} privileged EXEC command to see if CPU protection is enabled.

Figure 79 Input and Output Policy Relationship



Input Policy Maps

Input policy map classification criteria include matching a CoS, a DSCP, or an IP precedence value or matching an access control list (ACL) or VLAN ID (for per-port, per-VLAN QoS). Input policy maps can have any of these actions:

- Setting or marking a CoS, a DSCP, an IP precedence, or QoS group value
- Individual policing
- Aggregate policing

Only input policies provide matching on access groups or VLAN IDs, and only output policies provide matching on QoS groups. You can assign a QoS group number in an input policy and match it in the output policy. The class **class-default** is used in a policy map for any traffic that does not explicitly match any other class in the policy map. Input policy maps do not support queuing and scheduling keywords, such as **bandwidth**, **queue-limit**, **priority**, and **shape average**.

An input policy map can have a maximum of 64 classes plus **class-default**. You can configure a maximum of 64 classes in an input policy.

Output Policy Maps

Output policy map classification criteria include matching a CoS, a DSCP, an IP precedence, or a QoS group value. Output policy maps can have any of these actions:

- Queuing (**queue-limit**)
- Scheduling (**bandwidth**, **priority**, and **shape average**)

Output policy maps do not support matching of access groups. You can use QoS groups as an alternative by matching the appropriate access group in the input policy map and setting a QoS group. In the output policy map, you can then match the QoS group. See Classification Based on QoS Groups, page 622 for more information.

Output policies do not support marking or policing (except in the case of priority with policing). There is no egress packet marking on the switch (no **set** command in an output policy).

The class **class-default** is used in a policy map for any traffic that does not explicitly match any other class in the policy map. There can be a maximum of four classes in the output policy map (including class-default) because egress ports have a maximum of four queues.

An output policy map attached to an egress port can match only the packets that have already been matched by an input policy map attached to the ingress port for the packets. You can attach an output policy map to any or all ports on the switch. The switch supports configuration and attachment of a unique output policy map for each port. However, these output policy maps can contain only three unique configurations of queue limits. These three unique queue-limit configurations can be included in as many output policy maps as there are ports on the switch. There are no limitations on the configurations of bandwidth, priority, or shaping.

You can configure the output policy classification criteria for CPU-generated traffic by using the **cpu traffic qos** [**cos** *value* | **dscp** *value* | **precedence** *value* | **qos-group** *value*] global configuration command.

Classification

Classification distinguishes one kind of traffic from another by examining the fields in the packet header. When a packet is received, the switch examines the header and identifies all key packet fields. A packet can be classified based on an ACL, on the DSCP, the CoS, or the IP precedence value in the packet, or by the VLAN ID. Figure 80 on page 618 has examples of classification information carried in a Layer 2 or a Layer 3 IP packet header, using six bits from the deprecated IP type of service (ToS) field to carry the classification information.

- On ports configured as Layer 2 IEEE 802.1Q trunks, all traffic is in 802.1Q frames except for traffic in the native VLAN. Layer 2 802.1Q frame headers have a 2-byte Tag Control Information field that carries the CoS value, called the User Priority bits, in the three most-significant bits, and the VLAN ID value in the 12 least-significant bits. Other frame types cannot carry Layer 2 CoS values.

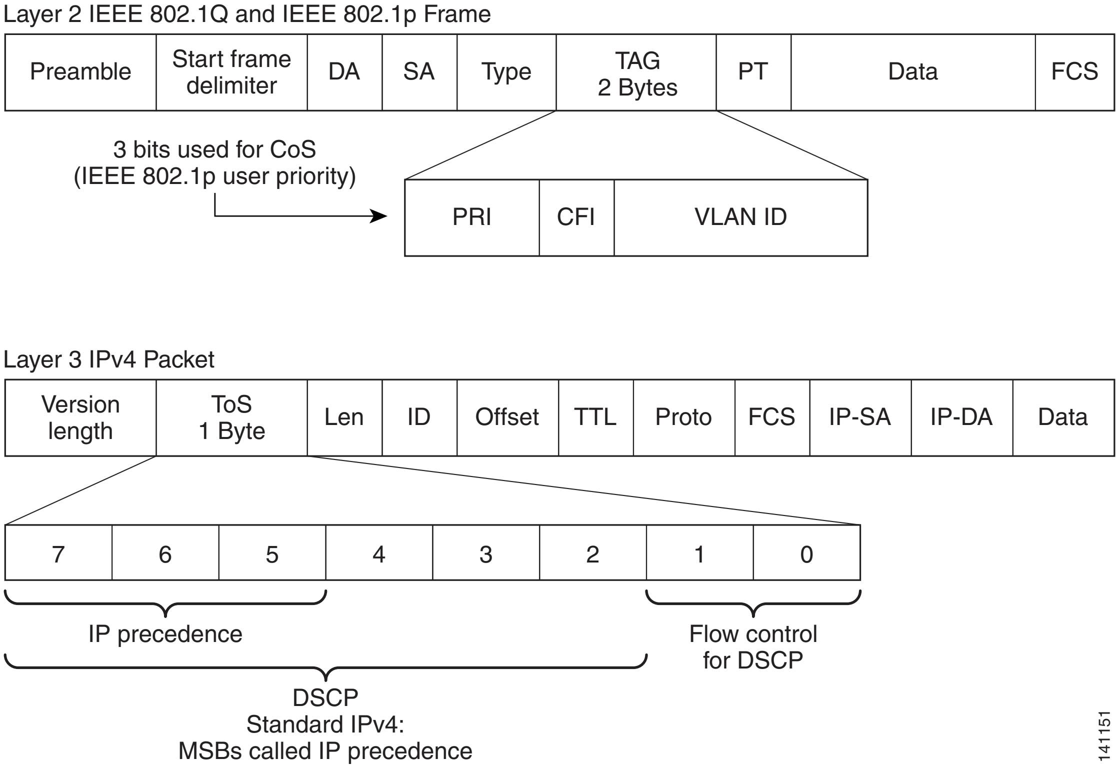
  Layer 2 CoS values range from 0 to 7.

- 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 are backward-compatible with IP precedence values.

  IP precedence values range from 0 to 7. DSCP values range from 0 to 63.

- Output remarking is based on the Layer 2 or Layer 3 marking type, marking value and packet type.

Figure 80 QoS Classification Layers in Frames and Packets



These sections contain additional information about classification:

- Class Maps, page 618
- The match Command, page 619
- Classification Based on Layer 2 CoS, page 619
- Classification Based on IP Precedence, page 619
- Classification Based on IP DSCP, page 620
- Classification Comparisons, page 621
- Classification Based on QoS ACLs, page 622
- Classification Based on QoS Groups, page 622
- Classification Based on VLAN IDs, page 623

Class Maps

As explained previously, you use an MQC class map to name a specific traffic flow (or class) and to isolate it from all other traffic. A class map defines the criteria used to match against a specific traffic flow to further classify it. If you have more than one type of traffic that you want to classify, you can create another class map and use a different name. When you enter the **class-map** command with a class-map name, the switch enters the class-map configuration mode. In this mode, you define the match criterion for the traffic by using the **match** class-map configuration command. After a packet is matched against the class-map criteria, it is acted on by the associated action specified in a policy map.

You can match more than one criterion for classification. You can also create a class map that requires that all matching criteria in the class map be in the packet header by using the **class map match-all** *class-map name* global configuration command to enter class map configuration mode.

Note: You can configure only one match entry in a **match-all** class map.

You can use the **class map match-any** *class-map name* global configuration command to define a classification with any of the listed criteria.

Note: If you do not enter **match-all** or **match-any**, the default is to match all. A match-all class map cannot have more than one classification criterion (match statement). A class map with no match condition has a default of **match all**.

The match Command

To configure the type of content used to classify packets, you use the **match** class-map configuration command to specify the classification criteria. If a packet matches the configured criteria, it belongs to a specific class and is forwarded according to the specified policy. For example, you can use the **match** class-map command with CoS, IP DSCP, and IP precedence values. These values are referred to as *markings* on a packet. You can also match an access group, a QoS group, or a VLAN ID or ID range for per-port, per-VLAN QoS.

- For an input policy map, you cannot configure an IP classification (**match ip dscp**, **match ip precedence**, **match access-group** for an IP ACL) and a non-IP classification (**match cos** or **match access-group** for a MAC ACL) in the same policy map or class map.
- When an input policy map with only Layer 2 classification is attached to a routed port or a switch port containing a routed switch virtual interface (SVI), the service policy acts only on switching eligible traffic and not on routing eligible traffic.
- On an 802.1Q tunnel port, you can use only an input policy map with Layer 2 classification based on MAC ACLs to classify traffic. Input policy maps with Layer 3 classification, match Layer 2 CoS classification, or per-port, per-VLAN policies are not supported on tunnel ports.
- In an output policy map, no two class maps can have the same classification criteria, that is, the same match qualifiers and values.

This example shows how to create a class map *example* to define a class that matches any of the listed criteria. In this example, if a packet is received with the DSCP equal to 32 or a 40, the packet is identified (classified) by the class map.

```
Switch(config)# class-map match-any example
Switch(config-cmap)# match ip dscp 32
Switch(config-cmap)# match ip dscp 40
Switch(config-cmap)# exit
```

Classification Based on Layer 2 CoS

You can use the **match** command to classify Layer 2 traffic based on the CoS value, which ranges from 0 to 7.

Note: A **match cos** command is supported only on Layer 2 802.1Q trunk ports.

This example shows how to create a class map to match a CoS value of 5:

```
Switch(config)# class-map premium
Switch(config-cmap)# match cos 5
Switch(config-cmap)# exit
```

Classification Based on IP Precedence

You can classify IPv4 traffic based on the packet IP precedence values, which range from 0 to 7.

This example shows how to create a class map to match an IP precedence value of 4:

```
Switch(config)# class-map sample
Switch(config-cmap)# match ip precedence 4
Switch(config-cmap)# exit
```

Classification Based on IP DSCP

When you classify IPv4 traffic based on IP DSCP value, and enter the **match ip dscp** class-map configuration command, you have several classification options:

- Entering a specific DSCP value (0 to 63).
- Using the Default service, which corresponds to an IP precedence and DSCP value of 0. The default per-hop behavior (PHB) is usually best-effort service.
- Using Assured Forwarding (AF) by entering the binary representation of the DSCP value. AF sets the relative probability that a specific class of packets is forwarded when congestion occurs and the traffic does not exceed the maximum permitted rate. AF *per-hop behavior* provides delivery of IP packets in four different AF classes: AF11-13 (the highest), AF21-23, AF31-33, and AF41-43 (the lowest). Each AF class could be allocated a specific amount of buffer space and drop probabilities, specified by the binary form of the DSCP number. When congestion occurs, the drop precedence of a packet determines the relative importance of the packet within the class. An AF41 provides the best probability of a packet being forwarded from one end of the network to the other.
- Entering Class Selector (CS) service values of 1 to 7, corresponding to IP precedence bits in the ToS field of the packet.
- Using Expedited Forwarding (EF) to specify a low-latency path. This corresponds to a DSCP value of 46. EF services use priority queuing to preempt lower priority traffic classes.

This display shows the available classification options:

```
Switch(config-cmap)# match ip dscp ?
  <0-63>   Differentiated services codepoint value
  af11     Match packets with AF11 dscp (001010)
  af12     Match packets with AF12 dscp (001100)
  af13     Match packets with AF13 dscp (001110)
  af21     Match packets with AF21 dscp (010010)
  af22     Match packets with AF22 dscp (010100)
  af23     Match packets with AF23 dscp (010110)
  af31     Match packets with AF31 dscp (011010)
  af32     Match packets with AF32 dscp (011100)
  af33     Match packets with AF33 dscp (011110)
  af41     Match packets with AF41 dscp (100010)
  af42     Match packets with AF42 dscp (100100)
  af43     Match packets with AF43 dscp (100110)
  cs1      Match packets with CS1(precedence 1) dscp (001000)
  cs2      Match packets with CS2(precedence 2) dscp (010000)
  cs3      Match packets with CS3(precedence 3) dscp (011000)
  cs4      Match packets with CS4(precedence 4) dscp (100000)
  cs5      Match packets with CS5(precedence 5) dscp (101000)
  cs6      Match packets with CS6(precedence 6) dscp (110000)
  cs7      Match packets with CS7(precedence 7) dscp (111000)
  default  Match packets with default dscp (000000)
  ef       Match packets with EF dscp (101110)
```

For more information on DSCP prioritization, see RFC-2597 (AF per-hop behavior), RFC-2598 (EF), or RFC-2475 (DSCP).

802.1Q Tunneling CoS Mapping

The switch supports VLAN mapping from the customer VLAN-ID (C-VLAN) to a service-provider VLAN-ID (S-VLAN). For QoS, the switch can set the service-provider CoS (S-CoS) from either the customer CoS (C-CoS) or the customer DSCP (C-DSCP) value, and can map the inner CoS to the outer CoS for any traffic with traditional 802.1Q tunneling (QinQ) or selective QinQ VLAN mapping. This default allows copying the customer CoS into the service provider network.

The switch supports C-CoS to S-CoS propagation for traditional QinQ and for selective QinQ on trunk ports. This is the default behavior and does not require configuration. When you configure traditional QinQ or selective QinQ on Layer 2 trunk ports using 1-to-2 VLAN mapping, the switch also allows setting the S-CoS from C-DSCP.

For traffic entering the switch on 802.1Q tunnel ports or trunk ports configured for VLAN mapping, the switch has the ability to examine the customer packet header and set the service-provider CoS value (S-CoS) from either the customer CoS value or the customer DSCP value.

Configuring CoS matching on 802.1Q mapped ports is handled in this way:

- On interfaces configured for 802.1Q tunneling (on tunnel or trunk ports) or selective 802.1Q (on trunk ports), the CoS value of the VLAN tag (inner VLAN or C-VLAN) received on the interface (C-CoS) is automatically reflected in the tunnel VLAN tag (outer VLAN or S-VLAN) by default.
- The **set cos** policy-map class configuration commands always apply to the outer-most VLAN tag after processing is complete, that is the S-VLAN-ID. For example, in 802.1Q tunnels, entering a **set cos** command changes only the CoS value of the outer tag of the encapsulated packet.
- When you configure a policy by entering the **match dscp** class map configuration command and you enter the **set cos** policy-map class configuration command for QinQ and selective QinQ mapping interfaces, a DSCP match sets the outer CoS of the encapsulated value.
- You can set DSCP based on matching the outer VLAN.
- If you enter the **match cos** command on interfaces configured for traditional QinQ or for selective QinQ mapping, the match is to the outer CoS, which is the reflected inner Cos (C-CoS).

Classification Comparisons

Table 57 on page 621 shows suggested IP DSCP, IP precedence, and CoS values for typical traffic types.

Table 57 Typical Traffic Classifications

Traffic Type	DSCP per-hop	DSCP (decimal)	IP Precedence	CoS
Voice-bearer—traffic in a priority queue or the queue with the highest service weight and lowest drop priority.	EF	46	5	5
Voice control—signalling traffic, related to call setup, from a voice gateway or a voice application server.	AF31	26	3	3
Video conferencing—in most networks, video conferencing over IP has similar loss, delay, and delay variation requirements as voice over IP traffic.	AF41	34	4	4
Streaming video—relatively high bandwidth applications with a high tolerance for loss, delay, and delay variation. Usually considered more important than regular background applications such as e-mail and web browsing.	AF13	14	1	1

Table 57 Typical Traffic Classifications (continued)

Traffic Type	DSCP per-hop	DSCP (decimal)	IP Precedence	CoS
Mission critical date (gold data)—delay-sensitive applications critical to the operation of an enterprise.				
Level 1 Level 2 Level 3	AF21 AF22 AF23	18 20 22	2 2 2	2 2 2
Less critical data (silver data)—noncritical, but relatively important data.				
Level 1 Level 2 Level 3	AF11 AF12 AF13	10 12 14	1 1 1	1 1 1
Best-effort data (bronze data)—other traffic, including all noninteractive traffic, regardless of importance.	Default	0	0	0
Less than best-effort data—noncritical, bandwidth-intensive data traffic given the least preference. This is the first traffic type to be dropped.				
Level 1 Level 2 Level 3		2 4 6	0 0 0	0 0 0

Classification Based on QoS ACLs

Packets can also be classified in input policy maps based on an ACL lookup. The ACL classification is communicated to an output policy by assigning a QoS group or number in the input policy map. To classify based on ACL lookup, you first create an IP or MAC ACL. Configure a class map and use the **match access-group** {*acl-number* | *acl name*} class-map configuration command, and attach the class map to a policy map.

Note: You cannot configure **match access-group** for an output policy map.

You can use IP standard, IP extended, or Layer 2 MAC ACLs to define a group of packets with the same characteristics (a class). You use the **access-list** global configuration command to configure IP ACLS to classify IP traffic based on Layer 3 and Layer 4 parameters. You use the **mac access-list extended** global configuration command to configure Layer 2 MAC ACLs to classify IP and non-IP traffic based on Layer 2 parameters.

Note: You cannot match IP fragments against configured IP extended ACLs to enforce QoS. IP fragments are sent as best-effort. IP fragments are denoted by fields in the IP header.

You can use only ACLs with a permit action in a **match access-group** command. ACLs with a deny action are never matched in a QoS policy.

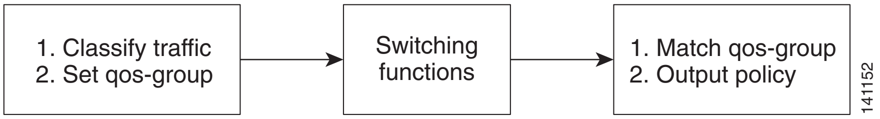
Note: Only one access-group is supported per class for an input policy map.

Classification Based on QoS Groups

A QoS group is an internal label used by the switch to identify packets as a members of a specific class. The label is not part of the packet header and is restricted to the switch that sets the label. QoS groups provide a way to tag a packet for subsequent QoS action without explicitly marking (changing) the packet. You can then communicate an ACL match from an input policy map to an output policy map.

A QoS group is identified at ingress and used at egress; it is assigned in an input policy to identify packets in an output policy. See on page 622. The QoS groups help aggregate different classes of input traffic for a specific action in an output policy.

Figure 81 QoS Groups



You can use QoS groups to aggregate multiple input streams across input classes and policy maps for the same QoS treatment on the egress port. Assign the same QoS group number in the input policy map to all streams that require the same egress treatment, and match to the QoS group number in the output policy map to specify the required queuing and scheduling actions.

You can also use QoS groups to identify traffic entering a particular interface if the traffic must be treated differently at the output based on the input interface.

You can use QoS groups to configure per-port, per-VLAN QoS output policies on the egress interface for bridged traffic on the VLAN. Assign a QoS group number to a VLAN on the ingress interface by configuring a per-port, per-VLAN input policy. Then use the same QoS-group number for classification at the egress. Because the VLAN of bridged traffic does not change during forwarding through the switch, the QoS-group number assigned to the ingress VLAN can be used on the egress interface to identify the same VLAN.

You can use the **cpu traffic qos** [**cos** *value* | **dscp** *value* | **precedence** *value* | **qos-group** *value*] global configuration command to configure a QoS group number for CPU-generated traffic.

Independently you can assign QoS-group numbers at the ingress to any combination of interfaces, VLANs, traffic flows, and aggregated traffic. To assign QoS-group numbers, configure a QoS group marking in an input policy map, along with any other marking or policing actions required in the input policy map for the same service class. This allows the input marking and policing functions to be decoupled from the egress classification function if necessary because only the QoS group must be used for egress classification.

To communicate an ACL classification to an output policy, you assign a QoS number to specify packets at ingress. This example identifies specific packets as part of QoS group 1 for later processing in an output policy:

```
Switch(config)# policy-map in-gold-policy
Switch(config-pmap)# class in-class1
Switch(config-pmap-c)# set qos-group 1
Switch(config-cmap-c)# exit
Switch(config-cmap)# exit
```

You use the **set qos-group** command only in an input policy. The assigned QoS group identification is subsequently used in an output policy with no mark or change to the packet. You use the **match qos-group** in the output policy.

Note: You cannot configure **match qos-group** for an input policy map.

This example creates an output policy to match the QoS group created in the input policy map *in-gold-policy*. Traffic internally tagged as *qos-group 1* is identified and processed by the output policy.

```
Switch(config)# class-map out-class1
Switch(config-cmap)# match qos-group 1
Switch(config-cmap)# exit
```

The switch supports a maximum of 100 QoS groups.

Classification Based on VLAN IDs

With classification based on VLAN IDs, you can apply QoS policies to frames carried on a user-specified VLAN for a given interface. Per-VLAN classification is not required on access ports because access ports carry traffic for a single VLAN. If you try to attach an input per-port, per VLAN hierarchical policy to a port that is not a trunk port, the configuration is rejected.

The switch supports two policy levels: a *parent* level and a *child* level. With the QoS parent-child structure, you can reference a child policy in a parent policy to provide additional control of a specific traffic type. For per-port, per-VLAN QoS, the parent-level class map specifies only the VLAN match criteria, and the child-level class maps provide more detailed classification for frames matching the parent-level class map.You can configure multiple service classes at the parent level to match different combinations of VLANs, and you can apply independent QoS policies to each parent service class using any child policy map.

Note: A per-port, per-VLAN parent-level class map supports only a child-policy association; it does not allow any actions to be configured. In addition, for a parent-level class map, you cannot configure an action or a child-policy association for the class **class-default**.

Per-port, per-VLAN QoS has these limitations:

- You can apply a per-port, per-VLAN hierarchical policy map only to trunk ports.
- You can configure classification based on VLAN ID only in the parent level of a per-port, per-VLAN hierarchical policy map.
- When the child policy map attached to a VLAN or set of VLANs contains only Layer 3 classification (**match ip dscp**, **match ip precedence**, **match IP ACL**), you must be careful to ensure that these VLANs are not carried on any port other than the one on which this per-port, per-VLAN policy is attached. Not following this restriction could result in improper QoS behavior for traffic ingressing the switch on these VLANs.
- We also recommend that you restrict VLAN membership on the trunk ports to which the per-port, per-VLAN is applied by using the **switchport trunk allowed vlan** interface configuration command. Overlapping VLAN membership between trunk ports that have per-port, per-VLAN policies with Layer 3 classification could also result in unexpected QoS behavior.

In this example, the class maps in the child-level policy map specify matching criteria for voice, data, and video traffic, and the child policy map sets the action for input policing each type of traffic. The parent-level policy map specifies the VLANs to which the child policy maps are applied on the specified port.

```
Switch(config)# class-map match-any dscp-1 data
Switch(config-cmap)# match ip dscp 1
Switch(config-cmap)# exit
Switch(config)# class-map match-any dscp-23 video
Switch(config-cmap)# match ip dscp 23
Switch(config-cmap)# exit
Switch(config)# class-map match-any dscp-63 voice
Switch(config-cmap)# match ip dscp-63
Switch(config-cmap)# exit
Switch(config)# class-map match-any customer-1-vlan
Switch(config-cmap)# match vlan 100
Switch(config-cmap)# match vlan 200
Switch(config-cmap)# match vlan 300
Switch(config-cmap)# exit
```

Note: You can also enter the match criteria as **match vlan 100 200 300** with the same result.

```
Switch(config)# policy-map child policy-1
Switch(config-pmap)# class dscp-63 voice
Switch(config-pmap-c)# police cir 10000000 bc 50000
Switch(config-pmap-c)# conform-action set-cos-transmit 5
Switch(config-pmap-c)# exceed-action drop
Switch(config-pmap-c)# exit
Switch(config-pmap)# class dscp-1 data
Switch(config-pmap-c)# set cos 0
Switch(config-pmap-c)# exit
Switch(config-pmap)# class dscp-23 video
Switch(config-pmap-c)# set cos 4
Switch(config-pmap-c)# set ip precedence 4
Switch(config-pmap-c)# exit

Switch(config)# policy-map parent-customer-1
Switch(config-pmap)# class customer-1-vlan
Switch(config-pmap-c)# service-policy ingress-policy-1
Switch(config-pmap-c)# exit

Switch(config)# interface fastethernet0/1
Switch(config-if)# switchport mode trunk
Switch(config-if)# service-policy input customer-1-ingress
Switch(config-pmap-c)# exit
```

Note: Each per-port, per-VLAN parent policy class, except **class-default**, can have a child policy association.

see Configuring Per-Port Per-VLAN QoS with Hierarchical Input Policy Maps, page 665 for configuration information, including configuration guidelines and limitations.

Table Maps

You can use table maps to manage a large number of traffic flows with a single command. You can specify table maps in **set** commands and use them as mark-down mapping for the policers. You can also use table maps to map an incoming QoS marking to a replacement marking without having to configure a large number of explicit matches and sets. Table maps are used only in input policy maps.

Table maps can be used to:

- Correlate specific CoS, DSCP, or IP precedence values to specific CoS, DSCP, or IP precedence values
- Mark down a CoS, DSCP, or IP precedence value
- Assign defaults for unmapped values

A table map includes one of these default actions:

- **default** *default-value*—applies a specific default value (0 to 63) for all unmapped values
- **default copy**—maps all unmapped values to the equivalent value in another qualifier
- **default ignore**—makes no changes for unmapped values

This example creates a table to map specific CoS values to DSCP values. The **default** command maps all unmapped CoS values to a DSCP value of 63.

```
Switch(config)# table-map cos-dscp-tablemap
Switch(config-tablemap)# map from 5 to 46
Switch(config-tablemap)# map from 6 to 56
Switch(config-tablemap)# map from 7 to 57
Switch(config-tablemap)# default 63
Switch(config-tablemap)# exit
```

The switch supports a maximum of 256 unique table maps. You can enter up to 64 different **map from**-**to** entries in a table map. These table maps are supported on the switch:

- DSCP to CoS
- DSCP to precedence
- DSCP to DSCP
- CoS to DSCP
- CoS to precedence
- CoS to CoS

- Precedence to CoS
- Precedence to DSCP
- Precedence to precedence

Table maps modify only one parameter (CoS, IP precedence, or DSCP, whichever is configured) and are only effective when configured with a **set** command in a policy map or with a **conform-action** or **exceed-action** command in a police function. Individual policers also support the **violate-action** command, but aggregate policers do not support table maps with violate-action.

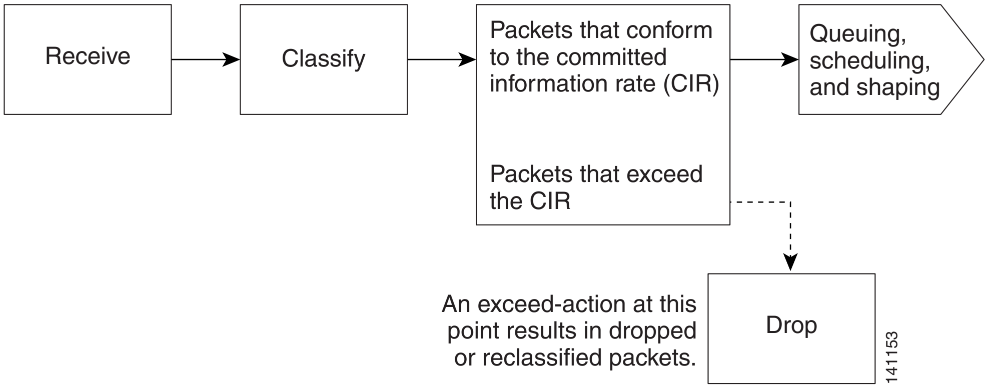
Table maps are not supported in output policy maps. For more information, set the Configuring Table Maps, page 650.

Policing

After a packet is classified, you can use policing as shown in Figure 82 on page 626 to regulate the class of traffic. The policing function limits the amount of bandwidth available to a specific traffic flow or prevents a traffic type from using excessive bandwidth and system resources. A policer identifies a packet as in or out of profile by comparing the rate of the inbound traffic to the configuration profile of the policer and traffic class. Packets that exceed the permitted average rate or burst rate are *out of profile* or *nonconforming*. These packets are dropped or modified (marked for further processing), depending on the policer configuration.

Policing is used primarily on receiving interfaces. You can attach a policy map with a policer only in an input service policy. The only policing allowed in an output policy map is in priority classes. See Unconditional Priority Policing, page 630.

Figure 82 Policing of Classified Packets


Receive
Classify
Packets that conform to the committed information rate (CIR)
Packets that exceed the CIR
Queuing, scheduling, and shaping
An exceed-action at this point results in dropped or reclassified packets.
Drop
141153

These sections describe the types of policing supported on the switch:

- Individual Policing, page 626
- Aggregate Policing, page 628
- Unconditional Priority Policing, page 630

Individual Policing

Individual policing applies only to input policy maps. In policy-map configuration mode, you enter the **class** command followed by class-map name, and enter policy-map class configuration mode.

Cisco Industrial Ethernet Switches support 1-rate, 2-color ingress policing and 2-rate, 3-color policing for individual or aggregate policing.

For 1-rate, 2-color policing, youYou use the **police** policy-map class configuration command to define the policer, the committed rate limitations of the traffic, committed burst size limitations of the traffic, and the action to take for a class of traffic that is below the limits (**conform-action**) and above the limits (**exceed-action**). If you do not specify burst size (bc), the system calculates an appropriate burst size value. The calculated value is appropriate for most applications. For more information, see Attaching a Traffic Policy to an Interface, page 651.

When you configure a 2-rate policer, in addition to configuring the committed information rate (CIR) for updating the first token bucket, you also configure the peak information rate (PIR) at which the second token bucket is updated. If you do not configure a PIR, the policer is a standard 1-rate, 2-color policer.

For 2-rate, 3-color policing, you can then optionally set actions to perform on packets that conform to the specified CIR and PIR (conform-action), packets that conform to the PIR, but not the CIR (exceed-action), and packets that exceed the PIR value (violate-action).

- If you set the CIR value equal to the PIR, a traffic rate that is less than or equal to the CIR is in the conform range. Traffic that exceeds the CIR is in the violate range.
- If you set the PIR greater than the CIR, a traffic rate less than the CIR is in the conform range. A traffic rate that exceeds the CIR but is less than or equal to the PIR is in the exceed range. A traffic rate that exceeds the PIR is in the violate range.
- If you do not configure a PIR, the policer is configured as a 1-rate, 2-color policer.

Setting the burst sizes too low can reduce throughput in situations with bursty traffic. Setting burst sizes too high can allow too high a traffic rate.

Note: The switch supports byte counters for byte-level statistics for conform, exceed, and violate classes in the **show policy-map interface** privileged EXEC command output.

To make the policy map effective, you attach it to a physical port by using the **service-policy input** interface configuration command. Policing is done only on received traffic, so you can only attach a policer to an input service policy.

You can use the **conform-action** and **exceed-action** policy-map class configuration commands or the **conform-action** and **exceed-action** policy-map class police configuration commands to specify the action to be taken when the packet conforms to or exceeds the specified traffic rate.

Conform actions are to send the packet without modifications, to set a new CoS, DSCP, or IP precedence value, or to set a QoS group value for classification at the egress. Exceed actions are to drop the packet, to send the packet without modification, to set a new CoS, DSCP, or IP precedence to a value, or to set a QoS group value for classification at the egress.

You can use the **conform-action**, **exceed-action**, and **violate-action** policy-map class configuration commands or the **conform-action**, **exceed-action**, and **violate-action** policy-map class police configuration commands to specify the action to be taken when the packet conforms to or exceeds the specified traffic rates. Conform, exceed, and violate actions are to drop the packet, to send the packet without modifications, to set a new CoS, DSCP, or IP precedence value, or to set a QoS group value for classification at the egress.

You can configure each marking action by using explicit values, table maps, or a combination of both. Table maps list specific traffic attributes and map (or convert) them to other attributes.

You can configure multiple conform and exceed actions simultaneously for each service class. You can configure multiple conform, exceed, and violate actions simultaneously for each service class. If you do not configure a **violate-action**, by default the violate class is assigned the same action as the **exceed-action**.

After you create a table map, you configure a policy-map policer to use the table map.

Note: When you use a table map in an input policy map, the protocol type for the **from**–action in the table map must be the same as the protocol type of the associated classification. For example, if a class map represents IP classification, the **from**–type action in the table map must be either **dscp** or **precedence**. If the class map represents a non-IP classification, the **from**–type action in the table map must be **cos**.

To configure multiple actions in a class, you can enter multiple conform or exceed action entries conform, exceed, or violate action entries in policy-map class police configuration mode, as in this example:

```
Switch(config)# policy-map map1
Switch(config-pmap)# class class1
Switch(config-pmap-c)# police 100000 500000
Switch(config-pmap-c-police)# conform-action set-cos-transmit 4
Switch(config-pmap-c-police)# conform-action set-dscp-transmit dscp table conform-dscp-to-dscp-mutation
Switch(config-pmap-c-police)# conform-action set-qos-transmit 10
Switch(config-pmap-c-police)# exceed-action set-cos-transmit 2
Switch(config-pmap-c-police)# exceed-action set-dscp-transmit dscp table exceed-dscp-to-dscp-mutation
Switch(config-pmap-c-police)# exceed-action set-qos-transmit 20
Switch(config-pmap-c-police)# exit
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
```

Aggregate Policing

Aggregate policing applies only to input policy maps. An aggregate policer differs from an individual policer because it is shared by multiple traffic classes within a policy map. Cisco Industrial Ethernet Switches support 1-rate, 2-color ingress policing and 2-rate, 3-color policing for aggregate policing.

You can use the **policer aggregate** global configuration command to set a policer for all traffic received or sent on a physical interface. When you configure an aggregate policer, you can configure specific burst sizes and conform and exceed actions. If you do not specify burst size (**bc**), the system calculates an appropriate burst size value. The calculated value is appropriate for most applications.

When you configure a 2-rate policer, in addition to configuring the committed information rate (CIR) for updating the first token bucket, you also configure the peak information rate (PIR) at which the second token bucket is updated. If you do not configure a PIR, the policer is a standard 1-rate, 2-color policer.

For 2-rate, 3-color policing, you can then optionally set actions to perform on packets that conform to the specified CIR and PIR (conform-action), packets that conform to the PIR, but not the CIR (exceed-action), and packets that exceed the PIR value (violate-action).

Note: If the conform action is set to drop, the exceed and violate actions are automatically set to drop. If the exceed action is set to drop, the violate action is automatically set to drop.

- If you set the CIR value equal to the PIR, a traffic rate that is less than or equal to the CIR is in the conform range. Traffic that exceeds the CIR is in the violate range.
- If you set the PIR greater than the CIR, a traffic rate less than the CIR is in the conform range. A traffic rate that exceeds the CIR but is less than or equal to the PIR is in the exceed range. A traffic rate that exceeds the PIR is in the violate range.
- If you do not configure PIR, the policer is configured as a 1-rate, 2-color policer.

Setting the burst sizes too low can result in less traffic than expected. Setting burst sizes too high can result in more traffic than expected.

You can configure multiple conform and exceed actions simultaneously for each service class. Conform actions are to send the packet without modifications, to set a QoS group value for classification at the egress, or to set a new CoS, DSCP, or IP precedence value. Exceed actions are to drop the packet, to send the packet without modification, to set a QoS group for classification at the egress, or to set a new CoS, DSCP, or IP precedence to a value. You can configure each marking conform or exceed action by using explicit values, using table maps, or using a combination of both. Table maps list specific traffic attributes and map (or convert) them to other attributes.

You can configure multiple conform, exceed, and violate actions simultaneously for each service class. You can use the **conform-action**, **exceed-action**, and **violate-action** policy-map class configuration commands or the **conform-action**, **exceed-action**, and **violate-action** policy-map class police configuration commands to specify the action to be taken when the packet conforms to or exceeds the specified traffic rates. Conform, exceed, and violate actions are to drop the packet, to send the packet without modifications, to set a new CoS, DSCP, or IP precedence value, or to set a QoS group value for classification at the egress.

Note: If the conform action is set to drop, the exceed and violate actions are automatically set to drop. If the exceed action is set to drop, the violate action is automatically set to drop.

You can configure each marking conform, exceed, or violate action by using explicit values, using table maps, or using a combination of both. If you do not configure a **violate-action**, by default the violate class is assigned the same action as the **exceed-action.**

Table maps list specific traffic attributes and map (or convert) them to other attributes. Table maps are not supported for **violate-action** for aggregate policing unless you configure a table map for exceed-action and no explicit action is configured for violate action.

After you create a table map, you configure a policy-map policer to use the table map.

Note: When you use a table map in an input policy map, the protocol type for the **from**-action in the table map must be the same as the protocol type of the associated classification. For example, if a class map represents IP classification, the **from**-type action in the table map must be either **dscp** or **precedence**. If the class map represents a non-IP classification, the **from**-type action in the table map must be **cos**.

You can configure multiple conform and exceed actions conform, exceed, and violate actions simultaneously for an aggregate policer as parameters in the **policer aggregate** global configuration command, but you must enter the actions in a particular order. See the configuration guideline in Configuring Input Policy Maps with Aggregate Policing, page 659.

After you configure the aggregate policer, you create a policy map and an associated class map, associate the policy map with the aggregate policer, and apply the service policy to a port.

Note: Only one policy map can use any specific aggregate policer. Aggregate policing cannot be used to aggregate traffic streams across multiple interfaces. It can be used only to aggregate traffic streams across multiple classes in a policy map attached to an interface and aggregate streams across VLANs on a port in a per-port, per-VLAN policy map.

After you configure the policy map and policing actions, attach the policy to an ingress port by using the **service-policy** interface configuration command.

The class maps in this example refer to access lists.

```
Switch(config)# policer aggregate agg1 cir 23000 bc 10000 conform-action set-dscp-transmit 46
exceed-action drop
Switch(config)# class-map testclass
Switch(config-cmap)# match access-group 1
Switch(config-cmap)# exit
Switch(config)# class-map videoclass
Switch(config-cmap)# match access-group 2
Switch(config-cmap)# exit
Switch(config)# policy-map testexample
Switch(config-pmap)# class testclass
Switch(config-pmap-c)# police aggregate agg1
Switch(config-pmap-c)# exit
Switch(config-pmap)# class video-class
Switch(config-pmap-c)# police aggregate agg1
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy input testexample
Switch(config-if)# exit
```

For configuration information, see Configuring Input Policy Maps with Aggregate Policing, page 659.

You can also use aggregate policing to regulate traffic streams across VLANs, as in this example:

```
Switch(config)# policer aggregate agg1 cir 23000 bc 10000 conform-action set-dscp-transmit af31
set-cos-transmit 3 exceed-action set-dscp-transmit af11 set-cos-transmit 1
Switch(config)# class-map video-provider-1
Switch(config-cmap)# match access-group 1
Switch(config-cmap)# exit
Switch(config)# class-map video-provider-2
Switch(config-cmap)# match access-group 2
Switch(config-cmap)# exit
Switch(config)# class-map match-any customer1-provider-100
Switch(config-cmap)# match vlan 100
Switch(config-cmap)# exit
Switch(config)# class-map match-any customer1-provider-200
Switch(config-cmap)# match vlan 200
Switch(config-cmap)# exit
Switch(config)# policy-map child-policy-1
Switch(config-pmap)# class video-provider-1
Switch(config-pmap-c)# set dscp af41
Switch(config-pmap-c)# exit
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# police aggregate agg1
Switch(config-pmap-c)# exit
Switch(config)# policy-map child-policy-2
Switch(config-pmap)# class video-provider-2
Switch(config-pmap-c)# set dscp cs4
Switch(config-pmap-c)# exit
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# police aggregate agg1
Switch(config-pmap-c)# exit
Switch(config)# policy-map customer-1-ingress
Switch(config-pmap)# class customer1-provider-100
Switch(config-pmap-c)# service-policy child-policy-1
Switch(config-pmap-c)# exit
Switch(config-pmap)# class customer1-provider-200
Switch(config-pmap-c)# service-policy child-policy-2
Switch(config-pmap-c)# exit
Switch(config)# interface fastethernet0/1
Switch(config-if)# switchport mode trunk
Switch(config-if)# service-policy input customer-1-ingress
Switch(config-pmap-c)# exit
```

Unconditional Priority Policing

Priority policing applies only to output policy maps. You can use the **priority** policy-map class configuration command in an output policy map to designate a low-latency path, or class-based priority queuing, for a specific traffic class. With strict priority queuing, the packets in the priority queue are scheduled and sent until the queue is empty, at the expense of other queues. Excessive use of high-priority queuing can create congestion for lower priority traffic.

To eliminate this congestion, you can use the priority with police feature (priority policing) to reduce the bandwidth used by the priority queue and allocate traffic rates on other queues. Priority with police is the only form of policing supported in output policy maps.

Note: You can configure 1-rate, 2-color policers for output policy maps with priority. You cannot configure 2-rate, 3-color policers for output policies.

See also the Configuring Output Policy Maps with Class-Based Priority Queuing, page 675.

Note: You cannot configure a policer committed burst size for an unconditional priority policer. Any configured burst size is ignored.

This example shows how to use the **priority** with **police** commands to configure *out-class1* as the priority queue, with traffic going to the queue limited to 20,000,000 bps so that the priority queue never uses more than that. Traffic above that rate is dropped. This allows other traffic queues to receive some port bandwidth, in this case a minimum bandwidth guarantee of 500,000 and 200,000 kbps. The class **class-default** queue gets the remaining port bandwidth.

```
Switch(config)# policy-map policy1
Switch(config-pmap)# class out-class1
Switch(config-pmap-c)# priority
Switch(config-pmap-c)# police 200000000
Switch(config-pmap-c)# exit
Switch(config-pmap)# class out-class2
Switch(config-pmap-c)# bandwidth 500000
Switch(config-pmap-c)# exit
Switch(config-pmap)# class out-class3
Switch(config-pmap-c)# bandwidth 200000
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# service-policy output policy1
Switch(config-if)# exit
```

Marking

You can use packet marking in input policy maps to set or modify the attributes for traffic belonging to a specific class. After network traffic is organized into classes, you use marking to identify certain traffic types for unique handling. For example, you can change the CoS value in a class or set IP DSCP or IP precedence values for a specific type of traffic. These new values are then used to determine how the traffic should be treated. You can also use marking to assign traffic to a QoS group within the switch.

Traffic marking is typically performed on a specific traffic type at the ingress port. The marking action can cause the CoS, DSCP, or precedence bits to be rewritten or left unchanged, depending on the configuration. This can increase or decrease the priority of a packet in accordance with the policy used in the QoS domain so that other QoS functions can use the marking information to judge the relative and absolute importance of the packet. The marking function can use information from the policing function or directly from the classification function.

You can specify and mark traffic by using the **set** commands in a policy map for all supported QoS markings (CoS, IP DSCP, IP precedence, and QoS groups). A **set** command unconditionally *marks* the packets that match a specific class. You then attach the policy map to an interface as an input policy map.

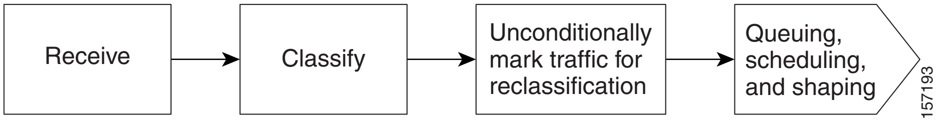
You can also mark traffic by using the **set** command with table maps.Table maps list specific traffic attributes and maps (or converts) them to another attribute. A table map establishes a to-from relationship for the attribute and defines the change to be made.

You can simultaneously configure actions to modify DSCP, precedence, and COS markings in the packet for the same service along with QoS group marking actions. You can use the QoS group number defined in the marking action for egress classification.

Note: When you use a table map in an input policy map, the protocol type of the **from**-type action in the table map must be the same as the protocol type of the associated classification. For example, if a class map represents an IP classification, the **from**-type action in the table map must be either **dscp** or **precedence**. If the class map represents a non-IP classification, the **from**-type action in the table map must be **cos**.

After you create a table map, you configure a policy map to use the table map. See Congestion Management and Scheduling, page 635. Figure 83 on page 632 shows the steps for marking traffic.

Figure 83 Marking of Classified Traffic


Receive
Classify
Unconditionally mark traffic for reclassification
Queuing, scheduling, and shaping
157193

This example uses a policy map to remark a packet. The first marking (the **set** command) applies to the QoS default class map that matches all traffic not matched by class *AF31-AF33* and sets all traffic to an IP DSCP value of 1. The second marking sets the traffic in classes AF31 to AF33 to an IP DSCP of 3.

```
Switch(config)# policy-map Example
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# set ip dscp 1
Switch(config-pmap-c)# exit
Switch(config-pmap)# class AF31-AF33
Switch(config-pmap-c)# set ip dscp 3
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy input Example
Switch(config-if)# exit
```

QoS Treatment for Performance-Monitoring Protocols

Two-Way Active Measurement Protocol

For information about the Two-Way Active Measurement Protocol (TWAMP), see Understanding TWAMP, page 41-14 and Configuring TWAMP, page 41-15.

QoS Treatment for IP-SLA and TWAMP Probes

The QoS treatment for IP-SLA and TWAMP probes must exactly reflect the effects that occur to the normal data traffic crossing the device.

The generating device should not change the probe markings. It should queue these probes based on the configured queueing policies for normal traffic.

Marking

By default, the class of service (CoS) marking of CFM traffic (including IP SLAs using CFM probes) is not changed. This feature cannot change this behavior.

By default, IP traffic marking (including IP SLA and TWAMP probes) is not changed. This feature can change this behavior.

Queuing

The CFM traffic (including IP SLAs using CFM probes) is queued according to its CoS value and the output policy map configured on the egress port, similar to normal traffic. This feature cannot change this behavior.

IP traffic (including IP SLA and TWAMP probes) is queued according to the markings specified in the **cpu traffic qos** global configuration command and the output policy map on the egress port. If this command is not configured, all IP traffic is statically mapped to a queue on the egress port.

QoS Marking for CPU-Generated Traffic

You can use QoS marking to set or modify the attributes of traffic from the CPU. The QoS marking action can cause the CoS, DSCP, or IP precedence bits in the packet to be rewritten or left unchanged. QoS uses packet markings to identify certain traffic types and how to treat them on the local switch and the network.

You can also use marking to assign traffic to a QoS group within the switch. This QoS group is an internal label that does not modify the packet, but it can be used to identify the traffic type when configuring egress queuing on the network port.

You can specify and mark traffic CPU-generated traffic by using these global configuration commands:

cpu traffic qos cos {*cos_value* | **cos** [**table-map** *table-map-name*] | **dscp** [**table-ma**p *table-map-name*] | **precedence** [**table-map** *table-map-name*]}

cpu traffic qos dscp {*dscp_value* | **cos** [**table-map** *table-map-name*] | **dscp** [**table-map** *table-map-name*] | **precedence** [**table-map** *table-map-name*]}

cpu traffic qos precedence {*precedence_value* | **cos** [**table-map** *table-map-name*] | **dscp** [**table-map** *table-map-name*] | **precedence** [**table-map** *table-map-name*]}

cpu traffic qos qos-group *value*

You can mark CoS, IP-DSCP, IP precedence, and QoS group by configuring an explicit value or by using the **table-map** keyword. Table maps list specific traffic attributes and map (or convert) them to another attribute. A table map establishes a to-from relationship for the attribute and defines the change to be made:

- Marking CoS by using the CoS, or the IP-DSCP, or the IP precedence of IP CPU-packets
- Marking CoS by using the CoS of non-IP CPU-packets.
- Marking IP DSCP by using the CoS, or the IP-DSCP, or the IP precedence of the CPU-packet
- Marking IP precedence by using the CoS, or the IP-DSCP, or the IP precedence of the CPU-packet

You can configure either IP-DSCP or IP precedence marking.

You can also simultaneously configure marking actions to modify CoS, IP-DSCP or IP precedence, and QoS group.

The **cpu traffic qos** command specifies the traffic to which it applies: all CPU traffic, only CPU IP traffic, or only CPU non-IP traffic. All other traffic retains its QoS markings. This feature does not affect CFM traffic (including Layer 2 IP SLA probes using CFM).

Note: The switch provides the ability to mark CoS, IP-DSCP and IP precedence of CPU-generated traffic by using table maps.

QoS Queuing for CPU-Generated Traffic

You can use the QoS markings established for the CPU-generated traffic by the **cpu traffic qos** global configuration command as packet identifiers in the class-map of an output policy-map to map CPU traffic to class-queues in the output policy-map on the egress port. You can then use output policy-maps on the egress port to configure queuing and scheduling for traffic leaving the switch from that port.

If you want to map *all* CPU-generated traffic to a single class in the output policy-maps without changing the CoS, IP DSCP, or IP-precedence packet markings, you can use QoS groups for marking CPU-generated traffic.

If you want to map *all* CPU-generated IP traffic to classes in the output policy maps based on IP-DSCP or IP precedence without changing those packet markings, you can use a table map:

- Configure IP-DSCP or IP precedence marking by using **DSCP** or **precedence** as the **map from** value *without* a table map.
- Configure IP-DSCP or IP-precedence marking by using **DSCP** or **precedence** as the **map from** value *with* a table map, using only the **default** and **copy** keywords.

If you want to map *all* CPU-generated traffic to classes in the output policy maps based on the CoS without changing the CoS packet markings, you can use the table map:

- Configure CoS marking by using **CoS** as the **map from** value *without* a table map.
- Configure CoS marking using **CoS** as the **map from** value *with* a table map, using only the **default** and **copy** keywords.

For details about table maps, see Table Maps, page 625.

Using the **cpu traffic qos** global configuration command with table mapping, you can configure multiple marking and queuing policies to work together or independently. You can queue native VLAN traffic based on the CoS markings configured using the **cpu traffic qos** global configuration command.

The **cpu traffic qos** command specifies the traffic to which it applies: all CPU traffic, only CPU-IP traffic, or only CPU non-IP traffic. All other traffic is statically mapped to a CPU-default queue on the egress port. All CFM traffic (including Layer 2 IP SLA probes using CFM) is mapped to classes in the output policy map and queued based on their CoS value.

Note: The switch provides the ability to queue based on the CoS, IP-DSCP, and IP precedence of CPU-generated traffic.

Configuration Guidelines

- This feature must be configured globally for a switch; it cannot be configured per-port or per-protocol.
- Enter each **cpu traffic qos** marking action on a separate line.
- The **cpu traffic qos cos** global configuration command configures CoS marking for CPU-generated traffic by using either a specific CoS value or a table map, but not both. A new configuration overwrites the existing configuration.
- The **cpu traffic qos dscp** global configuration command configures IP-DSCP marking for CPU-generated IP traffic by using either a specific DSCP value or a table map, but not both. A new configuration overwrites the existing configuration.
- The **cpu traffic qos precedence** global configuration command configures IP-precedence marking for CPU-generated IP traffic by using either a specific precedence value or a table map, but not both. A new configuration overwrites the existing configuration.
- The **cpu traffic qos dscp** and **cpu traffic qos precedence** global configuration commands are mutually exclusive. A new configuration overwrites the existing configuration.

- When the **cpu traffic qos dscp** global configuration command is configured with table maps, you can configure only one **map from** value at a time–DSCP, precedence, or CoS. A new configuration overwrites the existing configuration. Packets marked by this command can be classified and queued by an output policy map based on the marked DSCP or precedence value.
- When the **cpu traffic qos precedence** global configuration command is configured with table maps, you can configure only one **map from** value at a time–DSCP, precedence, or CoS. A new configuration overwrites the existing configuration. Packets marked by this command can be classified and queued by an output policy map based on the marked precedence or DSCP value.
- You cannot configure a **map from** value of both DSCP and precedence. A new configuration overwrites the existing configuration.
- When the **cpu traffic qos cos** global configuration command is configured with table maps, you can configure two **map from** values at a time–CoS and either DSCP or precedence.
- If the **cpu traffic qos cos** global configuration command is configured with only a **map from** value of DSCP or precedence:
  - The CoS value of IP packets is mapped by using the DSCP (or precedence) value in the packet and the configured table map. Packets can be classified and queued by an output policy map based on the marked CoS value.
  - The CoS value of non-IP packets remains unchanged.
- If the **cpu traffic qos cos** global configuration command is configured with a **map from** value of CoS:
  - The CoS value of IP packets is mapped by using the CoS value in the packet and the configured table map. Packets can be classified and queued by an output policy map based on the marked CoS value.
  - The CoS value of non-IP packets is mapped by using the CoS value in the packet and the configured table map. Packets can be classified and queued by an output policy map based on the marked CoS value.
- If the **cpu traffic qos cos** global configuration command is configured with a **map from** value of DSCP or precedence and CoS:
  - The CoS value of IP packets is mapped by using the DSCP or precedence value in the packet and the configured table map. Packets can be classified and queued by an output policy map based on the marked CoS value.
  - The CoS value of non-IP packets is mapped by using the CoS value in the packet and the configured table map. Packets can be classified and queued by an output policy map based on the marked CoS value.
- The **cpu traffic qos qos-group** global configuration command can be used to configure QoS group marking for CPU-generated traffic only for a specific QoS group. The **table-map** option is not available.

Congestion Management and Scheduling

Cisco Modular QoS CLI (MQC) provides several related mechanisms to control outgoing traffic flow. They are implemented in output policy maps to control output traffic queues. The scheduling stage holds packets until the appropriate time to send them to one of the four traffic queues. Queuing assigns a packet to a particular queue based on the packet class, and is enhanced by the WTD algorithm for congestion avoidance. You can use different scheduling mechanisms to provide a guaranteed bandwidth to a particular class of traffic while also serving other traffic in a fair way. You can limit the maximum bandwidth that can be consumed by a particular class of traffic and ensure that delay-sensitive traffic in a low-latency queue is sent before traffic in other queues.

The switch supports these scheduling mechanisms:

- Traffic shaping

You use the **shape average** policy map class configuration command to specify that a class of traffic should have a maximum permitted average rate. You specify the maximum rate in bits per second.

- Class-based-weighted-fair-queuing (CBWFQ)

  You can use the **bandwidth** policy-map class configuration command to control the bandwidth allocated to a specific class. Minimum bandwidth can be specified as a bit rate or a percentage of total bandwidth or of remaining bandwidth.

- Priority queuing or class-based priority queuing

  You use the **priority** policy-map class configuration command to specify the priority of a type of traffic over other types of traffic. You can specify strict priority for the high-priority traffic and allocate any excess bandwidth to other traffic queues, or specify priority with unconditional policing of high-priority traffic and allocate the known remaining bandwidth among the other traffic queues.

  - To configure strict priority, use only the **priority** policy-map class configuration command to configure the priority queue. Use the **bandwidth remaining percent** policy-map class configuration command for the other traffic classes to allocate the excess bandwidth in the desired ratios.

  - To configure priority with unconditional policing, configure the priority queue by using the **priority** policy-map class configuration command and the **police** policy-map class configuration command to unconditionally rate-limit the priority queue. In this case, you can configure the other traffic classes with **bandwidth** or **shape average**, depending on requirements.

These sections contain additional information about scheduling:

- Traffic Shaping, page 636
- Class-Based Weighted Fair Queuing, page 637
- Priority Queuing, page 638

Traffic Shaping

Traffic shaping is a traffic-control mechanism similar to traffic policing. While traffic policing is used in input policy maps, traffic shaping occurs as traffic leaves an interface. The switch can apply class-based shaping to classes of traffic leaving an interface and port shaping to all traffic leaving an interface. Configuring a queue for traffic shaping sets the maximum bandwidth or peak information rate (PIR) of the queue.

Note: You cannot configure traffic shaping (**shape average**) and CBWFQ (**bandwidth**) or priority queuing (**priority**) for the same class in an output policy map. You cannot configure traffic shaping for a traffic class when strict priority (priority without police) is configured for another class within the output policy-map.

Class-Based Shaping

Class-based shaping uses the **shape average** policy-map class configuration command to limit the rate of data transmission as the number of bits per second to be used for the committed information rate for a class of traffic. The switch supports separate queues for three classes of traffic. The fourth queue is always the default queue for class **class-default**, unclassified traffic.

Note: Configuring traffic shaping also automatically sets the minimum bandwidth guarantee or committed information rate (CIR) of the queue to the same value as the PIR.

Port Shaping

To configure port shaping (a transmit port shaper), create a policy map that contains only a default class, and use the **shape average** command to specify the maximum bandwidth for a port.

This example shows how to configure a policy map that shapes a port to 90 Mbps, allocated according to the *out-policy* policy map configured in the previous example. The **service-policy** policy map class command is used to create a child policy to the parent:

```
Switch(config)# policy-map out-policy-parent
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# shape average 90000000
Switch(config-pmap-c)# service-policy out-policy
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy output out-policy-parent
Switch(config-if)# exit
```

Parent-Child Hierarchy

The switch also supports *parent* policy levels and *child* policy levels for traffic shaping. The QoS parent-child structure is used for specific purposes where a child policy is referenced in a parent policy to provide additional control of a specific traffic type.

The first policy level, the parent level, is used for port shaping, and you can specific only one class of type **class-default** within the policy. This is an example of a parent-level policy map:

```
Switch(config)# policy-map parent
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# shape average 50000000
Switch(config-pmap-c)# exit
```

The second policy level, the *child* level, is used to control a specific traffic stream or class, as in this example:

```
Switch(config)# policy-map child
Switch(config-pmap)# class class1
Switch(config-pmap-c)# priority
Switch(config-pmap-c)# exit
```

Note: The total of the minimum bandwidth guarantees (CIR) for each queue of the child policy cannot exceed the total port-shape rate.

This is an example of a parent-child configuration:

```
Switch(config)# policy-map parent
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# shape average 50000000
Switch(config-pmap-c)# service-policy child
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy output parent
Switch(config-if)# exit
```

Class-Based Weighted Fair Queuing

You can configure class-based weighted fair queuing (CBWFQ) to set the relative precedence of a queue by allocating a portion of the total bandwidth that is available for the port. You use the **bandwidth** policy-map class configuration command to set the output bandwidth for a class of traffic as a rate (kilobits per second), a percentage of total bandwidth, or a percentage of remaining bandwidth.

Note: When you configure bandwidth in a policy map, you must configure all rates in the same format, either a configured rate or a percentage. The total of the minimum bandwidth guarantees (CIR) for each queue of the policy cannot exceed the total speed of the parent.

- When you use the **bandwidth** policy-map class configuration command to configure a class of traffic as an absolute rate (kilobits per second) or a percentage of total bandwidth, this represents the minimum bandwidth guarantee (CIR) for that traffic class. This means that the traffic class gets at least the bandwidth indicated by the command, but is not limited to that bandwidth. Any excess bandwidth on the port is allocated to each class in the same ratio in which the CIR rates are configured.

  You cannot configure bandwidth as an absolute rate or a percentage of total bandwidth when strict priority (priority without police) is configured for another class in the output policy.

- When you use the **bandwidth** policy-map class configuration command to configure a class of traffic as a percentage of *remaining* bandwidth, this represents the portion of the excess bandwidth of the port that is allocated to the class. This means that the class is allocated bandwidth only if there is excess bandwidth on the port, and if there is no minimum bandwidth guarantee for this traffic class.

  You can configure bandwidth as percentage of remaining bandwidth only when strict priority (priority without police) is configured for another class in the output policy map.

For more information, see Configuring Output Policy Maps with Class-Based-Weighted-Queuing, page 672.

Note: You cannot configure bandwidth and traffic shaping (**shape average**) or priority queuing (**priority**) for the same class in an output policy map.

Note: When you configure CIR bandwidth for a class as an absolute rate or percentage of the total bandwidth, any excess bandwidth remaining after servicing the CIR of all the classes in the policy map is divided among the classes in the same proportion as the CIR rates. If the CIR rate of a class is configured as 0, that class is also not eligible for any excess bandwidth and as a result receives no bandwidth.

Priority Queuing

You can use the **priority** policy-map class configuration command to ensure that a particular class of traffic is given preferential treatment. With strict priority queuing, the priority queue is constantly serviced. All packets in the queue are scheduled and sent until the queue is empty. Priority queuing allows traffic for the associated class to be sent before packets in other queues are sent.

Note: You should exercise care when using the **priority** command. Excessive use of strict priority queuing might cause congestion in other queues.

The switch supports strict priority queuing or priority used with the **police** policy-map command.

- *Strict priority queuing* (priority without police) assigns a traffic class to a low-latency queue to ensure that packets in this class have the lowest possible latency. When this is configured, the priority queue is continually serviced until it is empty, possibly at the expense of packets in other queues.

  You cannot configure priority without policing for a traffic class when traffic shaping or CBWFQ are configured for another class in the same output policy map.

- You can use priority with the **police** policy-map command, or *unconditional priority policing*, to reduce the bandwidth used by the priority queue. This is the only form of policing that is supported in output policy maps. Using this combination of commands configures a maximum rate on the priority queue, and you can use the **bandwidth** and **shape average** policy-map commands for other classes to allocate traffic rates on other queues.

  When priority is configured in an output policy map *without* the **police** command, you can only configure the other queues for sharing by using the **bandwidth remaining percent** policy-map command to allocate excess bandwidth.

Priority queuing has these restrictions:

- You can associate the **priority** command with a single unique class for all attached output polices on the switch.
- You cannot configure priority and any other scheduling action (**shape average** or **bandwidth**) in the same class.

- You cannot configure priority queuing for the **class-default** of an output policy map.

For more information, see Configuring Output Policy Maps with Class-Based Priority Queuing, page 675.

This example shows how to configure the class *out-class1* as a strict priority queue so that all packets in that class are sent before any other class of traffic. Other traffic queues are configured so that *out-class-2* gets 50 percent of the remaining bandwidth and *out-class3* gets 20 percent of the remaining bandwidth. The class **class-default** receives the remaining 30 percent with no guarantees.

```
Switch(config)# policy-map policy1
Switch(config-pmap)# class out-class1
Switch(config-pmap-c)# priority
Switch(config-pmap-c)# exit
Switch(config-pmap)# class out-class2
Switch(config-pmap-c)# bandwidth remaining percent 50
Switch(config-pmap-c)# exit
Switch(config-pmap)# class out-class3
Switch(config-pmap-c)# bandwidth remaining percent 20
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet 0/1
Switch(config-if)# service-policy output policy1
Switch(config-if)# exit
```

This example shows how to use the **priority** with **police** commands to configure *out-class1* as the priority queue, with traffic going to the queue limited to 20000000 bps so that the priority queue will never use more than that. Traffic above that rate is dropped. The other traffic queues are configured to use 50 and 20 percent of the bandwidth that is left, as in the previous example.

```
Switch(config)# policy-map policy1
Switch(config-pmap)# class out-class1
Switch(config-pmap-c)# priority
Switch(config-pmap-c)# police 200000000
Switch(config-pmap-c)# exit
Switch(config-pmap)# class out-class2
Switch(config-pmap-c)# bandwidth percent 50
Switch(config-pmap-c)# exit
Switch(config-pmap)# class out-class3
Switch(config-pmap-c)# bandwidth percent 20
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit

Switch(config)# interface gigabitethernet 0/1
Switch(config-if)# service-policy output policy1
Switch(config-if)# exit
```

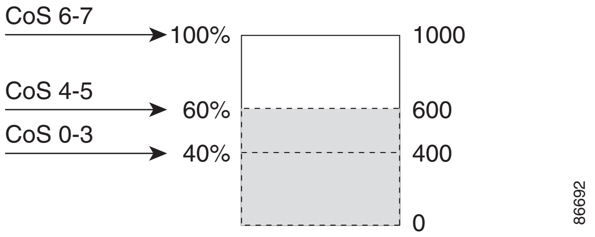
Congestion Avoidance and Queuing

Congestion avoidance uses algorithms such as tail drop to control the number of packets entering the queuing and scheduling stage to avoid congestion and network bottlenecks. The switch uses weighted tail drop (WTD) to manage the queue sizes and provide a drop precedence for traffic classifications. You set the queue size limits depending on the markings of the packets in the queue. Each packet that travels through the switch can be assigned to a specific queue and threshold. For example, specific DSCP or CoS values can be mapped to a specific egress queue and threshold.

WTD is implemented on traffic queues to manage the queue size and to provide drop precedences for different traffic classifications. As a frame enters a particular queue, WTD uses the packet classification to subject it to different thresholds. If the total destination queue size is greater than the threshold of any reclassified traffic, the next frame of that traffic is dropped.

Figure 84 on page 640 shows an example of WTD operating on a queue of 1000 frames. Three drop percentages are configured: 40 percent (400 frames), 60 percent (600 frames), and 100 percent (1000 frames). These percentages mean that traffic reclassified to the 40-percent threshold is dropped when the queue depth exceeds 400 frames, traffic reclassified to 60 percent is dropped when the queue depth exceeds 600 frames, and traffic up to 400 frames can be queued at the 40-percent threshold, up to 600 frames at the 60-percent threshold, and up to 1000 frames at the 100-percent threshold.

Figure 84 WTD and Queue Operation



In this example, CoS values 6 and 7 have a greater importance than the other CoS values, and they are assigned to the 100-percent drop threshold (queue-full state). CoS values 4 and 5 are assigned to the 60-percent threshold, and CoS values 0 to 3 are assigned to the 40-percent threshold.

If the queue is already filled with 600 frames, and a new frame arrives containing CoS values 4 and 5, the frame is subjected to the 60-percent threshold. When this frame is added to the queue, the threshold would be exceeded, so the switch drops it.

WTD is configured by using the **queue-limit** policy-map class command. The command adjusts the queue size (buffer size) associated with a particular class of traffic. You specify the threshold as the number of packets, where each packet is a fixed unit of 256 bytes. You can specify different queue sizes for different classes of traffic (CoS, DSCP, precedence, or QoS group) in the same queue. Setting a queue limit establishes a drop threshold for the associated traffic when congestion occurs.

Note: You cannot configure queue size by using the **queue-limit** policy map class command without first configuring a scheduling action (**bandwidth**, **shape average**, or **priority**). The only exception to this is when you configure queue-limit for the **class-default** of an output policy map.

The switch supports up to three unique queue-limit configurations across all output policy maps. Within an output policy map, only four queues (classes) are allowed, including the class default. Each queue has three thresholds defined. Only three unique threshold value configurations are allowed on the switch. However, multiple policy maps can share the same queue-limits. When two policy maps a share queue-limit configuration, all threshold values must be the same for all the classes in both policy maps.

For more information, see Configuring Output Policy Maps with Class-Based-Weighted-Queuing, page 672.

This example configures *class A* to match DSCP values and a policy map, *PM1*. The DSCP values of 30 and 50 are mapped to unique thresholds (32 and 64, respectively). The DSCP values of 40 and 60 are mapped to the maximum threshold of 112 packets.

```
Switch(config)# class-map match-any classA
Switch(config-cmap)# match ip dscp 30 40 50 60
Switch(config-cmap)# exit
Switch(config)# policy-map PM1
Switch(config-pmap)# class classA
Switch(config-pmap-c)# bandwidth percent 50
Switch(config-pmap-c)# queue-limit 112
Switch(config-pmap-c)# queue-limit dscp 30 32
Switch(config-pmap-c)# queue-limit dscp 50 64
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet 0/1
Switch(config-if)# service-policy output PM1
Switch(config-if)# exit
```

You can use these same queue-limit values in multiple output policy maps on the switch. However, changing one of the queue-limit values in a class creates a new, unique queue-limit configuration. You can attach only three unique queue-limit configurations in output policy maps to interfaces at any one time. If you attempt to attach an output policy map with a fourth unique queue-limit configuration, you see this error message:

```
QoS: Configuration failed. Maximum number of allowable unique queue-limit configurations exceeded.
```

Note: When you configure a queue limit for a class in an output policy map, all other output policy maps must use the same qualifier type and qualifier value format. Only the queue-limit threshold values can be different. For example, when you configure *class A* queue limit thresholds for **dscp 30** and **dscp 50** in policy map *PM1*, and you configure *class A* queue limits in policy map *PM2*, you must use **dscp 30** and **dscp 50** as qualifiers. You cannot use **dscp 20** and **dscp 40**. The threshold values can be different, but different threshold values creates a new queue-limit configuration.

By default, the total amount of buffer space is divided equally among all ports and all queues per port, which is adequate for many applications. You can decrease the queue size for latency-sensitive traffic or increase the queue size for bursty traffic.

Note: When you use the **queue-limit** command to configure queue thresholds for a class, the WTD thresholds must be less than or equal to the queue maximum threshold. A queue size configured with no qualifier must be larger than any queue sizes configured with qualifiers.

When you configure queue limit, the range for the number of packets is from 16 to 544, in multiples of 16, where each packet is a fixed unit of 256 bytes.

Note: For optimal performance, we strongly recommend that you configure the queue-limit to 272 or less.

Queue bandwidth and queue size (queue limit) are configured separately and are not interdependent. You should consider the type of traffic being sent when you configure bandwidth and queue-limit:

- A large buffer (queue limit) can better accommodate bursty traffic without packet loss, but at the cost of increased latency.
- A small buffer reduces latency but is more appropriate for steady traffic flows than for bursty traffic.
- Very small buffers are typically used to optimize priority queuing. For traffic that is priority queued, the buffer size usually needs to accommodate only a few packets; large buffer sizes that increase latency are not usually necessary. For high-priority latency-sensitive packets, configure a relatively large bandwidth and relatively small queue size.

These restrictions apply to WTD qualifiers:

- You cannot configure more than two threshold values for WTD qualifiers (**cos**, **dscp**, **precedence**, **qos-group**) by using the **queue-limit** command. However, there is no limit to the number of qualifiers that you can map to these thresholds. You can configure a third threshold value to set the maximum queue by using the **queue-limit** command with no qualifiers.
- A WTD qualifier in the **queue-limit** command must be the same as at least one **match** qualifier in the associated class map.

This example shows how to configure bandwidth and queue limit so that *out-class1*, *out-class2*, *out-class3*, and **class-default** get a minimum of 40, 20, 10 and 10 percent of the traffic bandwidth, respectively. The corresponding queue-sizes are set to 48, 32, 16 and 272 (256-byte) packets:

```
Switch(config)# policy-map out-policy
Switch(config-pmap)# class outclass1
Switch(config-pmap-c)# bandwidth percent 40
Switch(config-pmap-c)# queue-limit 48
Switch(config-pmap-c)# exit
Switch(config-pmap)# class outclass2
Switch(config-pmap-c)# bandwidth percent 20
Switch(config-pmap-c)# queue-limit 32
Switch(config-pmap-c)# exit
Switch(config-pmap)# class outclass3
Switch(config-pmap-c)# bandwidth percent 10
Switch(config-pmap-c)# queue-limit 16
Switch(config-pmap-c)# exit
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# bandwidth percent 10
Switch(config-pmap-c)# queue-limit 272
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet 0/1
Switch(config-if)# service-policy output out-policy
Switch(config-if)# exit
```

You can configure and attach as many output policy maps as there are switch ports, but only three unique queue-limit configurations are allowed. When another output policy map uses the same queue-limit and class configurations, even if the bandwidth percentages are different, it is considered to be the same queue-limit configuration.

Configuring QoS

Before configuring QoS, you must have a thorough understanding of these factors:

- The types of applications used and the traffic patterns on your network.
- Traffic characteristics and needs of your network. Is the traffic bursty? Do you need to reserve bandwidth for voice and video streams?
- Bandwidth requirements and speed of the network.
- Location of congestion points in the network.

These sections describe how to classify, police, and mark incoming traffic, and schedule and queue outgoing traffic. Depending on your network configuration, you must perform one or more of these tasks.

- Default QoS Configuration, page 642
- QoS Configuration Guidelines, page 643
- Using ACLs to Classify Traffic, page 644
- Using Class Maps to Define a Traffic Class, page 648
- Configuring Table Maps, page 650
- Attaching a Traffic Policy to an Interface, page 651
- Configuring Input Policy Maps, page 652
- Configuring Output Policy Maps, page 670
- Configuring QoS Marking and Queuing for CPU-Generated Traffic, page 682

Default QoS Configuration

There are no policy maps, class maps, table maps, or policers configured. At the egress port, all traffic goes through a single default queue that is given the full operational port bandwidth. The default size of the default queue is 160 (256-byte) packets.

The packets are not modified (the CoS, DSCP, and IP precedence values in the packet are not changed). Traffic is switched in pass-through mode without any rewrites and classified as best effort without any policing.

QoS Configuration Guidelines

- You can configure QoS only on physical ports.
- On a port configured for QoS, all traffic received through the port is classified, policed, and marked according to the input policy map attached to the port. On a trunk port configured for QoS, traffic in all VLANs received through the port is classified, policed, and marked according to the policy map attached to the port. If a per-port, per-VLAN policy map is attached, traffic on the trunk port is classified, policed, and marked for the VLANs specified in the parent-level policy, according to the child policy map associated with each VLAN.
- If you have EtherChannel ports configured on your switch, you must configure QoS classification, policing, mapping, and queuing on the individual physical ports that comprise the EtherChannel. You must decide whether the QoS configuration should match on all ports in the EtherChannel.
- Control traffic (such as spanning-tree bridge protocol data units [BPDUs] and routing update packets) received by the switch are subject to all ingress QoS processing.
- You are likely to lose data when you change queue settings; therefore, try to make changes when traffic is at a minimum.
- When you try to attach a new policy to an interface and this brings the number of policer *instances* to more than 1024 minus 1 more than the number of interfaces on the switch255, you receive an error message, and the configuration fails.
- When you try to attach new policy to an interface, increasing the number of policer *profiles* to more than 256, you receive an error message, and the configuration fails. A profile is a combination of commit rate, peak rate, commit burst, and peak burst. You can attach one profile to multiple instances, but if one of these characteristics differs, the policer is considered to have a new profile.
- You can specify 256 *unique* VLAN classification criteria within a per-port, per-VLAN policy-map, across all ports on the switch. Any policy attachment or change that causes this limit to be exceeded fails with a *VLAN label resources exceeded* error message.
- You can attach per-port and per-port, per-VLAN policy-maps across all ports on the switch until QoS ACE classification resource limitations are reached. Any policy attachment or change that causes this limit to be exceeded fails with a *TCAM resources exceeded* error message.
- When CPU protection is enabled, you can configure only 45 policers per port. Disabling CPU protection allows you to configure up to 64 policers per port. You can enter the **show policer cpu uni-eni** {**drop** | **rate**} privileged EXEC command to see if CPU protection is enabled.
- Note these limitations when you disable CPU protection:
  - When CPU protection is disabled, you can configure a maximum of 63 policers per port (62 on every 4th port) for user-defined classes, and one for class-default. Any policy attachment or change that causes this limit to be exceeded fails with a *policer resources exceeded* error message.
  - When CPU protection is disabled, you can configure a maximum of 256 policers on the switch. Any policy attachment or change that causes this limit to be exceeded fails with a *policer resources exceeded* error message.
  - If you disable CPU protection and attach a policy map with more than 45 policers, and then enable CPU protection again, and reload, 19 policers per port are again required for CPU protection. During reload, the policers 46 and above will reach the *policer resources exceeded* error condition and no policers are attached to those classes.

- If the number of internal QoS labels exceeds 256, you receive an error message.
- Table maps are not supported for **violate-action** for aggregate policing unless you configure a table map for exceed-action and no explicit action is configured for violate-action. For both individual and aggregate policers, if you do not configure a **violate-action**, by default the violate class is assigned the same action as the **exceed-action**.
- If double-tagged packets are received on a trunk or 802.1Q tunnel interface, these packets can be classified on DSCP and IP precedence along with other parameters, but you cannot set DSCP or IP precedence on the outgoing packets. You can set CoS on the outgoing packets.

See the configuration sections for specific QoS features for more configuration guidelines related to each feature.

Using ACLs to Classify Traffic

You can classify IP traffic by using IP standard or IP extended ACLs. You can classify IP and non-IP traffic by using Layer 2 MAC ACLs.

Follow these guidelines when configuring QoS ACLs:

- You cannot match IP fragments against configured IP extended ACLs to enforce QoS. IP fragments are sent as best-effort. IP fragments are denoted by fields in the IP header.
- The switch supports only one access group per class in an input policy map.
- You cannot configure **match-access** group in an output policy map.

These sections describe how to create QoS ACLs:

- Creating IP Standard ACLs, page 644
- Creating IP Extended ACLs, page 645
- Creating Layer 2 MAC ACLs, page 647

Creating IP Standard ACLs

Beginning in privileged EXEC mode, follow these steps to create an IP standard ACL for IP traffic:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**access-list** *access-list-number* **permit** *source* [*source-wildcard*]	Create an IP standard ACL, repeating the command as many times as necessary. ■ For *access-list-number*, enter the access list number. The range is 1 to 99 and 1300 to 1999. ■ Always use the **permit** keyword for ACLs used as match criteria in QoS policies. QoS policies do not match ACLs that use the **deny** keyword. ■ For *source*, enter the network or host from which the packet is being sent. You can use the **any** keyword as an abbreviation for 0.0.0.0 255.255.255.255. ■ (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source.
or	**ip access-list standard** *name*	Define a standard IPv4 access list using a name, and enter access-list configuration mode. The name can be a number from 1 to 99. In access-list configuration mode, enter **permit** *source* [*source-wildcard*]
3.	**end**	Return to privileged EXEC mode.
4.	**show access-lists**	Verify your entries.
5.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete an access list, use the **no access-list** *access-list-number* global configuration command.

This example shows how to allow access for only those hosts on the three specified networks. The wildcard bits apply to the host portions of the network addresses.

```
Switch(config)# access-list 1 permit 192.5.255.0 0.0.0.255
Switch(config)# access-list 1 permit 128.88.0.0 0.0.255.255
Switch(config)# access-list 1 permit 36.0.0.0 0.0.0.255
```

Creating IP Extended ACLs

Beginning in privileged EXEC mode, follow these steps to create an IP extended ACL for IP traffic:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**access-list** *access-list-number* **permit** *protocol* {*source source-wildcard destination destination-wildcard*} [**precedence** *precedence*] [**tos** *tos*] [**dscp** *dscp*] **Note:** If you enter a **dscp** value, you cannot enter **tos** or **precedence**. You can enter both a **tos** and a **precedence** value with no **dscp**.	Create an IP extended ACL. Repeat the step as many times as necessary. ■ For *access-list-number*, enter the access list number. The range is 100 to 199 and 2000 to 2699. ■ Always use the **permit** keyword for ACLs used as match criteria in QoS policies. QoS policies do not match **deny** ACLs. ■ For *protocol*, enter the name or number of an IP protocol. Use the question mark (?) to see a list of available protocols. To match any Internet protocol (including ICMP, TCP, and UDP), enter **ip**. ■ The *source* is the number of the network or host sending the packet. ■ The *source-wildcard* applies wildcard bits to the source. ■ The *destination* is the network or host number receiving the packet. ■ The *destination-wildcard* applies wildcard bits to the destination. You can specify source, destination, and wildcards as: ■ The 32-bit quantity in dotted-decimal format. ■ The keyword **any** for 0.0.0.0 255.255.255.255 (any host). ■ The keyword **host** for a single host 0.0.0.0. Other keywords are optional and have these meanings: ■ **precedence**—Enter to match packets with a precedence level specified as a number from 0 to 7 or by name: **routine** (**0**), **priority** (**1**), **immediate** (**2**), **flash** (**3**), **flash-override** (**4**), **critical** (**5**), **internet** (**6**), **network** (**7**). ■ **tos**—Enter to match by type of service level, specified by a number from 0 to 15 or a name: **normal** (**0**), **max-reliability** (**2**), **max-throughput** (**4**), **min-delay** (**8**). ■ **dscp**—Enter to match packets with the DSCP value specified by a number from 0 to 63, or use the question mark (?) to see a list of available values.
or	**ip access-list extended** *name*	Define an extended IPv4 access list using a name, and enter access-list configuration mode. The *name* can be a number from 100 to 199. In access-list configuration mode, enter **permit** *protocol* {*source source-wildcard destination destination-wildcard*} [**precedence** *precedence*] [**tos** *tos*] [**dscp** *dscp*] as defined in Step 2.
3.	**end**	Return to privileged EXEC mode.
4.	**show access-lists**	Verify your entries.
5.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete an access list, use the **no access-list** *access-list-number* global configuration command.

This example shows how to create an ACL that permits IP traffic from any source to any destination that has the DSCP value set to 32:

```
Switch(config)# access-list 100 permit ip any any dscp 32
```

This example shows how to create an ACL that permits IP traffic from a source host at 10.1.1.1 to a destination host at 10.1.1.2 with a precedence value of 5:

```
Switch(config)# access-list 100 permit ip host 10.1.1.1 host 10.1.1.2 precedence 5
```

Creating Layer 2 MAC ACLs

Beginning in privileged EXEC mode, follow these steps to create a Layer 2 MAC ACL for non-IP traffic:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**mac access-list extended** *name*	Create a Layer 2 MAC ACL by specifying the name of the list and enter extended MAC ACL configuration mode.
3.	**permit** {**host** *src-MAC-addr mask* | **any** | **host** *dst-MAC-addr* | *dst-MAC-addr mask*} [*type mask*]	Always use the **permit** keyword for ACLs used as match criteria in QoS policies. ■ For *src-MAC-addr*, enter the MAC address of the host from which the packet is being sent. You can specify in hexadecimal format (H.H.H), use the **any** keyword for *source* 0.0.0, *source-wildcard* ffff.ffff.ffff, or use the **host** keyword for *source* 0.0.0. ■ For *mask,* enter the wildcard bits by placing ones in the bit positions that you want to ignore. ■ For *dst-MAC-addr*, enter the MAC address of the host to which the packet is being sent. You can specify in hexadecimal format (H.H.H), use the **any** keyword for *source* 0.0.0, *source-wildcard* ffff.ffff.ffff, or use the **host** keyword for *source* 0.0.0. ■ (Optional) For *type mask*, specify the Ethertype number of a packet with Ethernet II or SNAP encapsulation to identify the protocol of the packet. For *type*, the range is from 0 to 65535, typically specified in hexadecimal. For *mask*, enter the *don't care* bits applied to the Ethertype before testing for a match.
4.	**end**	Return to privileged EXEC mode.
5.	**show access-lists** [*access-list-number* | *access-list-name*]	Verify your entries.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete an access list, use the **no mac access-list extended** *access-list-name* global configuration command.

This example shows how to create a Layer 2 MAC ACL with two **permit** statements. The first statement allows traffic from the host with MAC address 0001.0000.0001 to the host with MAC address 0002.0000.0001. The second statement allows only Ethertype XNS-IDP traffic from the host with MAC address 0001.0000.0002 to the host with MAC address 0002.0000.0002.

```
Switch(config)# mac access-list extended maclist1
Switch(config-ext-macl)# permit 0001.0000.0001 0.0.0 0002.0000.0001 0.0.0
Switch(config-ext-macl)# permit 0001.0000.0002 0.0.0 0002.0000.0002 0.0.0 xns-idp
Switch(config-ext-macl)# exit
```

Using Class Maps to Define a Traffic Class

You use the **class-map** global configuration command to name and to isolate a specific traffic flow (or class) from all other traffic. A class map defines the criteria to use to match against a specific traffic flow to further classify it. Match statements can include criteria such as an ACL, CoS value, DSCP value, IP precedence values, QoS group values, or VLAN IDs. You define match criterion with one or more **match** statements entered in the class-map configuration mode.

Follow these guidelines when configuring class maps:

- A **match-all** class map cannot have more than one classification criterion (one match statement), but a **match-any** class map can contain multiple match statements.
- The **match cos** and **match vlan** commands are supported only on Layer 2 802.1Q trunk ports.
- You use a class map with the **match vlan** command in the parent policy in input hierarchical policy maps for per-port, per-VLAN QoS on trunk ports. A policy is considered a parent policy map when it has one or more of its classes associated with a child policy map. Each class within a parent policy map is called a parent class. You can configure only the **match vlan** command in parent classes. You cannot configure the **match vlan** command in classes within the child policy map.
- For an input policy map, you cannot configure an IP classification (**match ip dscp**, **match ip precedence**, **match access-group** for an IP ACL) and a non-IP classification (**match cos** or **match access-group** for a MAC ACL) in the same policy map or class map. For a per-port, per-VLAN hierarchical policy map, this applies to the child policy map.
- You cannot configure **match qos-group** for an input policy map.
- In an output policy map, no two class maps can have the same classification criteria; that is, the same match qualifiers and values.
- The maximum number of class maps on the switch is 1024.

Beginning in privileged EXEC mode, follow these steps to create a class map and to define the match criterion to classify traffic:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**class-map** [**match-all** | **match-any**] *class-map-name*	Create a class map, and enter class-map configuration mode. By default, no class maps are defined. ■ (Optional) Use the **match-all** keyword to perform a logical-AND of all matching statements under this class map. All match criteria in the class map must be matched. ■ (Optional) Use the **match-any** keyword to perform a logical-OR of all matching statements under this class map. One or more match criteria must be matched. ■ For *class-map-name*, specify the name of the class map. If no matching statements are specified, the default is **match-all**. **Note:** A **match-all** class map cannot have more than one classification criterion (match statement).
3.	**match** {**access-group** *acl-index-or-name* | **cos** *cos-list* | **ip dscp** *dscp-list* | **ip precedence** *ip-precedence-list* | **qos-group** *value* | **vlan** *vlan-list*}	Define the match criterion to classify traffic. By default, no match criterion is defined. Only one match type per class map is supported, and only one ACL per class map is supported. ■ For **access-group** *acl-index-or-name,* specify the number or name of an ACL. Matching access groups is supported only in input policy maps. ■ For **cos** *cos-list*, enter a list of up to four CoS values in a single line to match against incoming packets. Separate each value with a space. You can enter multiple *cos-list* lines to match more than four CoS values. The range is 0 to 7. ■ For **ip dscp** *dscp-list*, enter a list of up to eight IPv4 DSCP values to match against incoming packets. Separate each value with a space. You can enter multiple *dscp-list* lines to match more than eight DSCP values. The numerical range is 0 to 63. You can also configure DSCP values in other forms. See Classification Based on IP DSCP, page 620. ■ For **ip precedence** *ip-precedence-list*, enter a list of up to four IPv4 precedence values to match against incoming packets. Separate each value with a space. You can enter multiple *ip-precedence-list* lines to match more than four precedence values. The range is 0 to 7. ■ For **vlan** *vlan-list,* specify a VLAN ID or a range of VLANs to be used in a parent policy map for per-port, per-VLAN QoS on a trunk port. The VLAN ID range is 1 to 4094. ■ For **qos-group** *value,* specify the QoS group number. The range is 0 to 99. Matching of QoS groups is supported only in output policy maps.
4.	**end**	Return to privileged EXEC mode.
5.	**show class-map**	Verify your entries.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** form of the appropriate command to delete an existing class map or remove a match criterion.

This example shows how to create access list 103 and configure the class map called *class1*. The *class1* has one match criterion, which is access list 103. It permits traffic from any host to any destination that matches a DSCP value of 10.

```
Switch(config)# access-list 103 permit any any dscp 10
Switch(config)# class-map class1
Switch(config-cmap)# match access-group 103
Switch(config-cmap)# exit
```

This example shows how to create a class map called *class2*, which matches incoming traffic with DSCP values of 10, 11, and 12.

```
Switch(config)# class-map match-any class2
Switch(config-cmap)# match ip dscp 10 11 12
Switch(config-cmap)# exit
```

This example shows how to create a class map called *class3*, which matches incoming traffic with IP-precedence values of 5, 6, and 7:

```
Switch(config)# class-map match-any class3
Switch(config-cmap)# match ip precedence 5 6 7
Switch(config-cmap)# exit
```

This example shows how to create a parent class-map called *parent-class*, which matches incoming traffic with VLAN IDs in the range from 30 to 40.

```
Switch(config)# class-map match-any parent-class
Switch(config-cmap)# match vlan 30-40
Switch(config-cmap)# exit
```

Configuring Table Maps

You can configure table maps to manage a large number of traffic flows with a single command. You use table maps to correlate specific DSCP, IP precedence and CoS values to each other, to mark down a DSCP, IP precedence, or CoS value, or to assign default values. You can specify table maps in **set** commands and use them as mark-down mapping for the policers.

These table maps are supported on the switch:

- DSCP to CoS, precedence, or DSCP
- CoS to DSCP, precedence, or CoS
- Precedence to CoS, DSCP, or precedence

Note these guidelines when configuring table maps:

- The switch supports a maximum of 256 unique table maps.
- The maximum number of map statements within a table map is 64.
- Table maps cannot be used in output policy maps.
- Table maps are not supported for **violate-action** for aggregate policing unless you configure a table map for exceed-action and no explicit action is configured for violate-action.

Beginning in privileged EXEC mode, follow these steps to create a table map:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**table-map** *table-map-name*	Create a table map by entering a table-map name and entering table-map configuration mode.
3.	**map from** *from-value* **to** *to-value*	Enter the mapping values to be included in the table. For example, if the table map is a DSCP-to-CoS table map, the *from-value* would be the DSCP value and the *to_value* would be the CoS value. Both ranges are from 0 to 63. Enter this command multiple times to include all the values that you want to map.
4.	**default** {*default-value* | **copy** | **ignore**}	Set the default behavior for a value not found in the table map. ■ Enter a *default-value* to specify a certain value. For example, in a DSCP-to-CoS table map, this would be a specific CoS value to apply to all unmapped DSCP values. The range is from 0 to 63. ■ Enter **copy** to map unmapped values to an equivalent value. In a DSCP-to-CoS table map, this command maps all unmapped DSCP values to the equivalent CoS value. ■ Enter **ignore** to leave unmapped values unchanged. In a DSCP-to-CoS table map, the switch does not change the CoS value of unmapped DSCP values.
5.	**end**	Return to privileged EXEC mode.
6.	**show table-map** [*table-map-name*]	Verify your entries.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete a table map, use the **no table-map** *table-map-name* global configuration command.

This example shows how to create a DSCP-to-CoS table map. A complete table would typically include additional map statements for the higher DSCP values. The default of 4 in this table means that unmapped DSCP values will be assigned a CoS value of 4.

```
Switch(config)# table-map dscp-to-cos
Switch(config-tablemap)# map from 1 to 1
Switch(config-tablemap)# map from 2 to 1
Switch(config-tablemap)# map from 3 to 1
Switch(config-tablemap)# map from 4 to 2
Switch(config-tablemap)# map from 5 to 2
Switch(config-tablemap)# map from 6 to 3
Switch(config-tablemap)# default 4
Switch(config-tablemap)# end
Switch# show table-map dscp-to-cos
```

Attaching a Traffic Policy to an Interface

You use the **service-policy** interface configuration command to attach a traffic policy to an interface and to specify the direction in which the policy should be applied: either an input policy map for incoming traffic or an output policy map for outgoing traffic. Input and output policy maps support different QoS features. See Configuring Input Policy Maps, page 652 and the Configuring Output Policy Maps, page 670 for restrictions on input and output policy maps.

You can attach a service policy only to a physical port. You can attach only one input policy map and one output policy map per port.

Beginning in privileged EXEC mode, follow these steps to attach a policy map to a port:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Specify the port to attach to the policy map, and enter interface configuration mode. Valid interfaces are physical ports.
3.	**service-policy** {**input** | **output**} *policy-map-name*	Specify the policy-map name and whether it is an input policy map or an output policy map.
4.	**end**	Return to privileged EXEC mode.
5.	**show policy-map interface** [*interface-id*]	Verify your entries.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the policy map and port association, use the **no service-policy** {**input** | **output**} *policy-map-name* interface configuration command.

Configuring Input Policy Maps

Policy maps specify which traffic class to act on and what actions to take. All traffic that fails to meet matching criteria of a traffic class belongs to the default class. Input policy maps regulate traffic entering the switch. In an input policy, you can match CoS, DSCP, IP precedence, ACLs, or VLAN IDs and configure individual policing, aggregate policing, or marking to a CoS, DSCP, IP precedence, or QoS group value.

Follow these guidelines when configuring input policy maps:

- You can attach only one input policy map per port.

- The maximum number of policy maps configured on the switch is 256.

- The total number of configurable policer profiles on the switch is 256; the total number of supported policer instances on the switch is 1024 minus one more than the total number of interfaces on the switch. On a 24-port switch, the number of available policer instances is 999. You can use a policer profile in multiple instances.

- The maximum number of classes in each input policy map is 64 plus **class-default**.

- The number of input policy maps that can be attached in a switch is limited by the availability of hardware resources. If you attempt to attach an input policy map that causes any hardware resource limitation to be exceeded, the configuration fails.

- After you have attached a single-level policy map to an interface by using the **service-policy input** interface configuration command, you can modify the policy without detaching it from the interface. You can add or delete classification criteria, add or delete classes, add or delete actions, or change the parameters of the configured actions (policers, rates, mapping, marking, and so on). This also applies to changing criteria for the child policy of a hierarchical policy map, as in a per-port per-VLAN hierarchical policy map.

  For the parent policy of a hierarchical policy map, you cannot add or delete a class at the parent level if the policy map is attached to an interface. You must detach the policy from the interface, modify the policy, and then re-attach it to the interface.

- You can configure a maximum 2-level hierarchical policy map as an input policy map only with VLAN-based classification at the parent level and no VLAN-based classification at the child level.

- When an input policy map with only Layer 2 classification is attached to a routed port or a switch port containing a routed SVI, the service policy acts only on switching eligible traffic and not on routing eligible traffic.

- On an 802.1Q tunnel port, you can use only an input policy map with Layer 2 classification based on MAC ACLs to classify traffic. Input policy maps with Layer 3 classification or with Layer 2 classification based on CoS or VLAN ID are not supported on tunnel ports.
- Input policy maps support policing and marking, not scheduling or queuing. You cannot configure **bandwidth**, **priority**, **queue-limit**, or **shape average** in input policy maps.

These sections describe how to configure different types of input policy maps:

- Configuring Input Policy Maps with Individual Policing, page 653
- Configuring Input Policy Maps with Aggregate Policing, page 659
- Configuring Input Policy Maps with Marking, page 664
- Configuring Per-Port Per-VLAN QoS with Hierarchical Input Policy Maps, page 665

Configuring Input Policy Maps with Individual Policing

You use the **police** policy-map class configuration command to configure individual policers to define the committed rate limitations, committed burst size limitations of the traffic, and the action to take for a class of traffic.

Follow these guidelines when configuring individual policers:

- Policing is supported only on input policy maps.
- The switch supports a maximum of 229 policers. (228 user-configurable policers and 1 policer reserved for internal use).
- When CPU protection is enabled (the default), you can configure 45 ingress policers per port. If you disable CPU protection by entering the **no policer cpu uni all** global configuration command and reloading the switch, you can configure a maximum of 63 policers per port (62 on every 4th port) for user-defined classes and one for class-default. You can enter the **show policer cpu uni-eni** {**drop** | **rate**} privileged EXEC command to see if CPU protection is enabled.
- When you use a table map for police exceed-action in an input policy map, the protocol type of the *map from* type of action must be the same as the protocol type of the associated classification. For example, if the associated class map represents an IP classification, the **map from** type of action that references the table map must be **dscp** or **precedence**. If the associated class map represents a non-IP classification, the **map from** type of action that references the table map must be **cos**.
- 2-rate, 3-color policing is supported only on input policy maps; 1-rate, 2-color policing is supported on both input and output policy maps.
- The number of policer instances on the switch can be 1024 minus 1 more than the number interfaces. The switch supports a maximum of 256 policer profiles.
- If you do not configure a violate-action, by default the violate class is assigned the same action as the exceed-action.

Beginning in privileged EXEC mode, follow these steps to create an input policy map with individual policing:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**policy-map** *policy-map-name*	Create a policy map by entering the policy map name, and enter policy-map configuration mode. By default, no class maps are defined.
3.	**class** {*class-map-name* | **class-default**}	Enter a class-map name or **class-default** to match all unclassified packets, and enter policy-map class configuration mode. If you enter a class-map name, you must have already created the class map by using the **class-map** global configuration command.
4.	**police** {*rate-bps* / **cir** *cir-bps*} [*burst-bytes* / **bc** *burst-bytes*]	Define a policer for the class of traffic. By default, no policer is defined. ■ For *rate-bps,* specify average traffic rate in bits per second (bps). The range is 8000 to 1000000000. ■ For **cir** *cir-bps,* specify a committed information rate (CIR) in bits per second (bps). The range is 8000 to 1000000000. ■ For *burst-bytes* (optional)*,* specify the normal burst size in bytes. The range is 8000 to 1000000. ■ For **bc** *burst-bytes* (optional)*,* specify the conformed burst (bc) or the number of acceptable burst bytes. The range is 8000 to 1000000.
5.	**conform-action cos** {*cos_value* / **cos** [**table** *table-map-name*] | **dscp** [**table** *table-map-name*] | **precedence** [**table** *table-map-name*]} or **conform-action** [**ip**] **dscp** {*dscp_value* | **cos** [**table** *table-map-name*] | **dscp** [**table** *table-map-name*] | **precedence** [**table** *table-map-name*]} or **conform-action** [**ip**] **precedence** {*precedence_value* | **cos** [**table** *table-map-name*] | **dscp** [**table** *table-map-name*] | **precedence** [**table** *table-map-name*]} or **conform-action qos-group** *value* or **transmit**	(Optional) Enter the action to be taken on packets that conform to the CIR. ■ For **cos** *cos_value,* enter a new CoS value to be assigned to the classified traffic. The range is 0 to 7. ■ For [**ip**] **dscp** *dscp_value*, enter a new DSCP value to be assigned to the classified traffic. The range is 0 to 63. ■ For [**ip**] **precedence** *precedence_value,* enter a new IP-precedence value to be assigned to the classified traffic. The range is 0 to 7. ■ Or you can configure a CoS, DSCP, or IP precedence table and optionally enter the table name. If you do not enter **table** *table-map name*, the table map default behavior is **copy**. See Configuring Table Maps, page 650. ■ For **qos-group** *value,* identify a QoS group to be used at egress to identify specific packets. The range is from 0 to 99. **Note:** You can enter a single conform-action as part of the command string following the police command. You can also press Enter after the **police** command to enter policy-map class police configuration mode, where you can enter multiple actions. In policy-map class police configuration mode, you must enter an action to take.

	Command	Purpose
6.	**exceed-action cos** {*cos_value* / **cos** [**table** *table-map-name*] | **dscp** [**table** *table-map-name*] | **precedence** [**table** *table-map-name*]} or **exceed-action** [**ip**] **dscp** {*dscp_value* | **cos** [**table** *table-map-name*] | **dscp** [**table** *table-map-name*] | **precedence** [**table** *table-map-name*]} or **exceed-action** [**ip**] **precedence** {*precedence_value* | **cos** [**table** *table-map-name*] | **dscp** [**table** *table-map-name*] | **precedence** [**table** *table-map-name*]} or **exceed-action qos-group** *value*	(Optional) Enter the action to be taken for packets that do not conform to the CIR. - For **cos** *cos_value,* enter a new CoS value to be assigned to the classified traffic. The range is 0 to 7. - For [**ip**] **dscp** *dscp_value*, enter a new DSCP value to be assigned to the classified traffic. The range is 0 to 63. - For [**ip**] **precedence** *precedence_value,* enter a new IP-precedence value to be assigned to the classified traffic. The range is 0 to 7. - Or you can configure a CoS, DSCP, or IP precedence table and optionally enter the table name. If you do not enter **table** *table-map name*, the table map default behavior is **copy**. See Configuring Table Maps, page 650. - For **qos-group** *value,* identify a QoS group to be used at egress to identify specific packets. The range is from 0 to 99. **Note:** You can enter a single exceed-action as part of the command string following the **police** command. Or you can press Enter after the **police** command to enter policy-map class police configuration mode, where you can enter multiple actions. In policy-map class police configuration mode, you must enter an action to take. **Note:** If you explicitly configure **exceed-action drop** as keywords in the command, you must enter policy-map class police configuration mode and enter the **no exceed-action drop** command to remove the previously configured exceed action before you can enter the new exceed-action.
7.	**exit**	Return to policy-map configuration mode.
8.	**exit**	Return to global configuration mode.
9.	**interface** *interface-id*	Enter interface configuration mode for the interface to which you want to attach the policy.
10.	**service-policy input** *policy-map-name*	Attach the policy map (created in Step 2) to the ingress interface.
11.	**end**	Return to privileged EXEC mode.
12.	**show policy-map** [*policy-map-name* [**class** *class-map-name*]]	Verify your entries.
13.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Beginning in privileged EXEC mode, follow these steps to create an input policy map with individual 2-rate, 3-color policing:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**policy-map** *policy-map-name*	Create a policy map by entering the policy map name, and enter policy-map configuration mode. By default, no class maps are defined.
3.	**class** {*class-map-name* | **class-default**}	Enter a class-map name or **class-default** to match all unclassified packets, and enter policy-map class configuration mode. If you enter a class-map name, you must have already created the class map by using the **class-map** global configuration command.
4.	**police** {*rate-bps* | **cir** {*cir-bps*} [*burst-bytes*] [**bc** [*conform-burst*] [**pir** *pir-bps* [**be** *peak-burst*]]	Define a policer using one or two rates–committed information rate (CIR) and peak information rate (PIR) for the class of traffic. By default, no policer is defined. ■ For *rate-bps,* specify average traffic rate in bits per second (bps). The range is 8000 to 1000000000. ■ For **cir** *cir-bps,* specify a committed information rate at which the bc token bucket is updated in bits per second (b/s). The range is 8000 to 1000000000. ■ For *burst-bytes* (optional)*,* specify the normal burst size in bytes. The range is 8000 to 1000000. ■ (Optional) For **bc** *conform-burst,* specify the conformed burst used by the bc token bucket for policing.The range is 8000 to 1000000 bytes. ■ (Optional) For **pir** *pir-bps,* specify the peak information rate at which the be token bucket for policing is updated. The range is 8000 to 1000000000 b/s. If you do not enter a **pir** *pir-bps*, the policer is configured as a 1-rate, 2-color policer. ■ For **be** *peak-burst*, specify the peak burst size used by the be token bucket. The range is 8000 to 1000000 bytes. The default is internally calculated based on the user configuration.

	Command	Purpose
5.	**conform-action** [**drop | set-cos-transmit** {*cos_value* | [**cos** | **dscp** | **precedence**] [**table** *table-map name*]} | **set-dscp-transmit** {*dscp_value* | [**cos | dscp** | **precedence**] [**table** *table-map name*]} | **set-prec-transmit** {*precedence_value* | [**cos** | **dscp** | **precedence**] [**table** *table-map name*]} | **set-qos-transmit** *qos-group_value* | **transmit**] | **exceed-action** [**drop** | **set-cos-transmit** {*cos_value* | [**cos** | **dscp** | **precedence**] [**table** *table-map name*]} | **set-dscp-transmit** {*dscp_value* | [**cos** | **dscp** | **precedence**] [**table** *table-map name*]} | **set-prec-transmit** {*precedence_value* | [**cos** | **dscp** | **precedence**] [**table** *table-map name*]} | **set-qos-transmit** *qos-group_value* | **transmit**] | **violate- action** [**drop | set-cos-transmit** {*cos_value* | [**cos** | **dscp** | **precedence**] [**table** *table-map name*]} | **set-dscp-transmit** {*dscp_value* | [**cos | dscp** | **precedence**] [**table** *table-map name*]} | **set-prec-transmit** {*precedence_value* | [**cos** | **dscp** | **precedence**] [**table** *table-map name*]} | **set-qos-transmit** *qos-group_value* | **transmit**]	(Optional) Enter the action to be taken on packets, depending on whether or not they conform to the CIR and PIR. ■ (Optional) For **conform-action**, specify the action to perform on packets that conform to the CIR and PIR. The default is **transmit**. ■ (Optional) For **exceed-action**, specify the action to perform on packets that conform to the PIR but not the CIR. The default is **drop**. ■ (Optional) For **violate-action**, specify the action to perform on packets that exceed the PIR. The default is **drop**. ■ (Optional) For *action*, specify one of these actions to perform on the packets: – **drop**–Drop the packet. **Note:** If the conform action is set to **drop**, the exceed and violate actions are automatically set to **drop**. If the exceed action is set to **drop**, the violate action is automatically set to **drop**. – **set-cos-transmit** *cos-value*–Enter a new CoS value to be assigned to the packet, and send the packet. The range is from 0 to 7. – **set-dscp-transmit** *dscp-value*–Enter a new IP DSCP value to be assigned to the packet, and send the packet. The range is from 0 to 63. You can also enter a mnemonic name for a commonly used value. – **set-prec-transmit** *cos-value*–Enter a new IP precedence value to be assigned to the packet, and send the packet. The range is from 0 to 7. – **set-qos-transmit** *qos-group-value*–Identify a qos-group to be used at egress to specify packets. The range is from 0 to 99. – **transmit**–Send the packet without altering it. **Note:** You can enter a single conform-action as part of the command string following the police command. You can also press Enter after the **police** command to enter policy-map class police configuration mode, where you can enter multiple actions. In policy-map class police configuration mode, you must enter an action to take.
6.	**exit**	Return to policy-map configuration mode.
7.	**exit**	Return to global configuration mode.
8.	**interface** *interface-id*	Enter interface configuration mode for the interface to which you want to attach the policy.
9.	**service-policy input** *policy-map-name*	Attach the policy map (created in Step 2) to the ingress interface.

	Command	Purpose
10.	**end**	Return to privileged EXEC mode.
11.	**show policy-map** [*policy-map-name*| **interface**]	Verify your entries.
12.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

After you have created an input policy map, you attach it to an interface in the input direction. See Attaching a Traffic Policy to an Interface, page 651.

Use the **no** form of the appropriate command to delete an existing policy map, class map, or policer.

This example shows how to configure 2-rate, 3-color policing using policy-map configuration mode.

```
Switch(config)# class-map cos-4
Switch(config-cmap)# match cos 4
Switch(config-cmap)# exit
Switch(config)# policy-map in-policy
Switch(config-pmap)# class cos-4
Switch(config-pmap-c)# police cir 5000000 pir 8000000 conform-action transmit exceed-action
set-dscp-transmit 24 violate-action drop
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy input in-policy
Switch(config-if)# exit
```

This example shows how to create the same configuration using policy-map class police configuration mode.

```
Switch(config)# class-map cos-4
Switch(config-cmap)# match cos 4
Switch(config-cmap)# exit
Switch(config)# policy-map in-policy
Switch(config-pmap)# class cos-4
Switch(config-pmap-c)# police cir 5000000 pir 8000000
Switch(config-pmap-c-police)# conform-action transmit
Switch(config-pmap-c-police)# exceed-action set-dscp-transmit 24
Switch(config-pmap-c-police)# violate-action drop
Switch(config-pmap-c-police)# end
```

This example shows how to create a traffic classification with a CoS value of 4, create a policy map, and attach it to an ingress port. The average traffic rate is limited to 10000000 b/s with a burst size of 10000 bytes:

```
Switch(config)# class-map video-class
Switch(config-cmap)# match cos 4
Switch(config-cmap)# exit
Switch(config)# policy-map video-policy
Switch(config-pmap)# class video-class
Switch(config-pmap-c)# police 10000000 10000
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy input video-policy
Switch(config-if)# exit
```

This example shows how to create policy map with a conform action of **set dscp** and a default exceed action.

```
Switch(config)# class-map in-class-1
Switch(config-cmap)# match dscp 14
Switch(config-cmap)# exit
Switch(config)# policy-map in-policy
Switch(config-pmap)# class in-class-1
Switch(config-pmap-c)# police 230000 8000 conform-action set-dscp-transmit 33 exceed-action drop
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy input in-policy
Switch(config-if)# exit
```

This example shows how to use policy-map class police configuration mode to set multiple conform actions and an exceed action. The policy map sets a committed information rate of 23000 bits per second (bps) and a conform burst size of 10000 bytes. The policy map includes multiple conform actions (for DSCP and for Layer 2 CoS) and an exceed action.

```
Switch(config)# class-map cos-set-1
Switch(config-cmap)# match cos 3
Switch(config-cmap)# exit
Switch(config)# policy-map map1
Switch(config-pmap)# class cos-set-1
Switch(config-pmap-c)# police cir 23000 bc 10000
Switch(config-pmap-c-police)# conform-action set-dscp-transmit 48
Switch(config-pmap-c-police)# conform-action set-cos-transmit 5
Switch(config-pmap-c-police)# exceed-action drop
Switch(config-pmap-c-police)# exit
Switch(config-pmap)# exit
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy input map1
Switch(config-if)# exit
```

This example shows how to use policy-map class police configuration mode to set exceed action mark-down using table-maps. The policy map sets a committed information rate of 23000 bps and a conform burst-size of 10000 bytes. The policy map includes the default conform action (**transmit**) and the exceed action to mark the Layer 2 CoS value based on the table map and to mark IP DSCP to af41.

```
Switch(config)# policy-map in-policy
Switch(config-pmap)# class in-class-1
Switch(config-pmap-c)# police cir 23000 bc 10000
Switch(config-pmap-c-police)# exceed-action set-cos-transmit cos table police-cos-markdn-tablemap
Switch(config-pmap-c-police)# exceed-action set-dscp-transmit af41
Switch(config-pmap-c-police)# exit
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy input in-policy
Switch(config-if)# exit
```

Configuring Input Policy Maps with Aggregate Policing

You use the **policer aggregate** global configuration command to configure an aggregate policer. An aggregate policer is shared by multiple traffic classes within the same policy map. You define the aggregate policer, create a policy map, associate a class map with the policy map, associate the policy map with the aggregate policer, and apply the service policy to a port.

Follow these guidelines when configuring aggregate policers:

- Aggregate policing is supported only on input policy maps.
- The switch supports a maximum of 229 policers associated with ports (228 user-configurable policers and 1 policer reserved for internal use). You can configure up to 45 policers on a port.

- When CPU protection is enabled (the default), you can configure 45 ingress policers per port. If you disable CPU protection by entering the **no policer cpu uni all** global configuration command and reloading the switch, you can configure a maximum of 63 policers per port (62 on every 4th port) for user-defined classes and one for class-default. You can enter the **show policer cpu uni-eni** {**drop** | **rate**} privileged EXEC command to see if CPU protection is enabled.
- The maximum number of configured aggregate policers is 256.
- The number of policer instances on the switch can be 1024 minus 1 more than the total number interfaces on the switch. The switch supports a maximum of 256 policer profiles.
- If you do not configure a violate-action, by default the violate class is assigned the same action as the exceed-action.
- Only one policy map can use any specific aggregate policer. Aggregate policing cannot be used to aggregate streams across multiple interfaces. You can use aggregate policing only to aggregate streams across multiple classes in a policy map attached to an interface and to aggregate traffic streams across VLANs on a port in a per-port, per-VLAN policy map.
- When you use a table map for police exceed-action in an input policy map, the protocol type of the map from type of action must be the same as the protocol type of the associated classification. For example, if the associated class map represents an IP classification, the **map from** type of action that references the table map must be either **dscp** or **precedence**. If the associated class map represents a non-IP classification, the **map from** type of action that references the table map must be **cos**.
- Table maps are not supported for **violate-action** for aggregate policing unless a table map is configured for **exceed-action** and no explicit action is configured for violate-action.

You can configure multiple conform and exceed actions conform, exceed, and violate actions simultaneously for an aggregate policer as parameters in the **policer aggregate** global configuration command, but you must enter the actions in this order:

- **conform-action** must be followed by **drop** or **transmit** or by **set** actions in this order:

  set-qos-transmit

  set-dscp-transmit or **set-prec-transmit**

  set-cos-transmit

- **exceed-action** must be followed by **drop** or **transmit** or by **set** actions in this order:

  set-qos-transmit

  set-dscp-transmit or **set-prec-transmit**

  set-cos-transmit

- **violate-action** must be followed by **drop** or **transmit** or by **set** actions in this order:

  set-qos-transmit

  set-dscp-transmit or **set-prec-transmit**

  set-cos-transmit

Note: You do not configure aggregate policer conform-action, exceed-action, and violate-action in policy-map class police configuration mode; you must enter all actions in a string. Consequently, if you enter multiple conform, exceed, and violate actions, the command can become quite long, in which case it might be truncated and difficult to read.

Beginning in privileged EXEC mode, follow these steps to create an aggregate policer:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**policer aggregate** *aggregate-policer-name* {*rate-bps* / **cir** *cir-bps*} [**bc** *burst- value*] [**conform-action** [**set-cos-transmit** {*cos_value* | [**cos** | **dscp** | **precedence**] [**table** *table-map name*]} | **set-dscp-transmit** {*dscp_value* | [**cos | dscp** | **precedence**] [**table** *table-map name*]} | **set-prec-transmit** {*precedence_value* | [**cos** | **dscp** | **precedence**] [**table** *table-map name*]} | **set-qos-transmit** *qos-group_value* | **transmit**] [**exceed action** [**drop |** **set-cos-transmit** {*cos_value* | [**cos** | **dscp** | **precedence**] [**table** *table-map name*]} | **set-dscp-transmit** {*dscp_value* | [**cos | dscp** | **precedence**] [**table** *table-map name*]} | **set-prec-transmit** {*precedence_value* | [**cos** | **dscp** | **precedence**] [**table** *table-map name*]} | **set-qos-transmit** *qos-group_value* | **transmit**]]	Define the policer parameters that can be applied to multiple traffic classes within the same policy map. ■ For *aggregate-policer-name*, specify the name of the aggregate policer. ■ For *rate-bps,* specify average traffic rate in bits per second (bps). The range is 8000 to 1000000000. ■ For **cir** *cir-bps,* specify the committed information rate in bits per second. The range is 8000 to 1000000000 bps. ■ (Optional) For **bc** *burst-value*, specify conform burst and the number of acceptable burst bytes. The range is 8000 to 1000000 bytes. ■ (Optional) For **conform-action**, specify the action to take on packets that conform to the CIR. The default is to send the packet. ■ (Optional) For **exceed-action**, specify the action to take on packets that exceed the CIR. The default is to drop the packet. See the command reference for this release or Configuring Input Policy Maps with Individual Policing, page 653 for definitions of the available keywords.
3.	**policy-map** *policy-map-name*	Create a policy map by entering the policy map name, and enter policy-map configuration mode.
4.	**class** {*class-map-name* | **class-default**}	Enter a class-map name or **class-default** to match all unclassified packets, and enter policy-map class configuration mode. If you enter a class-map name, you must have already created the class map by using the **class-map** global configuration command.
5.	**police aggregate** *aggregate-policer-name*	Apply an aggregate policer to multiple classes in the same policy map. For *aggregate-policer-name*, enter the name specified in Step 2.
6.	**exit**	Return to policy-map configuration mode.
7.	**exit**	Return to global configuration mode.
8.	**interface** *interface-id*	Enter interface configuration mode for the interface to which you want to attach the policy.
9.	**service-policy input** *policy-map-name*	Attach the policy map (created in Step 3) to the ingress interface.
10.	**end**	Return to privileged EXEC mode.
11.	**end**	Return to privileged EXEC mode.
12.	**show policer aggregate** [*aggregate-policer-name*]	Verify your entries.
13.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Beginning in privileged EXEC mode, follow these steps to create a 2-rate, 3-color aggregate policer:

<table>
<tr><th></th><th>Command</th><th>Purpose</th></tr>
<tr><td>1.</td><td>**configure terminal**</td><td>Enter global configuration mode.</td></tr>
<tr><td>2.</td><td>**policer aggregate** *aggregate-policer-name* {*rate-bps* | **cir** *cir-bps*} [*burst-bytes*] [**bc** [*conform-burst*] [**pir** *pir-bps* [**be** *peak-burst*]] [**conform-action** [**drop | set-cos-transmit** {*cos_value* | [**cos** | **dscp** | **precedence**] [**table** *table-map name*]} | **set-dscp-transmit** {*dscp_value* | [**cos | dscp** | **precedence**] [**table** *table-map name*]} | **set-prec-transmit** {*precedence_value* | [**cos** | **dscp** | **precedence**] [**table** *table-map name*]} | **set-qos-transmit** *qos-group_value* | **transmit**]
[**exceed-action** [**drop** | **set-cos-transmit** {*cos_value* | [**cos** | **dscp** | **precedence**] [**table** *table-map name*]} | **set-dscp-transmit** {*dscp_value* | [**cos** | **dscp** | **precedence**] [**table** *table-map name*]} | **set-prec-transmit** {*precedence_value* | [**cos** | **dscp** | **precedence**] [**table** *table-map name*]} | **set-qos-transmit** *qos-group_value* | **transmit**]]
[**violate- action** [**drop | set-cos-transmit** {*cos_value* | [**cos** | **dscp** | **precedence**]} | **set-dscp-transmit** {*dscp_value* | [**cos | dscp** | **precedence**]} | **set-prec-transmit** {*precedence_value* | [**cos** | **dscp** | **precedence**]} | **set-qos-transmit** *qos-group_value* | **transmit**]]</td><td>Define the policer parameters that can be applied to multiple traffic classes within the same policy map.
■ For *aggregate-policer-name*, specify the name of the aggregate policer.
■ For *rate-bps,* specify average traffic rate in bits per second (b/s). The range is 8000 to 1000000000.
■ For **cir** *cir-bps,* specify a committed information rate (CIR) at which the first token bucket is updated in bits per second (b/s). The range is 8000 to 1000000000.
■ For *burst-bytes* (optional), specify the normal burst size in bytes. The range is 8000 to 1000000.
■ (Optional) For **bc** *conform-burst,* specify the conformed burst used by the first token bucket for policing.The range is 8000 to 1000000 bytes.
■ (Optional) For **pir** *pir-bps,* specify the peak information rate at which the second token bucket for policing is updated. The range is 8000 to 1000000000 bits per second. If you do not enter a **pir** *pir-bps*, the policer is configured as a 1-rate, 2-color policer.
■ For **be** *peak-burst*, specify the peak burst size used by the second token bucket. The range is 8000 to 1000000 bytes. The default is internally calculated based on the user configuration.
■ (Optional) For **conform-action**, specify the action to take on packets that conform to the CIR. The default is to send the packet.
Note: If the conform action is set to **drop**, the exceed and violate actions are automatically set to **drop**. If the exceed action is set to **drop**, the violate action is automatically set to **drop**.
■ (Optional) For **exceed-action**, specify the action to take on packets that exceed the CIR. The default is to drop the packet.
■ (Optional) For **violate-action**, specify the action to take on packets that exceed the CIR. The default is to drop the packet.
see Configuring Input Policy Maps with Individual Policing, page 653 for definitions of the action keywords.
Note: You cannot configure table maps for **violate-action** for aggregate policing unless a table map is configured for **exceed-action** and no explicit action is configured for violate-action.</td></tr>
<tr><td>3.</td><td>**policy-map** *policy-map-name*</td><td>Create a policy map by entering the policy map name, and enter policy-map configuration mode.</td></tr>
</table>

	Command	Purpose
4.	**class** {*class-map-name* | **class-default**}	Enter a class-map name or **class-default** to match all unclassified packets, and enter policy-map class configuration mode. If you enter a class-map name, you must have already created the class map by using the **class-map** global configuration command.
5.	**police aggregate** *aggregate-policer-name*	Apply an aggregate policer to multiple classes in the same policy map. For *aggregate-policer-name*, enter the name specified in Step 2.
6.	**exit**	Return to policy-map configuration mode.
7.	**exit**	Return to global configuration mode.
8.	**interface** *interface-id*	Enter interface configuration mode for the interface to which you want to attach the policy.
9.	**service-policy input** *policy-map-name*	Attach the policy map (created in Step 3) to the ingress interface.
10.	**end**	Return to privileged EXEC mode.
11.	**show policer aggregate** [*aggregate-policer-name*]	Verify your entries.
12.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

After you have created an aggregate policer, you attach it to an ingress port. See Attaching a Traffic Policy to an Interface, page 651.

To remove the specified aggregate policer from a policy map, use the **no police aggregate** *aggregate-policer-name* policy map configuration mode. To delete an aggregate policer and its parameters, use the **no policer aggregate** *aggregate-policer-name* global configuration command.

This example shows how to create an aggregate policer and attach it to multiple classes within a policy map. The policy map is attached to an ingress port.

```
Switch(config)# policer aggregate example 10900000 80000 conform-action transmit exceed-action drop
Switch(config)# class-map testclass1
Switch(config-cmap)# match access-group 1
Switch(config-cmap)# exit
Switch(config)# class-map testclass2
Switch(config-cmap)# match access-group 2
Switch(config-cmap)# exit
Switch(config)# policy-map testexample
Switch(config-pmap)# class testclass
Switch(config-pmap-c)# police aggregate example
Switch(config-pmap-c)# exit
Switch(config-pmap)# class testclass2
Switch(config-pmap-c)# police aggregate example
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy input testexample
Switch(config-if)# exit
```

This example shows how to create a 2-rate, 3-color aggregate policer and attach it to multiple classes within a policy map. The policy map is attached to an ingress port.

```
Switch(config)# policer aggregate example cir 10900000 pir 80000000 conform-action transmit
exceed-action drop violate-action drop
Switch(config)# class-map testclass1
Switch(config-cmap)# match access-group 1
Switch(config-cmap)# exit
Switch(config)# class-map testclass2
Switch(config-cmap)# match access-group 2
Switch(config-cmap)# exit
Switch(config)# policy-map testexample
Switch(config-pmap)# class testclass
Switch(config-pmap-c)# police aggregate example
Switch(config-pmap-c)# exit
Switch(config-pmap)# class testclass2
Switch(config-pmap-c)# police aggregate example
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy input testexample
Switch(config-if)# exit
```

Configuring Input Policy Maps with Marking

You use the **set** policy-map class configuration command to set or modify the attributes for traffic belonging to a specific class. Follow these guidelines when configuring marking in policy maps:

- You can configure a maximum of 100 QoS groups on the switch.
- When you use a table map for marking in an input policy map, the protocol type of the map from type of action must be the same as the protocol type of the associated classification. For example, if the associated class map represents an IP classification, the **map from** type of action that references the table map must be either **dscp** or **precedence**. If the associated class map represents a non-IP classification, the **map from** type of action that references the table map must be **cos**.

Beginning in privileged EXEC mode, follow these steps to create an input policy map that marks traffic:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**policy-map** *policy-map-name*	Create a policy map by entering the policy map name, and enter policy-map configuration mode.
3.	**class** {*class-map-name* | **class-default**}	Enter a class-map name, or **class-default** to match all unclassified packets, and enter policy-map class configuration mode. If you enter a class-map name, you must have already created the class map by using the **class-map** global configuration command.
4.	**set qos-group** *value* and/or **set cos** {*cos_value* / **cos** [**table** *table-map-name*] | **dscp** [**table** *table-map-name*] | **precedence** [**table** *table-map-name*]} and/or **set** [**ip**] **dscp** {*dscp_value* | **cos** [**table** *table-map-name*] | **dscp** [**table** *table-map-name*] | **precedence** [**table** *table-map-name*]} and/or **set** [**ip**] **precedence** {*precedence_value* | **cos** [**table** *table-map-name*] | **dscp** [**table** *table-map-name*] | **precedence** [**table** *table-map-name*]}	Mark traffic by setting a new value in the packet, specifying a table map, or specifying a QoS group. ■ For **qos-group** *value,* identify a QoS group to be used at egress to identify specific packets. The range is from 0 to 99. ■ For **cos** *cos_value,* enter a new CoS value to be assigned to the classified traffic. The range is 0 to 7. ■ For [**ip**] **dscp** *new-dscp*, enter a new DSCP value to be assigned to the classified traffic. The range is 0 to 63. ■ For [**ip**] **precedence** *new-precedence,* enter a new IP-precedence value to be assigned to the classified traffic. The range is 0 to 7. ■ You can also configure a CoS, DSCP, or IP precedence table and optionally enter the table name. If you do not enter **table** *table-map name*, the table map default behavior is **copy**. See Configuring Table Maps, page 650.

	Command	Purpose
5.	**exit**	Return to policy-map configuration mode.
6.	**exit**	Return to global configuration mode.
7.	**interface** *interface-id*	Enter interface configuration mode for the interface to which you want to attach the policy.
8.	**service-policy input** *policy-map-name*	Attach the policy map (created in Step 2) to the ingress interface.
9.	**end**	Return to privileged EXEC mode.
10.	**show policy-map** [*policy-map-name* [**class** *class-map-name*]]	Verify your entries.
11.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** form of the appropriate command to delete a policy map or table map or remove an assigned CoS, DSCP, precedence, or QoS-group value.

This example uses a policy map to remark a packet. The first marking (the **set** command) applies to the QoS default class map that matches all traffic not matched by class *AF31-AF33* and sets all traffic to an IP DSCP value of 1. The second marking sets the traffic in classes AF31 to AF33 to an IP DSCP of 3.

```
Switch(config)# policy-map Example
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# set ip dscp 1
Switch(config-pmap-c)# exit
Switch(config-pmap)# class AF31-AF33
Switch(config-pmap-c)# set ip dscp 3
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy input Example
Switch(config-if)# exit
```

Configuring Per-Port Per-VLAN QoS with Hierarchical Input Policy Maps

Per-port, per-VLAN QoS allows classification based on VLAN IDs for applying QoS for frames received on a given interface and VLAN. This is achieved by using a hierarchical policy map, with a parent policy and a child policy.

Note these guidelines and limitations when configuring per-port, per-VLAN QoS:

- The feature is supported only by using a two-level hierarchical input policy map, where the parent level defines the VLAN-based classification, and the child level defines the QoS policy to be applied to the corresponding VLAN or VLANs.

- You can configure multiple service classes at the parent level to match different combinations of VLANs, and you can apply independent QoS policies to each parent-service class using any child policy map

- A policy is considered a parent policy map when it has one or more of its classes associated with a child policy map. Each class within a parent policy-map is called a parent-class. In parent classes, you can configure only the **match vlan** class-map configuration command. You cannot configure the **match vlan** command in classes within the child policy map.

- A per-port, per-VLAN parent level class map supports only a child-policy association; it does not allow any actions to be configured. For a parent-level class map, you cannot configure an action or a child-policy association for the class **class-default**.

- You cannot configure a mixture of Layer 2 and Layer 3 class maps in a child policy map. When you attempt to associate such a child policy map with a parent policy, the configuration is rejected. However, you can associate Layer 2 child policies and Layer 3 child policies with different parent-level class maps.
- Per-port, per-VLAN QoS is supported only on 802.1Q trunk ports.
- When the child policy-map attached to a VLAN or set of VLANs contains only Layer 3 classification (**match ip dscp**, **match ip precedence**, **match IP ACLs**), take care to ensure that these VLANs are not carried on any other port besides the one on which the per-port, per-vlan policy is attached. Not following this rule could result in improper QoS behavior for traffic ingressing the switch on these VLANs.
- We also recommend that you restrict VLAN membership on the trunk ports to which the per-port, per-VLAN is applied by using the **switchport trunk allowed vlan** interface configuration command. Overlapping VLAN membership between trunk ports that have per-port, per-VLAN policies with Layer 3 classification could also result in unexpected QoS behavior.

Configuring per-port, per-VLAN QoS includes these tasks:

- Creating Child-Policy Class Maps, page 666
- Creating Parent-Policy Class Maps, page 668
- Creating Child Policy Maps, page 668
- Creating a Parent Policy Map, page 669
- Attaching a Parent Policy Map to an Interface, page 669

Creating Child-Policy Class Maps

Beginning in privileged EXEC mode, follow these steps to create one or more child-policy class maps:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**class-map** [**match-all** | **match-any**] *child-class-map-name*	Create a class map, and enter class-map configuration mode. By default, no class maps are defined. ■ (Optional) Use the **match-all** keyword to perform a logical-AND of all matching statements under this class map. All match criteria in the class map must be matched. ■ (Optional) Use the **match-any** keyword to perform a logical-OR of all matching statements under this class map. One or more match criteria must be matched. ■ For *class-map-name*, specify the name of the class map. If no matching statements are specified, the default is **match-all**. **Note:** A **match-all** class map cannot have more than one classification criterion (match statement).
3.	**match** {**access-group** *acl-index-or-name* | **cos** *cos-list* | **ip dscp** *dscp-list* | **ip precedence** *ip-precedence-list* | **qos-group** *value* | **vlan** *vlan-list*}	Define the match criterion to classify traffic. By default, no match criterion is defined. Only one match type per class map is supported, and only one ACL per class map is supported. ■ For **access-group** *acl-index-or-name,* specify the number or name of an ACL. Matching access groups is supported only in input policy maps. ■ For **cos** *cos-list*, enter a list of up to four CoS values in a single line to match against incoming packets. Separate each value with a space. You can enter multiple *cos-list* lines to match more than four CoS values. The range is 0 to 7. ■ For **ip dscp** *dscp-list*, enter a list of up to eight IPv4 DSCP values to match against incoming packets. Separate each value with a space. You can enter multiple *dscp-list* lines to match more than eight DSCP values. The numerical range is 0 to 63. You can also configure DSCP values in other forms. See Classification Based on IP DSCP, page 620. ■ For **ip precedence** *ip-precedence-list*, enter a list of up to four IPv4 precedence values to match against incoming packets. Separate each value with a space. You can enter multiple *ip-precedence-list* lines to match more than four precedence values. The range is 0 to 7. ■ For **qos-group** *value,* specify the QoS group number. The range is 0 to99. Matching of QoS groups is supported only in output policy maps. ■ For **vlan** *vlan-list,* specify a VLAN ID or a range of VLANs to be used in a parent policy map for per-port, per-VLAN QoS on a trunk port. The VLAN ID range is 1 to 4094.

	Command	Purpose
4.	**end**	Return to privileged EXEC mode.
5.	**show class-map**	Verify your entries.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Creating Parent-Policy Class Maps

Beginning in privileged EXEC mode, follow these steps to create one or more parent-policy class maps:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**class-map match-any** *parent-class-map-name*	Create a **match-any** class map for the parent policy, and enter class-map configuration mode. **Note:** You can enter **match-all** or not enter either **match-any** or **match-all** (to default to **match-all**) if you are going to match only one VLAN ID.
3.	**match vlan** *vlan-id*	Define the VLAN or VLANs on which to classify traffic. For *vlan-id,* specify a VLAN ID, a series of VLAN IDs separated by a space, or a range of VLANs separated by a hyphen to be used in a parent policy map for per-port, per-VLAN QoS on a trunk port. The VLAN ID range is 1 to 4094. You can also enter the **match vlan** command multiple times to match multiple VLANs.
4.	**end**	Return to privileged EXEC mode.
5.	**show class-map**	Verify your entries.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Creating Child Policy Maps

Beginning in privileged EXEC mode, follow these steps to create one or more child policy maps:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**policy-map** *child-policy-map-name*	Create a child policy map by entering the policy map name, and enter policy-map configuration mode.
3.	**class** {*child-class-map-name* | **class-default**}	Enter a child class-map name or **class-default** to match all unclassified packets, and enter policy-map class configuration mode.
4.	Use the **police** policy-map class configuration command to configure policers and the action to take for a class of traffic, or use the **set** policy-map class configuration command to mark traffic belonging to the class.	
5.	**end**	Return to privileged EXEC mode.
6.	**show policy-map** [*child-policy-map-name* [**class** *class-map-name*]]	Verify your entries.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Creating a Parent Policy Map

Beginning in privileged EXEC mode, follow these steps to create a parent policy map and attach it to an interface:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**policy-map** *parent-policy-map-name*	Create a parent policy map by entering the policy map name, and enter policy-map configuration mode.
3.	**class** *parent-class-map-name*	Enter the parent class-map name, and enter policy-map class configuration mode.
4.	**service policy** *child-policy-map-name*	Associate the child policy map with the parent policy map
5.	**end**	Return to privileged EXEC mode.
6.	**show policy-map** [*parent-policy-map-name* [**class** *class-map-name*]]	Verify your entries.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Attaching a Parent Policy Map to an Interface

Beginning in privileged EXEC mode, follow these steps to create attach the parent policy map to an interface:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Enter interface configuration mode for the interface to which you want to attach the policy.
3.	**switchport mode trunk**	Configure the port as a trunk port.
4.	**switchport trunk allowed vlan** *vlan-list*	(Recommended) Restrict VLAN membership for trunk ports to avoid overlapping VLAN membership if the per-port, per-VLAN policy includes Layer 3 classification.
5.	**service-policy input** *parent-policy-map-name*	Attach the parent policy map (created in the previous section) to the ingress interface.
6.	**end**	Return to privileged EXEC mode.
7.	**show policy-map interface** [*interface-id*]	Verify your entries.
8.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

This is an example of using multiple parent classes to classify matching criteria for voice and video on customer VLANs.

```
Switch(config)# class-map match-any video
Switch(config-cmap)# match ip dscp af41
Switch(config-cmap)# exit
Switch(config)# class-map match-any voice
Switch(config-cmap)# match ip dscp ef
Switch(config-cmap)# exit

Switch(config)# class-map match-any customer1-vlan
Switch(config-cmap)# match vlan 100-105
Switch(config-cmap)# exit
Switch(config)# class-map match-any customer2-vlan
Switch(config-cmap)# match vlan 110-120
Switch(config-cmap)# exit

Switch(config)# policy-map child-policy-1
Switch(config-pmap)# class voice
Switch(config-pmap-c)# police cir 10000000 bc 50000
Switch(config-pmap-c-police)# conform-action set-cos-transmit 5
Switch(config-pmap-c-police)# exceed-action drop
Switch(config-pmap-c-police)# exit
Switch(config-pmap-c)# exit
Switch(config-pmap)# class video
Switch(config-pmap-c)# set cos 4
Switch(config-pmap-c)# exit
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# set cos 0
Switch(config-pmap-c)# exit

Switch(config)# policy-map child-policy-2
Switch(config-pmap)# class voice
Switch(config-pmap-c)# police cir 5000000
Switch(config-pmap-c-police)# conform-action set-cos-transmit 5
Switch(config-pmap-c-police)# exceed-action drop
Switch(config-pmap-c-police)# exit
Switch(config-pmap-c)# exit
Switch(config-pmap)# class video
Switch(config-pmap-c)# police cir 40000000
Switch(config-pmap-c-police)# conform-action set-cos-transmit 4
Switch(config-pmap-c-police)# exceed-action set-cos-transmit 1
Switch(config-pmap-c-police)# exit
Switch(config-pmap-c)# exit
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# set cos 0
Switch(config-pmap-c)# exit

Switch(config)# policy-map uni-parent
Switch(config-pmap)# class customer1-vlan
Switch(config-pmap-c)# service-policy child-policy-1
Switch(config-pmap-c)# exit
Switch(config-pmap)# class customer2-vlan
Switch(config-pmap-c)# service-policy child-policy-2
Switch(config-pmap-c)# exit

Switch(config)# interface fastethernet0/1
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan 100-105, 110-120
Switch(config-if)# service-policy input uni-parent
Switch(config-pmap-c)# exit
```

Configuring Output Policy Maps

You use output policy maps to manage congestion avoidance, queuing, and scheduling of packets leaving the switch. The switch has four egress queues, and you use output policy maps to control the queue traffic. You configure shaping, queue-limit, and bandwidth on these queues. You can use high priority (class-based priority queuing). Policing is not supported on output policy maps, except when configuring priority with police for class-based priority queuing. Output policy map classification criteria are matching a CoS, DSCP, or IP precedence value or a QoS group.

Follow these guidelines when configuring output policy maps on physical ports:

- You can configure and attach as many output policy maps as there are ports on the switch. Multiple output policy maps can use the same queue-limit configuration. However, these policy maps can have only three unique queue-limit configurations.
- Output policy maps can have a maximum of four classes, including the class **class-default**.

- All output policy maps must have the same number of defined class-maps defined, either 1, 2, or 3.
- All output policy maps must use the same set of classes, although the actions for each class can differ for each output policy map.
- In a child policy map, the **class-default** supports all output policy map actions except **priority** and **police**. Action restrictions for **class-default** are the same as for other classes except that a queue limit configuration for **class-default** does not require a scheduling action.
- To classify based on criteria at the output, the criteria must be established at the input. You can establish criteria at the input through classification only when you configure only policing and not marking, or through explicit marking when you configure any marking (policing with **conform** or **exceed** marking or unconditional **set** marking).
- You cannot configure class-based priority queuing under the class **class-default** in an output policy map
- In an output policy map, unless priority queuing is configured, the class default receives a minimum bandwidth guarantee equal to the unconfigured bandwidth on the port.
- After you have attached an output policy map to an interface by using the **service-policy** interface configuration command, you can change only the parameters of the configured actions (rates, percentages, and so on) or add or delete classification criteria of the class map while the policy map is attached to the interface. To add or delete a class or action, you must detach the policy map from all interfaces, modify it, and then reattach it to interfaces.

  If you anticipate that you might need three classes in a policy map, you should define three classes when you create the policy map, even if you are not ready to use all three at that time. You cannot add a class to a policy map after it has been attached to an interface.
- When at least one output policy map is attached to a active port, other active ports without output policy maps attached might incorrectly schedule and incorrectly order traffic that uses the same classes as the attached output policy maps. We recommend attaching output policy maps to all ports that are in use. We also recommend putting any unused ports in the shutdown state by entering the **shutdown** interface configuration command. For example, if you attach an output policy map that shapes DSCP 23 traffic to a port, DSCP traffic that is sent out of any other port without a policy map attached could be incorrectly scheduled or ordered incorrectly with respect to other traffic sent out of the same port.
- We strongly recommended that you disable port speed autonegotiation when you attach an output policy map to a port to prevent the port from autonegotiating to a rate that would make the output policy map invalid. You can configure a static port speed by using the **speed** interface configuration command. If an output policy-map is configured on a port that is set for autonegotiation and the speed autonegotiates to a value that invalidates the policy, the port is put in the error-disabled state.
- You can attach only one output policy map per port.
- The maximum number of policy maps configured on the switch is 256.

These sections describe how to configure different types of output policy maps:

- Configuring Output Policy Maps with Class-Based-Weighted-Queuing, page 672
- Configuring Output Policy Maps with Class-Based Shaping, page 673
- Configuring Output Policy Maps with Port Shaping, page 674
- Configuring Output Policy Maps with Class-Based Priority Queuing, page 675
- Configuring Output Policy Maps with Weighted Tail Drop, page 679

Configuring Output Policy Maps with Class-Based-Weighted-Queuing

You use the **bandwidth** policy-map class configuration command to configure class-based weighted fair queuing (CBWFQ). CBWFQ sets the relative precedence of a queue by allocating a portion of the total bandwidth that is available for the port.

Follow these guidelines when configuring CBWFQ:

- When configuring bandwidth in a policy map, all rate configurations must be in the same format, either a configured rate or a percentage.
- The total rate of the minimum bandwidth guarantees for each queue of the policy cannot exceed the total speed for the interface.
- You cannot configure CBWFQ (**bandwidth**) and traffic (**shape average**) or priority queuing (**priority**) for the same class in an output policy map.
- You cannot configure bandwidth as an absolute rate or a percentage of total bandwidth when strict priority (priority without police) is configured for another class map.
- You can configure bandwidth as a percentage of remaining bandwidth only when strict priority (priority without police) is configured for another class in the output policy map.
- When you configure CIR bandwidth for a class as an absolute rate or a percentage of total bandwidth, any excess bandwidth that remains after servicing the CIR of all classes in the policy map is divided among the classes the same proportion as the CIR rates. If you configure the CIR rate of a class to be 0, that class is not eligible for any excess bandwidth and will receive no bandwidth.

Beginning in privileged EXEC mode, follow these steps to use CBWFQ to control bandwidth allocated to a traffic class by specifying a minimum bandwidth as a bit rate or a percentage:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**policy-map** *policy-map-name*	Create a policy map by entering the policy map name, and enter policy-map configuration mode.
3.	**class** {*class-map-name* | **class-default**}	Enter a *child class-map name* or **class-default** to match all unclassified packets, and enter policy-map class configuration mode.
4.	**bandwidth** {*rate* **| percent** *value* | **remaining percent** *value*}	Set output bandwidth limits for the policy-map class. ■ Enter a *rate* to set bandwidth in kilobits per second. The range is from 64 to 1000000. ■ Enter **percent** *value* to set bandwidth as a percentage of the total bandwidth. The range is 1 to 100 percent. ■ Enter **remaining percent** *value* to set bandwidth as a percentage of the remaining bandwidth. The range is 1 to 100 percent. This keyword is valid only when strict priority (priority without police) is configured for another class in the output policy map. You must specify the same units in each bandwidth configuration in an output policy (absolute rates or percentages). The total guaranteed bandwidth cannot exceed the total available rate.
5.	**exit**	Return to policy-map configuration mode.
6.	**exit**	Return to global configuration mode.

	Command	Purpose
7.	**interface** *interface-id*	Enter interface configuration mode for the interface to which you want to attach the policy.
8.	**service-policy output** *policy-map-name*	Attach the policy map (created in Step 2) to the egress interface.
9.	**end**	Return to privileged EXEC mode.
10.	**show policy-map** [*policy-map-name* [**class** *class-map-name*]]	Verify your entries.
11.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

After you have created an output policy map, you attach it to an egress port. See Attaching a Traffic Policy to an Interface, page 651.

Use the **no** form of the appropriate command to delete an existing policy map, class map, or bandwidth configuration.

This example shows how to set the precedence of a queue by allocating 25 percent of the total available bandwidth to the traffic class defined by the class map:

```
Switch(config)# policy-map gold_policy
Switch(config-pmap)# class out_class-1
Switch(config-pmap-c)# bandwidth percent 25
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# service-policy output gold_policy
Switch(config-if)# exit
```

Configuring Output Policy Maps with Class-Based Shaping

You use the **shape average** policy-map class configuration command to configure traffic shaping. Class-based shaping is a control mechanism that is applied to classes of traffic leaving an interface and uses the shape average command to limit the rate of data transmission used for the committed information rate (CIR) for the class.

Follow these guidelines when configuring class-based shaping:

- Configuring a queue for traffic shaping sets the maximum bandwidth or peak information rate (PIR) of the queue. Configuring traffic shaping automatically also sets the minimum bandwidth guarantee or CIR of the queue to the same value as the PIR.
- You cannot configure CBWFQ (**bandwidth**) or priority queuing (**priority**) and traffic (**shape average**) for the same class in an output policy map.
- You cannot configure traffic shaping for a traffic class when strict priority (priority without police) is configured for another class within the output policy-map.

Beginning in privileged EXEC mode, follow these steps to use class-based shaping to configure the maximum permitted average rate for a class of traffic:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**policy-map** *policy-map-name*	Create a policy map by entering the policy map name, and enter policy-map configuration mode.
3.	**class** {*class-map-name* | **class-default**}	Enter a *child class-map name* or **class-default** to match all unclassified packets, and enter policy-map class configuration mode.
4.	**shape average** *target bps*	Specify the average class-based shaping rate. For *target bps*, specify the average bit rate in bits per second. The range is from 64000 to 1000000000.
5.	**exit**	Return to policy-map configuration mode.
6.	**exit**	Return to global configuration mode.
7.	**interface** *interface-id*	Enter interface configuration mode for the interface to which you want to attach the policy.
8.	**service-policy output** *policy-map-name*	Attach the policy map (created in Step 2) to the egress interface.
9.	**end**	Return to privileged EXEC mode.
10.	**show policy-map** [*policy-map-name* [**class** *class-map-name*]]	Verify your entries.
11.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

After you have created an output policy map, you attach it to an egress port. See Attaching a Traffic Policy to an Interface, page 651.

Use the **no** form of the appropriate command to delete an existing policy map or class map or to delete a class-based shaping configuration.

Configuring Output Policy Maps with Port Shaping

Port shaping is applied to all traffic leaving an interface. It uses a policy map with only class default when the maximum bandwidth for the port is specified by using the **shape average** command. A child policy can be attached to the class-default in a hierarchical policy map format to specify class-based actions for the queues on the shaped port.

The total of the minimum bandwidth guarantees (CIR) for each queue of the child policy cannot exceed the total port shape rate.

Beginning in privileged EXEC mode, follow these steps to use port shaping to configure the maximum permitted average rate for a class of traffic:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**policy-map** *policy-map-name*	Create a hierarchical policy map by entering the hierarchical policy map name, and enter policy-map configuration mode for the parent policy.
3.	**class class-default**	Enter a policy-map class configuration mode for the default class.
4.	**shape average** *target bps*	Specify the average class-based shaping rate. For *target bps*, specify the average bit rate in bits per second. The range is from 4000000 to 1000000000.

	Command	Purpose
5.	**service-policy** *policy-map-name*	Specify the child policy-map to be used in the hierarchical policy map if required.
6.	**exit**	Return to policy-map configuration mode.
7.	**exit**	Return to global configuration mode.
8.	**interface** *interface-id*	Enter interface configuration mode for the interface to which you want to attach the policy.
9.	**service-policy output** *policy-map-name*	Attach the parent policy map (created in Step 2) to the egress interface.
10.	**end**	Return to privileged EXEC mode.
11.	**show policy-map** [*policy-map-name* [**class** *class-map-name*]]	Verify your entries.
12.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

After you have created the hierarchical output policy map, you attach it to an egress port. See Attaching a Traffic Policy to an Interface, page 651.

Use the **no** form of the appropriate command to delete an existing hierarchical policy map, to delete a port shaping configuration, or to remove the policy map from the hierarchical policy map.

This example shows how to configure port shaping by configuring a hierarchical policy map that shapes a port to 90 Mbps, allocated according to the *out-policy* policy map configured in the previous example.

```
Switch(config)# policy-map out-policy-parent
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# shape average 90000000
Switch(config-pmap-c)# service-policy out-policy
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy output out-policy-parent
Switch(config-if)# exit
```

Configuring Output Policy Maps with Class-Based Priority Queuing

You can use the **priority** policy-map class configuration command to ensure that a particular class of traffic is given preferential treatment. With strict priority queuing, the priority queue is constantly serviced; all packets in the queue are scheduled and sent until the queue is empty. Excessive use of the priority queues can possibly delay packets in other queues and create unnecessary congestion.

You can configure strict priority queuing (priority without police), or you can configure an unconditional priority policer (priority with police). Follow these guidelines when configuring priority queuing:

- You can associate the **priority** command with a single unique class for all attached output policies on the switch.
- When you configure a traffic class as a priority queue, you can configure only **police** and **queue-limit** as other queuing actions for the same class. You cannot configure **bandwidth** or **shape average** with priority queues in the same class.
- You cannot associate the **priority** command with the **class-default** of the output policy map.

Configuring Priority Without Police

Follow these guidelines when configuring strict priority queuing (priority without police):

- You cannot configure priority queuing without policing for a traffic class when class-based shaping (**shape average**) or CBWFQ (**bandwidth**) is configured for another class within the output policy-map.
- When you configure priority queuing without policing for a traffic class, you can only configure the other queues for sharing by using the **bandwidth remaining percent** policy-map class configuration command to allocate excess bandwidth. This command does not guarantee the allocated bandwidth, but does ensure the rate of distribution.

Beginning in privileged EXEC mode, follow these steps to configure a strict priority queue:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**class-map** *class-map-name*	Create classes for three egress queues. Enter match conditions classification for each class.
3.	**policy-map** *policy-map-name*	Create a policy map by entering the policy map name, and enter policy-map configuration mode.
4.	**class** *class-map-name*	Enter the name of the priority class (created by using the **class-map** global configuration command), and enter policy-map class configuration mode for the priority class.
5.	**priority**	Set the strict scheduling priority for this class. **Note:** Only one unique class map on the switch can be associated with a **priority** command. You cannot configure priority along with any other queuing action (**bandwidth** or **shape average**).
6.	**exit**	Exit policy-map class configuration mode for the priority class.
7.	**class** *class-map-name*	Enter the name of a nonpriority class, and enter policy-map class configuration mode for that class.
8.	**bandwidth remaining percent** *value*	Set output bandwidth limits for the policy-map class as a percentage of the remaining bandwidth. The range is 1 to 100 percent.
9.	**exit**	Exit policy-map class configuration mode for the class
10.	**exit**	Return to global configuration mode.
11.	**interface** *interface-id*	Enter interface configuration mode for the interface to which you want to attach the policy.
12.	**service-policy output** *policy-map-name*	Attach the policy map (created in Step 3) to the egress interface.
13.	**end**	Return to privileged EXEC mode.
14.	**show policy-map**	Verify your entries.
15.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

After you have created an output policy map, you attach it to an egress port. See Attaching a Traffic Policy to an Interface, page 651.

Use the **no** form of the appropriate command to delete an existing policy map or class map or to cancel strict priority queuing for the priority class or the bandwidth setting for the other classes.

This example shows how to configure the class *out-class1* as a strict priority queue so that all packets in that class are sent before any other class of traffic. Other traffic queues are configured so that *out-class-2* gets 50 percent of the remaining bandwidth and *out-class3* gets 20 percent of the remaining bandwidth. The class **class-default** receives the remaining 30 percent with no guarantees.

```
Switch(config)# policy-map policy1
Switch(config-pmap)# class out-class1
Switch(config-pmap-c)# priority
Switch(config-pmap-c)# exit
Switch(config-pmap)# class out-class2
Switch(config-pmap-c)# bandwidth remaining percent 50
Switch(config-pmap-c)# exit
Switch(config-pmap)# class out-class3
Switch(config-pmap-c)# bandwidth remaining percent 20
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# service-policy output policy1
Switch(config-if)# exit
```

Configuring Priority With Police

You can use the priority with police feature and configure an unconditional priority policer to limit the bandwidth used by the priority queue and allocate bandwidth or shape other queues. Follow these guidelines when configuring priority with police:

- You cannot configure a policer committed burst size for an unconditional priority policer even though the keyword is visible in the CLI help. Any configured burst size is ignored when you try to attach the output service policy.
- The allowed police rate range is 64000 to 1000000000 bps, even though the range that appears in the CLI help is 8000 to 1000000000. You cannot attach an output service policy with an out-of-range rate.
- You cannot configure priority with policing for a traffic class when **bandwidth remaining percent** is configured for another class in the same output policy map.
- You can configure 1-rate, 2-color policers for output policies with priority. You cannot configure 2-rate, 3-color policers for output policies.

Beginning in privileged EXEC mode, follow these steps to configure priority with police:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**class-map** *class-map-name*	Create classes for three egress queues. Enter match conditions classification for each class.
3.	**policy-map** *policy-map-name*	Create a policy map by entering the policy map name, and enter policy-map configuration mode.
4.	**class** *class-map-name*	Enter the name of the priority class (created by using the **class-map** global configuration command), and enter policy-map class configuration mode for the priority class.
5.	**priority**	Configure this class as the priority class. **Note:** Only one unique class map on the switch can be associated with a **priority** command.

	Command	Purpose
6.	**police** {*rate-bps* | **cir** *cir-bps*}	Define a policer for the priority class of traffic. ■ For *rate-bps,* specify average traffic rate in bits per second (bps). The range is 64000 to 1000000000. **Note:** When you use the **police** command with the **priority** command in an output policy, the police rate range and the CIR range is 64000 to 1000000000 bps, even though the range that appears in the CLI help is 8000 to 1000000000. You cannot attach an output service policy with an out-of-range rate. ■ For **cir** *cir-bps,* specify a committed information rate (CIR) in bits per second (bps). The range is 64000 to 1000000000. **Note:** Although visible in the command-line help string, the burst-size option is not supported in output policy maps. You cannot attach an output service policy map that has a configured burst size.
7.	**conform-action** [**transmit**]	(Optional) Enter the action to be taken on packets that conform to the CIR. If no action is entered, the default action is to send the packet. **Note:** You can enter a single conform-action as part of the command string following the **police** command. You can also enter a carriage return after the **police** command and enter policy-map class police configuration mode to enter the conform-action. When the *police* command is configured with priority in an output policy map, only the default conform-action of **transmit** is supported. Although visible in the command-line help string, the other police conform actions are not supported in output policy maps.
8.	**exceed-action** [**drop**]	(Optional) Enter the action to be taken for packets that do not conform to the CIR. If no action is entered, the default action is to drop the packet. **Note:** You can enter a single exceed-action as part of the command string following the **police** command. You can also enter a carriage return after the **police** command and enter policy-map class police configuration mode to enter the exceed-action. When the *police* command is configured with priority in an output policy map, only the default exceed-action of **drop** is supported. Although visible in the command-line help string, the other police exceed actions are not supported in output policy maps.
9.	**exit**	Exit policy-map class configuration mode for the priority class.
10.	**class** *class-map-name*	Enter the name of the first nonpriority class, and enter policy-map class configuration mode for that class.
11.	**bandwidth** {**rate** | **percent** *value*} or **shape average** *target bps*	Set output bandwidth limits for the policy-map class in kilobits per second (the range is 64 to 1000000) or a percentage of the total bandwidth (the range is 1 to 100 percent) or specify the average class-based shaping rate in bits per second (the range is 64000 to 1000000000).
12.	**exit**	Return to policy-map configuration mode.

	Command	Purpose
13.	**exit**	Return to global configuration mode.
14.	**interface** *interface-id*	Enter interface configuration mode for the interface to which you want to attach the policy.
15.	**service-policy output** *policy-map-name*	Attach the policy map (created in Step 3) to the egress interface.
16.	**end**	Return to privileged EXEC mode.
17.	**show policy-map**	Verify your entries.
18.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

After you have created an output policy map, you attach it to an egress port. See Attaching a Traffic Policy to an Interface, page 651.

Use the **no** form of the appropriate command to delete an existing policy map or class map or to cancel the priority queuing or policing for the priority class or the bandwidth setting for the other classes.

This example shows how to use the **priority** with **police** commands to configure *out-class1* as the priority queue, with traffic going to the queue limited to 20000000 bps so that the priority queue never uses more than that. Traffic above that rate is dropped. The other traffic queues are configured as in the previous example.

```
Switch(config)# policy-map policy1
Switch(config-pmap)# class out-class1
Switch(config-pmap-c)# priority
Switch(config-pmap-c)# police 20000000
Switch(config-pmap-c)# exit
Switch(config-pmap)# class out-class2
Switch(config-pmap-c)# bandwidth percent 50
Switch(config-pmap-c)# exit
Switch(config-pmap)# class out-class3
Switch(config-pmap-c)# bandwidth percent 20
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# service-policy output policy1
Switch(config-if)# exit
```

Configuring Output Policy Maps with Weighted Tail Drop

Weighted tail drop (WTD) adjusts the queue size (buffer size) associated with a traffic class. You configure WTD by using the **queue-limit** policy-map class configuration command.

Follow these guidelines when configuring WTD:

- Configuring WTD with the **queue-limit** command is supported only when you first configure a scheduling action, such as **bandwidth**, **shape average**, or **priority**. The exception to this is when you are configuring **queue-limit** in the **class-default**.
- You can configure and attach as many output policy maps as there are ports. Multiple output policy maps can use the same queue-limit configuration. However, these policy maps can have only three unique queue-limit configurations.
- You can use the **queue-limit** command to configure the queue-limit for CPU-generated traffic.
- When you use the **queue-limit** command to configure queue thresholds for a class, the WTD thresholds must be less than or equal to the queue maximum threshold. A queue size configured with no qualifier must be larger than any queue sizes configured with qualifiers.

- You cannot configure more than two unique threshold values for the WTD qualifiers (**cos**, **dscp**, **precedence**, or **qos-group**) in the **queue-limit** command. However, there is no limit to the number of qualifiers that you can map to those thresholds. You can configure a third unique threshold value to set the maximum queue, using the **queue-limit** command with no qualifiers.

- A WTD qualifier in the **queue-limit** command must be the same as at least one **match** qualifier in the associated class map.

- In an output policy map, when you configure a queue-limit for a unique class, all other output policy maps must use the same format of qualifier type and qualifier value. Only queue-limit threshold values can be different. For example, when you configure class A queue-limit thresholds for **dscp 30** and **dscp 50** in *policy-map1*, and you configure class A queue-limits in policy-map 2, you must use **dscp 30** and **dscp 50** as qualifiers. You cannot use **dscp 20** and **dscp 40**. The threshold values can be different, but different threshold values would create a new unique queue-limit configuration.

Beginning in privileged EXEC mode, follow these steps to use WTD to adjust the queue size for a traffic class:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**policy-map** *policy-map-name*	Create a policy map by entering the policy map name, and enter policy-map configuration mode.
3.	**class** {*class-map-name* | **class-default**}	Enter a child class-map name, or **class-default** to match all unclassified packets, and enter policy-map class configuration mode. ■ If you enter a class-map name, you must perform Step 4 to configure a scheduling action (**bandwidth**, **shape average**, or **priority)** before you go to Step 5 to configure queue-limit. ■ If you enter class-default, you can skip Step 4.
4.	**bandwidth** {*rate* | **percent** *value* | **remaining percent** *value*} or **shape average** *target bps* or **priority**	Configure a scheduling action for the traffic class. For more information, see Configuring Output Policy Maps with Class-Based-Weighted-Queuing, page 672, Configuring Output Policy Maps with Class-Based Shaping, page 673, Configuring Output Policy Maps with Port Shaping, page 674, or Configuring Output Policy Maps with Class-Based Priority Queuing, page 675.

	Command	Purpose
5.	**queue-limit** [**cos** *value* | **dscp** *value* **|** **precedence** *value* **|** **qos-group** *value*] *number-of-packets* [**packets**]}	Specify the queue size for the traffic class. ■ (Optional) For **cos** *value*, specify a CoS value. The range is from 0 to 7. ■ (Optional) For **dscp** *value*, specify a DSCP value. The range is from 0 to 63. ■ (Optional) For **precedence** *value*, specify an IP precedence value. The range is from 0 to 7. ■ (Optional) For **qos-group** *value*, enter a QoS group value. The range is from 0 to 99. ■ For *number-of-packets*, set the minimum threshold for WTD. The range is from 16 to 544, in multiples of 16, where each packet is a fixed unit of 256 bytes. **Note:** For optimal performance, we strongly recommend that you configure the queue-limit to 272 or less. The value is specified in packets by default, but the **packets** keyword is optional. **Note:** Multiple output policy maps can use the same queue-limit configuration. However these policy maps can have only three unique queue-limit configurations.
6.	**exit**	Return to policy-map configuration mode.
7.	**exit**	Return to global configuration mode.
8.	**interface** *interface-id*	Enter interface configuration mode for the interface to which you want to attach the policy.
9.	**service-policy output** *policy-map-name*	Attach the policy map (created in Step 2) to the egress interface. **Note:** If you try to attach an output policy map that contains a fourth queue-limit configuration, you see an error message, and the attachment is not allowed.
10.	**end**	Return to privileged EXEC mode.
11.	**show policy-map** [*policy-map-name* [**class** *class-map-name*]]	Verify your entries.
12.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

After you have created an output policy map, you attach it to an egress port. See Configuring Output Policy Maps, page 670.

Use the **no** form of the appropriate command to delete an existing policy map or class map or to delete a WTD configuration.

This example shows a policy map with a specified bandwidth and queue size. Traffic that is not DSCP 30 or 10 is assigned a queue limit of 112 packets. Traffic with a DSCP value of 30 is assigned a queue-limit of 48 packets, and traffic with a DSCP value of 10 is assigned a queue limit of 32 packets. All traffic not belonging to the class traffic is classified into class-default, which is configured with 10 percent of the total available bandwidth and a large queue size of 256 packets.

```
Switch(config)# policy-map gold-policy
Switch(config-pmap)# class traffic
Switch(config-pmap-c)# bandwidth percent 50
Switch(config-pmap-c)# queue-limit 112
Switch(config-pmap-c)# queue-limit dscp 30 48
Switch(config-pmap-c)# queue-limit dscp 10 32
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# bandwidth percent 10
Switch(config-pmap-c)# queue-limit 256
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface GigabitEthernet1/17
Switch(config-if)# service-policy output gold-policy
Switch(config-if)# exit
```

Configuring QoS Marking and Queuing for CPU-Generated Traffic

Beginning in privileged EXEC mode, follow these steps to configure marking and queuing of CPU-generated traffic. This procedure is optional.

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	Configure global table maps	Refer to the Configuring Table Maps, page 650.
3.	**cpu traffic qos cos {***cos-value* **| cos [table-map** *table-map-name***] | dscp [table-map** *table-map-name***] | prec [table-map** *table-map-name***]}**	Mark traffic by setting a new CoS value or by specifying a table map. ■ For *cos-value*, enter a new CoS value. The range is from 0 to 7. ■ You can also mark CoS based on the CoS, DSCP, or IP-precedence value. You can optionally use a table map to configure CoS. If you do not enter **table-map** *table-map-name*, the table map default behavior is **copy**. See Table Maps, page 625. When you complete this step, go to Step 7. on page 683.
4.	**cpu traffic qos dscp {***dscp_value* **| cos [table-map** *table-map-name***] | dscp [table-map** *table-map-name***] | prec [table-map** *table-map-name***]}**	Mark traffic by setting a new DsCP value or by specifying a table map. ■ For **dscp** *new-dscp*, enter a new DSCP value for the classified traffic. The range is 0 to 63. ■ You can also configure a table map to mark DSCP based on the CoS, DSCP, or IP-precedence value. You can optionally enter the table name. If you do not enter **table-map** *table-map-map* name, the table map default behavior is **copy**. See Table Maps, page 625. ■ For additional DSCP classification options, see Classification Based on IP DSCP, page 620. When you complete this step, go to Step 7. on page 683.

	Command	Purpose
5.	**cpu traffic qos precedence {***precedence_value* **| cos [table-map** *table-map-name***] | dscp [table-map** *table-map-name***] | prec** [**table-map** *table-map-name***]}**	Mark traffic by setting a new precedence value or by specifying a table map. ■ For **precedence** *new-precedence*, enter a new IP-precedence value as a number from 0 to 7 or by name: **routine** (**0**), **priority** (**1**), **immediate** (**2**), **flash** (**3**), **flash-override** (**4**), **critical** (**5**), **internet** (**6**), **network** (**7**). ■ You can also configure a table map to mark precedence based on the CoS, DSCP, or IP-precedence value. You can optionally enter the table name. If you do not enter **table-map** *table-map-map name*, the table map default behavior is **copy**. See Table Maps, page 625. When you complete this step, go to Step 7. on page 683.
6.	**cpu traffic qos qos-group** *qos-group-value*	Mark traffic by using a QoS group. For *qos-group-value*, identify a QoS group to use at egress. The range is 0 to 99. When you complete this step, go to Step 7. on page 683.
7.	**end**	Return to privileged EXEC mode.
8.	Configure output policy maps to map QoS markings like COS, IP DSCP, IP precedence and QoS group to class queues, configure queuing and scheduling	Refer to the Configuring Output Policy Maps, page 670.
9.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.
10.	**show running-config**	Display the configured class maps, policy maps, table maps, and CPU traffic QoS settings.
11.	**show cpu traffic qos**	Display the QoS marking values for CPU-generated traffic.
12.	**show table-map [***table-map-name]*	Display information for all table maps or the specified table map.
13.	**show policy-map** [*policy-map-name* | **interface** [*interface-id*] [**output**] [**class** *class-name*]]	Display QoS policy map information for the specified policy map name, interface, input or output policy maps, or policy-map class.

To disable any command, use the **no** form of the command.

Example 1

This example shows how to configure egress queuing based on the DSCP value of CPU-generated IP packets.

- All CPU-generated IP traffic queues on the egress port, based on its IP DSCP value, and the configured output policy map *output-policy*.
- All IP SLA or TWAMP probes with the DSCP value *ef* to simulate voice traffic are assigned to the *voice* class.
- All IP SLA or TWAMP probes with the DSCP values *af41*, *af42* and *af43* to simulate video traffic are assigned to the *video* class.
- All IP control protocol traffic with the DSCP values 48 and 56 are assigned to the *network-internetwork-control* class.
- The rest of the IP traffic is assigned to the default class.
- All CPU-generated non-IP traffic is statically mapped to a fixed queue on the egress port.

- All CFM traffic is queued to the default class because there is no class based on CoS.

```
Switch(config)# cpu traffic qos dscp dscp
```

Class:

```
Switch(config)# class-map match-any video
Switch(config-cmap)# match ip dscp af41 af42 af43
Switch(config-cmap)# exit

Switch(config)# class-map match-any voice
Switch(config-cmap)# match ip dscp ef
Switch(config-cmap)# exit

Switch(config)# class-map match-any network-internetwork-control
Switch(config-cmap)# match ip dscp 48 56
Switch(config-cmap)# exit
```

Policy:

```
Switch(config)# policy-map output-policy
Switch(config-pmap)# class voice
Switch(config-pmap-c)# priority
Switch(config-pmap-c)# police cir 10000000
Switch(config-pmap-c)# exit
Switch(config-pmap)# class video
Switch(config-pmap-c)# bandwidth percent 40
Switch(config-pmap-c)# exit
Switch(config-pmap)# class network-internetwork-control
Switch(config-pmap-c)# bandwidth percent 10
Switch(config-pmap-c)# exit
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# bandwidth percent 30
Switch(config-pmap-c)# exit
```

Interface

```
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy output output-policy
Switch(config-pmap-c)# exit
```

Example 2

This example shows how to mark the CoS of CPU-generated IP traffic (including IP-SLA and TWAMP) based on the DSCP value in the packet and to configure egress queuing based on the CoS value.

- All CPU-generated IP traffic queues on the egress port, based on the IP DSCP value and the configured output policy map called *output-policy*.
- All IP SLA or TWAMP probes with the DSCP value *ef* to simulate voice traffic are assigned to the *voice* class.
- All IP SLA or TWAMP probes with the DSCP values *af41*, *af42* and *af43* to simulate video traffic are assigned to the *video* class.
- All IP control protocol traffic with the DSCP values 48 and 56 are assigned to the *network-internetwork-control* class.
- The rest of the IP traffic is assigned to the default class.
- All CPU-generated non-IP traffic with CoS 5 is assigned to the *voice* class.
- All CPU-generated non-IP traffic with CoS 3 is assigned to the *video* class.

- All CPU-generated non-IP traffic with CoS 6 and 7 is assigned to the *network-internetwork-control* class.
- All CFM traffic with CoS 5 is assigned to the *voice* class.
- All CFM traffic with CoS 3 is assigned to the *video* class.
- All CFM traffic with CoS 6 and 7 is assigned to the *network-internetwork-control* class.

Table Map:

```
Switch(config)# table-map dscp-to-cos
Switch(config-tablemap)# map from 46 to 5
Switch(config-tablemap)# map from 48 to 6
Switch(config-tablemap)# map from 56 to 7
Switch(config-tablemap)# map from af41 to 3
Switch(config-tablemap)# map from af42 to 3
Switch(config-tablemap)# map from af43 to 3
Switch(config-tablemap)# default 0
Switch(config-tablemap)# end
```

CPU QoS:

```
Switch(config)# cpu traffic qos cos dscp table-map dscp-to-cos
Switch(config)# cpu traffic qos cos cos
```

Class:

```
Switch(config)# class-map match-any video
Switch(config-cmap)# match cos 3
Switch(config-cmap)# exit

Switch(config)# class-map match-any voice
Switch(config-cmap)# match cos 5
Switch(config-cmap)# exit

Switch(config)# class-map match-any network-internetwork-control
Switch(config-cmap)# match cos 6 7
Switch(config-cmap)# exit
```

Policy:

```
Switch(config)# policy-map output-policy
Switch(config-pmap)# class voice
Switch(config-pmap-c)# priority
Switch(config-pmap-c)# police cir 10000000
Switch(config-pmap-c)# exit
Switch(config-pmap)# class video
Switch(config-pmap-c)# bandwidth percent 40
Switch(config-pmap-c)# exit
Switch(config-pmap)# class network-internetwork-control
Switch(config-pmap-c)# bandwidth percent 10
Switch(config-pmap-c)# exit
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# bandwidth percent 30
Switch(config-pmap-c)# exit
```

Interface

```
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy output output-policy
Switch(config-pmap-c)# exit
```

Example 3

This example shows how to:

- Mark the DSCP value of CPU-generated IP traffic (including IP-SLA and TWAMP) based on the DSCP value in the packet.
- Mark the CoS of CPU-generated IP traffic (including IP-SLA and TWAMP) based on the DSCP value in the packet.
- Mark the CoS of CPU-generated non-IP traffic based on the CoS value in the packet.
- Mark all CPU-generated traffic with the QoS group.
- Configure egress queuing based on the QoS group.

The example has these results:

- All CPU-generated IP traffic with DSCP values 46, 48, and 56 retains the existing markings.
- For all other CPU-generated IP packets, the DSCP value is reset to 0.
- All CPU-generated IP traffic with DSCP values 46, 48, and 56 is mapped to the corresponding CoS values of 5, 6, and 7 respectively.
- For all other CPU-generated IP packets, the CoS value resets to 0.
- All CPU-generated non-IP traffic with the CoS values of 5, 6, and 7 retain the existing markings.
- For all other CPU-generated non-IP packets, the CoS value resets to 0.
- All CPU-generated traffic goes through a single class called *cpu-traffic*. The *user-voice* classes *user-voice* and *user-video* are reserved for user traffic. As a result, CPU traffic and user traffic are separated into different queues on the egress port.

Table Map

```
Switch(config)# table-map dscp-to-cos
Switch(config-tablemap)# map from 46 to 5
Switch(config-tablemap)# map from 48 to 6
Switch(config-tablemap)# map from 56 to 7
Switch(config-tablemap)# default 0
Switch(config-tablemap)# end

Switch(config)# table-map dscp-to-dscp
Switch(config-tablemap)# map from 46 to 46
Switch(config-tablemap)# map from 48 to 48
Switch(config-tablemap)# map from 56 to 56
Switch(config-tablemap)# default 0
Switch(config-tablemap)# end

Switch(config)# table-map cos-to-cos
Switch(config-tablemap)# map from 5 to 5
Switch(config-tablemap)# map from 6 to 6
Switch(config-tablemap)# map from 7 to 7
Switch(config-tablemap)# default 0
Switch(config-tablemap)# end
```

CPU QoS:

```
Switch(config)# cpu traffic qos dscp dscp table-map dscp-to-dscp
Switch(config)# cpu traffic qos cos dscp table dscp-to-cos
Switch(config)# cpu traffic qos cos cos table cos-to-cos
Switch(config)# cpu traffic qos qos-group 50
```

Class:

```
Switch(config)# class-map match-any cpu-traffic
Switch(config-cmap)# match qos-group 50
Switch(config-cmap)# exit

Switch(config)# class-map match-any user-video
Switch(config-cmap)# match cos 3
Switch(config-cmap)# exit

Switch(config)# class-map match-any user-voice
Switch(config-cmap)# match cos 5
Switch(config-cmap)# exit

Policy:
Switch(config)# policy-map output-policy
Switch(config-pmap)# class user-voice
Switch(config-pmap-c)# priority
Switch(config-pmap-c)# police cir 10000000
Switch(config-pmap-c)# exit
Switch(config-pmap)# class user-video
Switch(config-pmap-c)# bandwidth percent 40
Switch(config-pmap-c)# exit
Switch(config-pmap)# class cpu-traffic
Switch(config-pmap-c)# bandwidth percent 10
Switch(config-pmap-c)# exit
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# bandwidth percent 30
Switch(config-pmap-c)# exit
```

Interface:

```
Switch(config)# interface fastethernet0/1
Switch(config-if)# service-policy output output-policy
Switch(config-pmap-c)# exit
```

Displaying QoS Information

To display QoS information, use one or more of the privileged EXEC commands in Table 58 on page 687.

Table 58 Commands for Displaying Standard QoS Information

Command	Purpose
show class-map [*class-map-name*]	Display QoS class-map information for all class maps or the specified class map.
show policer aggregate [*aggregate-policer-name*]	Display information about all aggregate policers or the specified aggregate policer.
show policy-map [*policy-map-name* | **interface** [*interface-id*] [**input** | **output**] [**class** *class-name*]]	Display QoS policy map information for the specified policy map name, interface, input or output policy maps, or policy-map class.

Table 58 Commands for Displaying Standard QoS Information (continued)

Command	Purpose
show cpu traffic qos	Display the QoS marking values for CPU-generated traffic.
show running-config	Display the configured class maps, policy maps, table maps, and aggregate policers.
show table-map [*table-map-name*]	Display information for all configured table maps or the specified table map.

To test full-path QoS in both directions on an interface, you can configure Ethernet terminal loopback by entering the **ethernet loopback facility** interface configuration command. In terminal loopback mode, the port appears to be up but the link is actually down and no packets are sent out. Configuration changes on the port immediately affect the traffic being looped back.

QoS Statistics

There are several ways to display QoS input and output policy-map statistics.

For input policy maps, you can use the **show policy-map interface** [*interface-id*] privileged EXEC command to display per-class per-policer conform and exceed statistics. Policer conform statistics are the number of packets that conform to the configured policer profile; policer exceed statistics are the number of packets that exceed the configured policer profile. The switch does not support per-class classification statistics, but you can determine these statistics by configuring policing at line rate for the class. In this case, no packets exceed the configured policer profile, and the policer conform statistics would equal the class classification statistics.

This output also includes byte-level statistics for conform, exceed, and violate classes.

Another way to view input QoS statistics is in the output of the **show platform qos statistics interface** [*interface-id*] privileged EXEC command. The per-port frame statistics are sorted by the DSCP and CoS values of the incoming frames on the port. These statistics do not provide any information about the MQC input policy map configured on the interface.

For output policy maps, you can use the **show policy-map interface** [*interface-id*] command to display per-class classification statistics that show the total number of packets that match the specified class. This count includes the total number of packets that are sent and dropped for that class. You can use the same command to view the per-class tail drop statistics.

Configuration Examples for Policy Maps

This section includes configuration examples for configuring QoS policies on the Cisco IE switch, including configuration limitations and restrictions. The sections are broken into different configurations actions that a customer might do. Each section provides the exact sequence of steps that you must follow for successful configuration or modification.

- QoS Configuration for Customer A, page 689
- QoS Configuration for Customer B, page 690
- Modifying Output Policies and Adding or Deleting Classification Criteria, page 691
- Modifying Output Policies and Changing Queuing or Scheduling Parameters, page 691
- Modifying Output Policies and Adding or Deleting Configured Actions, page 692
- Modifying Output Policies and Adding or Deleting a Class, page 693

QoS Configuration for Customer A

This section provides examples of the initial configuration and activation of QoS policies for a customer switch. Input and output QoS service policies are configured based on the requirements and attached to relevant ports.

In the initial configuration for Customer A, Gigabit Ethernet ports 1 and 2 are network node interfaces (NNIs) and are enabled by default.

This is the overall sequence for initial configuration:

- Configure classes and policies.
- Shut down all active ports.
- Attach policies to ports to be activated.
- Take the ports out of the shut-down state.
- Leave unused ports shut down.

Note these restrictions for configuring output policies:

- You can define up to three classes in the output policy map.
- The defined classes must be the same as other output policy maps.
- The number of defined classes in each output policy map must be same.
- You must assign an action to each class; that is, there can be no empty class.
- Each class configuration must be based on the classification/marking done in the input policy-map.

This example configures classes for input service policies and defines three classes of service: gold, silver, and bronze. Because a **match-all** classification (the default) can have only single classification criterion, the **match-any** classification is used so that you can add classification criteria in the future.

```
Switch# config terminal
Switch(config)# class-map match-any gold-in
Switch(config-cmap)# match ip dscp af11
Switch(config-cmap)# exit
Switch(config)# class-map match-any silver-in
Switch(config-cmap)# match ip dscp af21
Switch(config-cmap)# exit
Switch(config)# class-map match-any bronze-in
Switch(config-cmap)# match ip dscp af31
Switch(config-cmap)# exit
```

This example shows how to configure an input policy map that marks the gold class and polices the silver class to 50 Mb/s and the bronze class to 20 Mb/s.

```
Switch(config)# policy-map input-all
Switch(config-pmap)# class gold-in
Switch(config-pmap-c)# set ip dscp af43
Switch(config-pmap-c)# exit
Switch(config-pmap)# class silver-in
Switch(config-pmap-c)# police 50000000
Switch(config-pmap)# class bronze-in
Switch(config-pmap-c)# police 20000000
Switch(config-pmap-c)# exit
```

This example configures classes for output service policies with three classes of service: gold, silver, and bronze. The gold class is configured to match the marked value in the input service policy. Because a **match-all** classification (the default) can have only single classification criterion, the **match-any** classification is used so that you can add classification criteria in the future.

```
Switch# config terminal
Switch(config)# class-map match-any gold-out
Switch(config-cmap)# match ip dscp af43
Switch(config-cmap)# exit
Switch(config)# class-map match-any silver-out
Switch(config-cmap)# match ip dscp af21
Switch(config-cmap)# exit
Switch(config)# class-map match-any bronze-out
Switch(config-cmap)# match ip dscp af31
Switch(config-cmap)# exit
```

This example configures one output service policy to be applied to both Gigabit Ethernet NNIs, providing priority with rate-limiting to the gold class, class-based shaping for the silver class, and a minimum bandwidth guarantee of 10 percent to the bronze class.

```
Switch(config)# policy-map output-g1-2
Switch(config-pmap)# class gold-out
Switch(config-pmap-c)# priority
Switch(config-pmap-c)# police 50000000
Switch(config-pmap-c)# exit
Switch(config-pmap)# class silver-out
Switch(config-pmap-c)# shape average 200000
Switch(config-pmap-c)# exit
Switch(config-pmap)# class bronze-out
Switch(config-pmap-c)# bandwidth percent 10
Switch(config-pmap-c)# exit
```

This example attaches the input and output service policies to the Gigabit Ethernet ports and activates them.

```
Switch(config)# interface range GigabitEthernet1/17-18
Switch(config-if-range)# service-policy input input-all
Switch(config-if-range)# service-policy output output-g1-2
Switch(config-if-range)# no shutdown
Switch(config-if-range)# exit
```

QoS Configuration for Customer B

This section provides examples for configuring and activating QoS policies on the switch for a new set of customers without affecting the current customers. Input and output QoS service policies are configured based on the requirements and attached to relevant ports. The example uses an existing input policy-map and configures a new output policy map for the new customers.

In the initial configuration for Customer B, Gigabit Ethernet ports 1 and 2 are NNIs and are enabled by default.

This is the overall sequence of initial configuration:

- Define any new required output policies.
- Attach input and output policies to ports to be activated.
- Take the ports out of the shut-down state.

Note these restrictions when configuring output policies:

- You can define up to three classes in the output policy map.
- The defined classes must be the same as other output policy maps.

- The number of defined classes in each output policy map must be same.
- You must assign an action to each class; that is, there can be no empty class.
- Each class configuration must be based on the classification/marking done in the input policy-map.

Modifying Output Policies and Adding or Deleting Classification Criteria

This section provides examples of updating an existing set of output policy maps to add or delete classification criteria. The modification might be required due to a change in the service provisioning requirements or a change in the input service policy map. You can make the change without shutting down any port.

In the initial configuration, Gigabit Ethernet ports 1 and 2 are NNIs and are enabled by default.

This is the overall sequence of initial configuration:

- Change the configured class map for an input service policy.
- Change the configured class map for an output service policy.

This example modifies classes for an input service policy by adding classification criteria to the silver-in class to also match dscp cs5. This is required for the output policy-map to match to dscp cs5.

```
Switch(config)# class-map match-any silver-in
Switch(config-cmap)# match ip dscp af21
Switch(config-cmap)# match ip dscp cs5
Switch(config-cmap)# exit
```

This example modifies classes for an output service policy, adding classification criteria to the silver-out class to also match dscp cs5. This adds dscp cs5 to the silver-out class on all configured and attached output service policies. The dscp cs5 flow now receives the same queuing and scheduling treatment as the silver-out class.

```
Switch# config terminal
Switch(config)# class-map match-any silver-out
Switch(config-cmap)# match ip dscp af21
Switch(config-cmap)# match ip dscp cs5
Switch(config-cmap)# exit
```

You should use the same procedure when deleting a match statement associated with a configured class.

Modifying Output Policies and Changing Queuing or Scheduling Parameters

This section provides examples of updating an existing set of output policy maps to modify the parameters of the configured queuing and scheduling actions. The modification in the output policy map might be required due to a change in the service provisioning requirements. You can make the change without shutting down any port.

In the initial configuration, Gigabit Ethernet ports 1 and 2 are NNIs and are enabled by default.

The requirement is to change the action parameters.

Note these restrictions when configuring output policies:

- You can define up to three classes in the output policy map.
- The defined classes must be the same as other output policy maps.
- The number of defined classes in each output policy map must be same.
- You must assign an action to each class; that is, there can be no empty class.

- Each class configuration must be based on the classification or marking done in the input policy-map.

Modifying Output Policies and Adding or Deleting Configured Actions

This section provides examples of updating an existing set of output policy maps to add or delete queuing and scheduling actions. The modification in the output policy map might be required due to a change in the service provisioning requirements. You can make the change without shutting down ports that are not configured with the output policy map to be modified. But you must shut down the ports that are configured with that output policy map. Customers not using this output policy map are not affected.

In the initial configuration, Gigabit Ethernet ports 1 and 2 are NNIs and are enabled by default.

This is the overall sequence of configuration:

- Shut down all active ports carrying the policy to be modified.
- Detach the output policy from all ports to which it is attached.
- Make modifications to the output policy.
- Reattach the output policy to the appropriate ports.
- Take the ports out of the shutdown state.

Note these restrictions for configuring output policies:

- You can define up to three classes in the output policy map.
- The defined classes must be the same as other output policy maps.
- The number of defined classes in each output policy map must be same.
- You must assign an action to each class; that is, there can be no empty class.
- Each class configuration must be based on the classification/marking done in the input policy-map.

These steps shut down all ports carrying the output policy, in this case only the Gigabit Ethernet ports.

```
Switch(config)# interface range GigabitEthernet1/17-18
Switch(config-if-range)# shutdown
Switch(config-if-range)# exit
```

These steps detach the output policy to be modified, in this case the one configured on the Gigabit Ethernet ports:

```
Switch(config)# interface range GigabitEthernet1/17-18
Switch(config-if-range)# no service-policy output output-g1-2
Switch(config-if-range)# exit
```

These steps modify the output service policy servicing the Gigabit Ethernet NNIs. Instead of providing a minimum bandwidth guarantee of 10 percent to the bronze class, the policy is modified to provide class-based shaping to 100000 bps.

```
Switch(config)# policy-map output-g1-2
Switch(config-pmap)# class bronze-out
Switch(config-pmap-c)# no bandwidth percent 10
Switch(config-pmap-c)# shape average 100000
Switch(config-pmap-c)# exit
```

These steps reattach the output policy to the Gigabit Ethernet ports:

```
Switch(config)# interface range GigabitEthernet1/17-18
Switch(config-if-range)# service-policy output output9-12
Switch(config-if-range)# exit
```

These steps activate all Gigabit Ethernet ports:

```
Switch(config)# interface range GigabitEthernet1/17-18
Switch(config-if-range)# no shutdown
Switch(config-if-range)# exit
```

Modifying Output Policies and Adding or Deleting a Class

This section provides examples of updating an existing set of output policy maps to add or delete entire classes. The modification in the output policy map might be required due to a change in the service provisioning requirements or a change in the input service policy. To make this change, you must shut down all active ports on the switch. For this kind of update to any output policy map, all customers could potentially be affected. To avoid this, we recommend that you consider possible future upgrades when you configure classes in output service policies.

In the initial configuration, Gigabit Ethernet ports 1 and 2 are NNIs and are enabled by default.

This is the overall sequence of configuration:

- Shut down all active ports.
- Detach the output policies from all Ethernet ports.
- Delete the class.
- Reattach the output policies to the Ethernet ports.
- Take the Ethernet ports out of the shutdown state.

These steps shut down all active and applicable Ethernet ports:

```
Switch(config)# interface range GigabitEthernet1/17-18
Switch(config-if-range)# shutdown
Switch(config-if-range)# exit
```

These steps detach all output policies from the affected Ethernet ports:

```
Switch(config)# interface range GigabitEthernet1/17-18
Switch(config-if-range)# no service-policy output output-g1-2
Switch(config-if-range)# exit
```

These steps delete a class from all output policy maps and input policy maps; the input policy can be left attached or can be detached:

```
Switch(config)# policy-map output1-8
Switch(config-pmap)# no class bronze-out
Switch(config-pmap-c)# exit
Switch(config)# policy-map output9-12
Switch(config-pmap)# no class bronze-out
Switch(config-pmap-c)# exit
Switch(config)# policy-map output-g1-2
Switch(config-pmap)# no class bronze-out
Switch(config-pmap-c)# exit
Switch(config)# policy-map input-all
Switch(config-pmap)# no class bronze-in
Switch(config-pmap-c)# exit
```

These steps reattach all policies to the Ethernet ports:

```
Switch(config)# interface range GigabitEthernet1/17-18
Switch(config-if-range)# service-policy output output9-12
Switch(config-if-range)# exit
```

These steps activate all applicable Ethernet ports:

```
Switch(config)# interface range GigabitEthernet1/17-18
Switch(config-if-range)# no shutdown
Switch(config-if-range)# exit
```

You should use the same procedure when adding a class to an attached output service policy.

Note: Problems can occur if you do not follow the previous sequence.

When a policy map is attached to an interface, all traffic that does not explicitly match the configured class maps within the policy map should go through the default queue (class **class-default**). However, in some cases, traffic that does not explicitly match the output policy-map classes could go through more than one queue. This queuing problem can occur when you do not follow the previous procedure and do not attach an output policy to all active ports.

Configuring Static IP Unicast Routing

This chapter describes how to configure IP Version 4 (IPv4) static IP unicast routing on the switch. Static routing is supported only on switched virtual interfaces (SVIs) and not on physical interfaces. The switch does not support routing protocols.

Restrictions for Static IP Unicast Routing

- By default, static IP routing is disabled on the switch.

Information About Configuring Static IP Unicast Routing

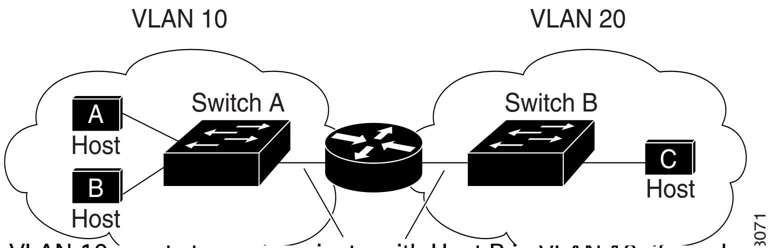
Note: When configuring routing parameters on the switch and to allocate system resources to maximize the number of unicast routes allowed, use the **sdm prefer lanbase-routing** global configuration command to set the Switch Database Management (SDM) feature to the routing template.

IP Routing

In some network environments, VLANs are associated with individual networks or subnetworks. In an IP network, each subnetwork is mapped to an individual VLAN. Configuring VLANs helps control the size of the broadcast domain and keeps local traffic local. However, network devices in different VLANs cannot communicate with one another without a Layer 3 device to route traffic between the VLANs, referred to as inter-VLAN routing. You configure one or more routers to route traffic to the appropriate destination VLAN.

Figure 85 on page 695 shows a basic routing topology. Switch A is in VLAN 10, and Switch B is in VLAN 20. The router has an interface in each VLAN.

Figure 85 Routing Topology Example



When Host A in VLAN 10 needs to communicate with Host B in VLAN 10, it sends a packet 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 uses the routing table to finds the correct outgoing interface, and forwards the packet on the VLAN 20 interface to Switch B. Switch B receives the packet and forwards it to Host C.

When static routing is enabled on Switch A and B, the router device is no longer needed to route packets.

Types of Routing

Routers and Layer 3 switches can route packets in these ways:

- Using default routing to send traffic with a destination unknown to the router to a default outlet or destination
- Using static routes to forward packets from predetermined ports through a single path into and out of a network
- Dynamically calculating routes by using a routing protocol

The switch supports static routes and default routes. It does not support routing protocols.

How to Configure Static IP Unicast Routing

Steps for Configuring Routing

In these procedures, the specified interface must be a switch virtual interface (SVI)—a VLAN interface created by using the **interface vlan** *vlan_id* global configuration command and by default a Layer 3 interface. All Layer 3 interfaces on which routing will occur must have IP addresses assigned to them. See Assigning IP Addresses to SVIs, page 696.

Note: The switch supports 16 static routes (including user-configured routes and the default route) and any directly connected routes and default routes for the management interface. The switch can have an IP address assigned to each SVI. Before enabling routing, enter the **sdm prefer lanbase-routing** global configuration command and reload the switch.

Procedures for configuring routing:

- To support VLAN interfaces, create and configure VLANs on the switch, and assign VLAN membership to Layer 2 interfaces.
- Configure Layer 3 interfaces (SVIs) and physical routed port (no switchport).
- Assign IP addresses to the Layer 3 interfaces.
- Configure static routes

Enabling IP Unicast Routing

By default, the switch is in Layer 2 switching mode, and IP routing is disabled. To use the Layer 3 capabilities of the switch, enable IP routing.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**ip routing**	Enables IP routing.
3.	**end**	Returns to privileged EXEC mode.

Assigning IP Addresses to SVIs

To configure IP routing, you need to assign IP addresses to Layer 3 network interfaces. This enables communication with the hosts on those interfaces that use IP. IP routing is disabled by default, and no IP addresses are assigned to SVIs.

An IP address identifies a destination for IP packets. Some IP addresses are reserved for special uses and cannot be used for host, subnet, or network addresses. RFC 1166, “Internet Numbers,” contains the official description of these IP addresses.

An interface can have one primary IP address. A a subnet mask identifies the bits that denote the network number in an IP address.

This task explains how to assign an IP address and a network mask to an SVI

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface vlan** *vlan_id*	Enters interface configuration mode, and specifies the Layer 3 VLAN to configure.
3.	**ip address** *ip-address subnet-mask*	Configures the IP address and IP subnet mask.
4.	**end**	Returns to privileged EXEC mode.

Configuring Static Unicast Routes

Static unicast routes are user-defined routes that cause packets moving between a source and a destination to take a specified path. Static routes can be important if the router cannot build a route to a particular destination and are useful for specifying a gateway of last resort to which all unroutable packets are sent.

Use the **no ip route** *prefix mask* {*address* | *interface*} global configuration command to remove a static route. The switch retains static routes until you remove them.

When an interface goes down, all static routes through that interface are removed from the IP routing table. When the software can no longer find a valid next hop for the address specified as the forwarding router's address in a static route, the static route is also removed from the IP routing table.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**ip route** *prefix mask* {*address* | *interface*} [*distance*]	Establishes a static route.
3.	**end**	Returns to privileged EXEC mode.

Monitoring and Maintaining the IP Network

Command	Description
show interfaces [*interface-id*]	Displays the administrative and operational status of all interface specified interface.

Additional References for Configuring IP Unicast Routing

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*
Cisco IOS IP address commands	*Cisco IOS IP Command Reference, Volume 1 of 3: Addressing and Services, Release 15.0*
Cisco IP routing configuration	*Cisco IOS IP Routing Configuration Guides, Release 15.0*
SDM template configuration	Configuring SDM Templates, page 137

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Configuring IPv6 Host Functions

This chapter describes how to configure IPv6 host functions on the switch.

Prerequisites Configuring IPv6 Host Functions

- To enable dual-stack environments (supporting both IPv4 and IPv6), you must configure the switch to use the a dual IPv4 and IPv6 switch database management (SDM) template. See Dual IPv4 and IPv6 Protocol Stacks, page 702.

Information About Configuring IPv6 Host Functions

IPv6

IPv4 users can move to IPv6 and receive services such as end-to-end security, quality of service (QoS), and globally unique addresses. The IPv6 address space reduces the need for private addresses and Network Address Translation (NAT) processing by border routers at network edges.

For information about how Cisco Systems implements IPv6, go to this URL:

http://www.cisco.com/en/US/products/ps6553/products_ios_technology_home.html

For information about IPv6 and other features in this chapter

- See the *Cisco IOS IPv6 Configuration Library* at this URL:

  http://www.cisco.com/en/US//docs/ios-xml/ios/ipv6/configuration/15-1mt/ipv6-15-1mt-book.html

This section describes IPv6 implementation on the switch. These sections are included:

- IPv6 Addresses, page 699
- Supported IPv6 Host Features, page 700
- How to Configure IPv6 Hosting, page 704

IPv6 Addresses

The switch supports only IPv6 unicast addresses. It does not support site-local unicast addresses, anycast addresses, or multicast addresses.

The IPv6 128-bit addresses are represented as a series of eight 16-bit hexadecimal fields separated by colons in the format: n:n:n:n:n:n:n:n. This is an example of an IPv6 address:

2031:0000:130F:0000:0000:09C0:080F:130B

For easier implementation, leading zeros in each field are optional. This is the same address without leading zeros:

2031:0:130F:0:0:9C0:80F:130B

You can also use two colons (::) to represent successive hexadecimal fields of zeros, but you can use this short version only once in each address:

```
2031:0:130F::09C0:080F:130B
```

For more information about IPv6 address formats, address types, and the IPv6 packet header, see the “Implementing IPv6 Addressing and Basic Connectivity” chapter of *Cisco IOS IPv6 Configuration Library* on Cisco.com.

In the “Implementing Addressing and Basic Connectivity” chapter, these sections apply to the switch:

- IPv6 Address Formats
- IPv6 Address Output Display
- Simplified IPv6 Packet Header

Supported IPv6 Host Features

These sections describe the IPv6 protocol features supported by the switch:

- 128-Bit Wide Unicast Addresses, page 700
- DNS for IPv6, page 701
- ICMPv6, page 701
- Neighbor Discovery, page 701
- Default Router Preference, page 701
- IPv6 Stateless Autoconfiguration and Duplicate Address Detection, page 701
- IPv6 Applications, page 702
- Dual IPv4 and IPv6 Protocol Stacks, page 702
- SNMP and Syslog Over IPv6, page 703
- HTTP over IPv6, page 703

Support on the switch includes expanded address capability, header format simplification, improved support of extensions and options, and hardware parsing of the extension header. The switch supports hop-by-hop extension header packets, which are routed or bridged in software.

128-Bit Wide Unicast Addresses

The switch supports aggregatable global unicast addresses and link-local unicast addresses. It does not support site-local unicast addresses.

- Aggregatable global unicast addresses are IPv6 addresses from the aggregatable global unicast prefix. The address structure enables strict aggregation of routing prefixes and limits the number of routing table entries in the global routing table. These addresses are used on links that are aggregated through organizations and eventually to the Internet service provider.

  These addresses are defined by a global routing prefix, a subnet ID, and an interface ID. Current global unicast address allocation uses the range of addresses that start with binary value 001 (2000::/3). Addresses with a prefix of 2000::/3(001) through E000::/3(111) must have 64-bit interface identifiers in the extended unique identifier (EUI)-64 format.

- Link local unicast addresses can be automatically configured on any interface by using the link-local prefix FE80::/10(1111 1110 10) and the interface identifier in the modified EUI format. Link-local addresses are used in the neighbor discovery protocol (NDP) and the stateless autoconfiguration process. Nodes on a local link use link-local addresses and do not require globally unique addresses to communicate. IPv6 routers do not forward packets with link-local source or destination addresses to other links.

For more information, see the section about IPv6 unicast addresses in the "Implementing IPv6 Addressing and Basic Connectivity" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

DNS for IPv6

IPv6 supports Domain Name System (DNS) record types in the DNS name-to-address and address-to-name lookup processes. The DNS AAAA resource record types support IPv6 addresses and are equivalent to an A address record in IPv4. The switch supports DNS resolution for IPv4 and IPv6.

ICMPv6

The Internet Control Message Protocol (ICMP) in IPv6 generates error messages, such as ICMP destination unreachable messages, to report errors during processing and other diagnostic functions. In IPv6, ICMP packets are also used in the neighbor discovery protocol and path MTU discovery.

Neighbor Discovery

The switch supports NDP for IPv6, a protocol running on top of ICMPv6, and static neighbor entries for IPv6 stations that do not support NDP. The IPv6 neighbor discovery process uses ICMP messages and solicited-node multicast addresses to determine the link-layer address of a neighbor on the same network (local link), to verify the reachability of the neighbor, and to keep track of neighboring routers.

The switch supports ICMPv6 redirect for routes with mask lengths less than 64 bits. ICMP redirect is not supported for host routes or for summarized routes with mask lengths greater than 64 bits.

Neighbor discovery throttling ensures that the switch CPU is not unnecessarily burdened while it is in the process of obtaining the next hop forwarding information to route an IPv6 packet. The switch drops any additional IPv6 packets whose next hop is the same neighbor that the switch is actively trying to resolve. This drop avoids further load on the CPU.

Default Router Preference

The switch supports IPv6 default router preference (DRP), an extension in router advertisement messages. DRP improves the ability of a host to select an appropriate router, especially when the host is multihomed and the routers are on different links. The switch does not support the Route Information Option in RFC 4191.

An IPv6 host maintains a default router list from which it selects a router for traffic to offlink destinations. The selected router for a destination is then cached in the destination cache. NDP for IPv6 specifies that routers that are reachable or probably reachable are preferred over routers whose reachability is unknown or suspect. For reachable or probably reachable routers, NDP can either select the same router every time or cycle through the router list. By using DRP, you can configure an IPv6 host to prefer one router over another, provided both are reachable or probably reachable.

For more information about DRP for IPv6, see the "Implementing IPv6 Addresses and Basic Connectivity" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

IPv6 Stateless Autoconfiguration and Duplicate Address Detection

The switch uses stateless autoconfiguration to manage link, subnet, and site addressing changes, such as management of host and mobile IP addresses. A host autonomously configures its own link-local address, and booting nodes send router solicitations to request router advertisements for configuring interfaces.

For more information about autoconfiguration and duplicate address detection, see the “Implementing IPv6 Addressing and Basic Connectivity” chapter of *Cisco IOS IPv6 Configuration Library* on Cisco.com.

IPv6 Applications

The switch has IPv6 support for these applications:

- Ping, traceroute, Telnet, TFTP, and FTP
- Secure Shell (SSH) over an IPv6 transport
- HTTP server access over IPv6 transport
- DNS resolver for AAAA over IPv4 transport
- Cisco Discovery Protocol (CDP) support for IPv6 addresses

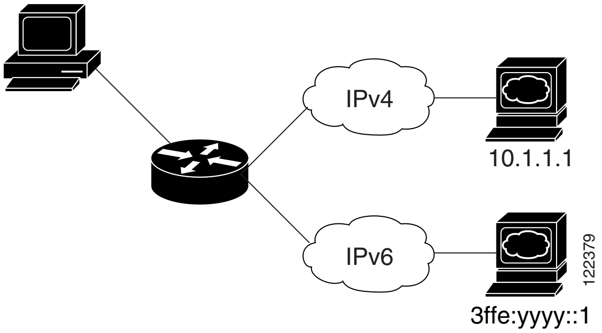
For more information about managing these applications, see the “Managing Cisco IOS Applications over IPv6” chapter and the “Implementing IPv6 Addressing and Basic Connectivity” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Dual IPv4 and IPv6 Protocol Stacks

You must use the dual IPv4 and IPv6 template to allocate ternary content addressable memory (TCAM) usage to both IPv4 and IPv6 protocols.

Figure 86 shows a router forwarding both IPv4 and IPv6 traffic through the same interface, based on the IP packet and destination addresses.

Figure 86 Dual IPv4 and IPv6 Support on an Interface



Use the dual IPv4 and IPv6 switch database management (SDM) template to enable dual-stack environments (supporting both IPv4 and IPv6).

The dual IPv4 and IPv6 templates allow the switch to be used in dual-stack environments.

- If you try to configure IPv6 without first selecting a dual IPv4 and IPv6 template, a warning message appears.
- In IPv4-only environments, the switch applies IPv4 QoS and ACLs in hardware. IPv6 packets are not supported.
- In dual IPv4 and IPv6 environments, the switch applies IPv4 QoS and ACLs in hardware.
- IPv6 QoS and ACLs are not supported.
- If you do not plan to use IPv6, do not use the dual-stack template because this template results in less TCAM capacity for each resource.

For more information about IPv4 and IPv6 protocol stacks, see the “Implementing IPv6 Addressing and Basic Connectivity” chapter of *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Static Routes for IPv6

Static routes are manually configured and define an explicit route between two networking devices. Static routes are useful for smaller networks with only one path to an outside network or to provide security for certain types of traffic in a larger network.

For more information about static routes, see the “Implementing Static Routes for IPv6” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

SNMP and Syslog Over IPv6

To support both IPv4 and IPv6, IPv6 network management requires both IPv6 and IPv4 transports. Syslog over IPv6 supports address data types for these transports.

SNMP and syslog over IPv6 provide these features:

- Support for both IPv4 and IPv6
- IPv6 transport for SNMP and to modify the SNMP agent to support traps for an IPv6 host
- SNMP- and syslog-related MIBs to support IPv6 addressing
- Configuration of IPv6 hosts as trap receivers

For support over IPv6, SNMP modifies the existing IP transport mapping to simultaneously support IPv4 and IPv6. These SNMP actions support IPv6 transport management:

- Opens User Datagram Protocol (UDP) SNMP socket with default settings
- Provides a new transport mechanism called *SR_IPV6_TRANSPORT*
- Sends SNMP notifications over IPv6 transport
- Supports SNMP-named access lists for IPv6 transport
- Supports SNMP proxy forwarding using IPv6 transport
- Verifies SNMP Manager feature works with IPv6 transport

For information on SNMP over IPv6, including configuration procedures, see the “Managing Cisco IOS Applications over IPv6” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

For information about syslog over IPv6, including configuration procedures, see the “Implementing IPv6 Addressing and Basic Connectivity” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

HTTP over IPv6

The HTTP client sends requests to both IPv4 and IPv6 HTTP servers, which respond to requests from both IPv4 and IPv6 HTTP clients. URLs with literal IPv6 addresses must be specified in hexadecimal using 16-bit values between colons.

The accept socket call chooses an IPv4 or IPv6 address family. The accept socket is either an IPv4 or IPv6 socket. The listening socket continues to listen for both IPv4 and IPv6 signals that indicate a connection. The IPv6 listening socket is bound to an IPv6 wildcard address.

The underlying TCP/IP stack supports a dual-stack environment. HTTP relies on the TCP/IP stack and the sockets for processing network-layer interactions.

Basic network connectivity (**ping**) must exist between the client and the server hosts before HTTP connections can be made.

Default IPv6 Settings

Feature	Default Setting
SDM template	Default.
IPv6 addresses	None configured.

How to Configure IPv6 Hosting

Configuring IPv6 Addressing and Enabling IPv6 Host

This section describes how to assign IPv6 addresses to individual Layer 3 interfaces and to globally forward IPv6 traffic on the switch.

Before configuring IPv6 on the switch, consider these guidelines:

- Be sure to select a dual IPv4 and IPv6 SDM template.
- In the **ipv6 address** interface configuration command, you must enter the *ipv6-address* and *ipv6-prefix* variables with the address specified in hexadecimal using 16-bit values between colons. The *prefix-length* variable (preceded by a slash [/]) is a decimal value that shows how many of the high-order contiguous bits of the address comprise the prefix (the network portion of the address).

To forward IPv6 traffic on an interface, you must configure a global IPv6 address on that interface. Configuring an IPv6 address on an interface automatically configures a link-local address and activates IPv6 for the interface. The configured interface automatically joins these required multicast groups for that link:

- solicited-node multicast group FF02:0:0:0:0:1:ff00::/104 for each unicast address assigned to the interface (this address is used in the neighbor discovery process.)
- all-nodes link-local multicast group FF02::1
- all-routers link-local multicast group FF02::2

For more information about configuring IPv6, see the “Implementing Addressing and Basic Connectivity for IPv6” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**sdm prefer dual-ipv4-and-ipv6 default**	Selects the SDM template that supports IPv4 and IPv6.
3.	**end**	Returns to privileged EXEC mode.
4.	**reload**	Reloads the operating system.
5.	**configure terminal**	Enters global configuration mode after the switch reloads.
6.	**interface** *interface-id*	Enters interface configuration mode, and specifies the interface to configure.

	Command	Purpose
7.	**ipv6 address** *ipv6-prefix/prefix length* **eui-64** or **ipv6 address** *ipv6-address* **link-local** or **ipv6 enable**	■ Specifies a global IPv6 address with an extended unique identifier (EUI) in the low-order 64 bits of the IPv6 address. ■ Specifies only the network prefix; the last 64 bits are automatically computed from the switch MAC address. This enables IPv6 processing on the interface. ■ Specifies a link-local address on the interface to be used instead of the link-local address that is automatically configured when IPv6 is enabled on the interface. This command enables IPv6 processing on the interface. ■ Automatically configures an IPv6 link-local address on the interface, and enable the interface for IPv6 processing. The link-local address can only be used to communicate with nodes on the same link.
8.	**exit**	Returns to global configuration mode.
9.	**end**	Returns to privileged EXEC mode.

Configuring Default Router Preference

Router advertisement messages are sent with the default router preference (DRP) configured by the **ipv6 nd router-preference** interface configuration command. If no DRP is configured, RAs are sent with a medium preference.

A DRP is useful when two routers on a link might provide equivalent, but not equal-cost routing, and policy might dictate that hosts should prefer one of the routers.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Enters interface configuration mode, and enters the Layer 3 interface on which you want to specify the DRP.
3.	**ipv6 nd router-preference** {**high** | **medium | low**}	Specifies a DRP for the router on the switch interface.
4.	**end**	Returns to privileged EXEC mode.

Configuring IPv6 ICMP Rate Limiting

ICMP rate limiting is enabled by default with a default interval between error messages of 100 milliseconds and a bucket size (maximum number of tokens to be stored in a bucket) of 10.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**ipv6 icmp error-interval** *interval* [*bucketsize*]	Configures the interval and bucket size for IPv6 ICMP error messages: ■ *interval*–The interval (in milliseconds) between tokens being added to the bucket. The range is from 0 to 2147483647 milliseconds. ■ *bucketsize*–(Optional) The maximum number of tokens stored in the bucket. The range is from 1 to 200.
3.	**end**	Returns to privileged EXEC mode.

Monitoring and Maintaining IPv6 Host Information

Command	Purpose
show ipv6 interface *interface-id*	Displays IPv6 interface status and configuration.
show ipv6 mtu	Displays IPv6 MTU per destination cache.
show ipv6 neighbors	Displays IPv6 neighbor cache entries.
show ipv6 prefix-list	Displays a list of IPv6 prefix lists.
show ipv6 protocols	Displays IPv6 routing protocols on the switch.
show ipv6 route	Displays the IPv6 route table entries.
show ipv6 static	Displays IPv6 static routes.
show ipv6 traffic	Displays IPv6 traffic statistics.
show ip http server history	Displays the previous 20 connections to the HTTP server, including the IP address accessed and the time when the connection was closed.
show ip http server connection	Displays the current connections to the HTTP server, including the local and remote IP addresses being accessed.
show ip http client connection	Displays the configuration values for HTTP client connections to HTTP servers.
show ip http client history	Displays a list of the last 20 requests made by the HTTP client to the server.

Configuration Examples for IPv6 Host Functions

Enabling IPv6: Example

This example shows how to enable IPv6 with both a link-local address and a global address based on the IPv6 prefix 2001:0DB8:c18:1::/64. The EUI-64 interface ID is used in the low-order 64 bits of both addresses. Output from the **show ipv6 interface** EXEC command shows how the interface ID (20B:46FF:FE2F:D940) is appended to the link-local prefix FE80::/64 of the interface.

```
Switch(config)# sdm prefer dual-ipv4-and-ipv6 default
Switch(config)# interface gigabitethernetfastethernet1/0/11
Switch(config-if)# ipv6 address 2001:0DB8:c18:1::/64 eui 64
Switch(config-if)# end
```

```
Switch# show ipv6 interface gigabitethernetfastethernet1/0/11
GigabitEthernetFastEthernet1/0/11 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::20B:46FF:FE2F:D940
  Global unicast address(es):
  2001:0DB8:c18:1:20B:46FF:FE2F:D940, subnet is 2001:0DB8:c18:1::/64 [EUI]
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::1:FF2F:D940
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds
  ND advertised reachable time is 0 milliseconds
  ND advertised retransmit interval is 0 milliseconds
  ND router advertisements are sent every 200 seconds
  ND router advertisements live for 1800 seconds
  Hosts use stateless autoconfig for addresses.
```

Configuring DRP: Example

This example shows how to configure a DRP of *high* for the router on an interface.

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# ipv6 nd router-preference high
Switch(config-if)# end
```

Configuring an IPv6 ICMP Error Message Interval

This example shows how to configure an IPv6 ICMP error message interval of 50 milliseconds and a bucket size of 20 tokens.

```
Switch(config)# ipv6 icmp error-interval 50 20
```

Displaying Show Command Output: Examples

This is an example of the output from the **show ipv6 interface** privileged EXEC command:

```
Switch# show ipv6 interface
Vlan1 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::20B:46FF:FE2F:D940
  Global unicast address(es):
    3FFE:C000:0:1:20B:46FF:FE2F:D940, subnet is 3FFE:C000:0:1::/64 [EUI]
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::1:FF2F:D940
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds
  ND advertised reachable time is 0 milliseconds
  ND advertised retransmit interval is 0 milliseconds
  ND router advertisements are sent every 200 seconds
  ND router advertisements live for 1800 seconds
<output truncated>
```

This is an example of the output from the **show ipv6 protocols** privileged EXEC command:

```
Switch# show ipv6 protocols
IPv6 Routing Protocol is “connected”
IPv6 Routing Protocol is “static”
IPv6 Routing Protocol is “rip fer”
  Interfaces:
    Vlan6
    FastEthernet0/4
    FastEthernet0/11
    FastEthernet0/12
    GigabitEthernet2/0/4
    GigabitEthernet2/0/
    GigabitEthernet1/0/12
Redistribution:
    None
```

This is an example of the output from the **show ipv6 neighbor** privileged EXEC command:

```
Switch# show ipv6 neighbors
IPv6 Address                              Age Link-layer Addr State Interface
3FFE:C000:0:7::777                          - 0007.0007.0007  REACH Vl7
3FFE:C101:113:1::33                         - 0000.0000.0033  REACH Fa1/0/13
```

This is an example of the output from the **show ipv6 route** privileged EXEC command:

```
Switch# show ipv6 route
IPv6 Routing Table - Default - 1 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
L   FF00::/8 [0/0]
     via Null0, receive
```

This is an example of the output from the **show ipv6 traffic** privileged EXEC command.

```
Switch# show ipv6 traffic
IPv6 statistics:
  Rcvd:  1 total, 1 local destination
         0 source-routed, 0 truncated
         0 format errors, 0 hop count exceeded
         0 bad header, 0 unknown option, 0 bad source
         0 unknown protocol, 0 not a router
         0 fragments, 0 total reassembled
         0 reassembly timeouts, 0 reassembly failures
  Sent:  36861 generated, 0 forwarded
         0 fragmented into 0 fragments, 0 failed
         0 encapsulation failed, 0 no route, 0 too big
         0 RPF drops, 0 RPF suppressed drops
  Mcast: 1 received, 36861 sent

ICMP statistics:
  Rcvd: 1 input, 0 checksum errors, 0 too short
        0 unknown info type, 0 unknown error type
        unreach: 0 routing, 0 admin, 0 neighbor, 0 address, 0 port
        parameter: 0 error, 0 header, 0 option
        0 hopcount expired, 0 reassembly timeout,0 too big
        0 echo request, 0 echo reply
        0 group query, 0 group report, 0 group reduce
        1 router solicit, 0 router advert, 0 redirects
        0 neighbor solicit, 0 neighbor advert
  Sent: 10112 output, 0 rate-limited
        unreach: 0 routing, 0 admin, 0 neighbor, 0 address, 0 port
        parameter: 0 error, 0 header, 0 option
        0 hopcount expired, 0 reassembly timeout,0 too big
        0 echo request, 0 echo reply
        0 group query, 0 group report, 0 group reduce
        0 router solicit, 9944 router advert, 0 redirects
        84 neighbor solicit, 84 neighbor advert

UDP statistics:
  Rcvd: 0 input, 0 checksum errors, 0 length errors
        0 no port, 0 dropped
  Sent: 26749 output

TCP statistics:
  Rcvd: 0 input, 0 checksum errors
  Sent: 0 output, 0 retransmitted
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*
Cisco IOS static IPv6 routing	"Implementing Static Routes for IPv6" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.
DRP for IPv6	"Implementing IPv6 Addresses and Basic Connectivity" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Configuring Link State Tracking

Restrictions for Configuring Link State Tracking

- To use this feature, the switch must be running the LAN Base image.
- An interface that is defined as an upstream interface cannot also be defined as a downstream interface in the same or a different link state group. The reverse is also true.
- An interface cannot be a member of more than one link state group.
- You can configure only two link state groups per switch.

Information About Configuring Link State Tracking

Link State Tracking

Link state tracking, also known as trunk failover, is a feature that binds the link state of multiple interfaces. For example, link state tracking provides redundancy in the network when used with server NIC adapter teaming. When the server network adapters are configured in a primary or secondary relationship known as teaming, if the link is lost on the primary interface, connectivity is transparently changed to the secondary interface.

Note: An interface can be an aggregation of ports (an EtherChannel), a single physical port in access or trunk mode, or a routed port.

Figure 87 on page 713 shows a network configured with link state tracking. To enable link state tracking, create a *link state group*, and specify the interfaces that are assigned to the link state group. In a link state group, these interfaces are bundled together. The *downstream interfaces* are bound to the *upstream interfaces*. Interfaces connected to servers are referred to as downstream interfaces, and interfaces connected to distribution switches and network devices are referred to as upstream interfaces.

The configuration in Figure 87 on page 713 ensures that the network traffic flow is balanced as follows:

- For links to switches and other network devices
  - Server 1 and server 2 use switch A for primary links and switch B for secondary links.
  - Server 3 and server 4 use switch B for primary links and switch A for secondary links.
- Link state group 1 on switch A
  - Switch A provides primary links to server 1 and server 2 through link state group 1. Port 1 is connected to server 1, and port 2 is connected to server 2. Port 1 and port 2 are the downstream interfaces in link state group 1.
  - Port 5 and port 6 are connected to distribution switch 1 through link state group 1. Port 5 and port 6 are the upstream interfaces in link state group 1.

- Link state group 2 on switch A
  - Switch A provides secondary links to server 3 and server 4 through link state group 2. Port 3 is connected to server 3, and port 4 is connected to server 4. Port 3 and port 4 are the downstream interfaces in link state group 2.
  - Port 7 and port 8 are connected to distribution switch 2 through link state group 2. Port 7 and port 8 are the upstream interfaces in link state group 2.
- Link state group 2 on switch B
  - Switch B provides primary links to server 3 and server 4 through link state group 2. Port 3 is connected to server 3, and port 4 is connected to server 4. Port 3 and port 4 are the downstream interfaces in link state group 2.
  - Port 5 and port 6 are connected to distribution switch 2 through link state group 2. Port 5 and port 6 are the upstream interfaces in link state group 2.
- Link state group 1 on switch B
  - Switch B provides secondary links to server 1 and server 2 through link state group 1. Port 1 is connected to server 1, and port 2 is connected to server 2. Port 1 and port 2 are the downstream interfaces in link state group 1.
  - Port 7 and port 8 are connected to distribution switch 1 through link state group 1. Port 7 and port 8 are the upstream interfaces in link state group 1.

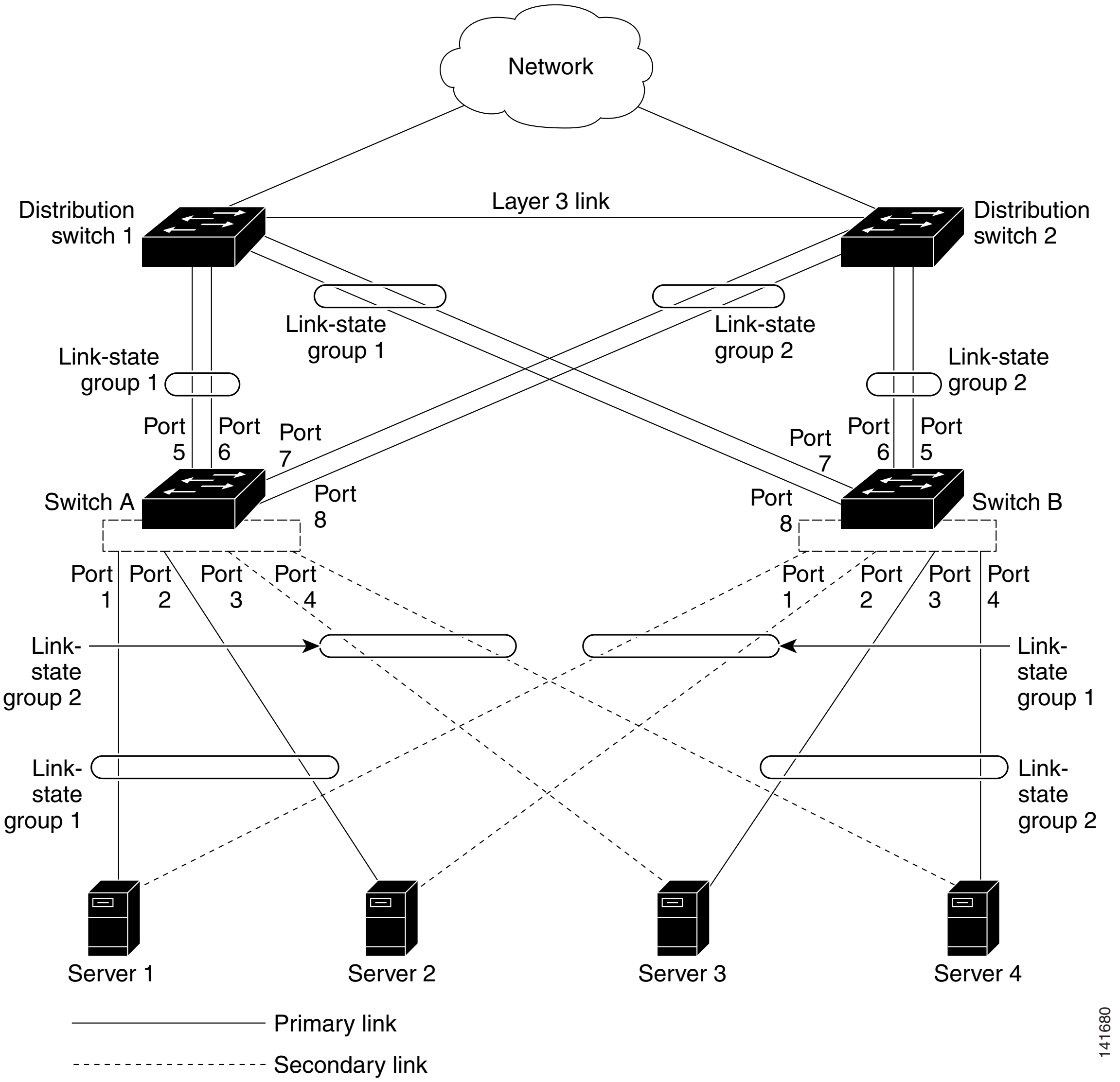
In a link state group, the upstream ports can become unavailable or lose connectivity because the distribution switch or router fails, the cables are disconnected, or the link is lost. These are the interactions between the downstream and upstream interfaces when link state tracking is enabled:

- If any of the upstream interfaces are in the link-up state, the downstream interfaces can change to or remain in the link-up state.
- If all of the upstream interfaces become unavailable, link state tracking automatically puts the downstream interfaces in the error-disabled state. Connectivity to and from the servers is automatically changed from the primary server interface to the secondary server interface.

  As an example of a connectivity change from link state group 1 to link state group 2 on switch A, see Figure 87 on page 713. If the upstream link for port 6 is lost, the link states of downstream ports 1 and 2 do not change. However, if the link for upstream port 5 is also lost, the link state of the downstream ports changes to the link-down state. Connectivity to server 1 and server 2 is then changed from link state group1 to link state group 2. The downstream ports 3 and 4 do not change state because they are in link-group 2.
- If the link state group is configured, link state tracking is disabled, and the upstream interfaces lose connectivity, the link states of the downstream interfaces remain unchanged. The server does not recognize that upstream connectivity has been lost and does not failover to the secondary interface.

You can recover a downstream interface link-down condition by removing the failed downstream port from the link state group. To recover multiple downstream interfaces, disable the link state group.

Figure 87 Typical Link State Tracking Configuration



Default Link State Tracking Configuration

There are no link state groups defined, and link state tracking is not enabled for any group.

How to Configure Link State Tracking

Configuring Link State Tracking

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**link state track** *number*	Creates a link state group, and enables link state tracking. The group number can be 1 to 2; the default is 1.
3.	**interface** *interface-id*	Specifies a physical interface or range of interfaces to configure, and enters interface configuration mode. Valid interfaces include switch ports in access or trunk mode (IEEE 802.1q), routed ports, or multiple ports bundled into an EtherChannel interface (static or LACP), also in trunk mode.
4.	**link state group** [*number*] {**upstream** | **downstream**}	Specifies a link state group, and configures the interface as either an **upstream** or **downstream** interface in the group.The group number can be 1 to 2; the default is 1.
5.	**end**	Returns to privileged EXEC mode.

Monitoring and Maintaining Link State Tracking

Command	Purpose
show link state group	Displays the link state group information.

Configuration Examples for Configuring Link State Tracking

Displaying Link State Information: Examples

Use the **show link state group** command to display the link state group information. Enter this command without keywords to display information about all link state groups. Enter the group number to display information specific to the group. Enter the detail keyword to display detailed information about the group.

This is an example of output from the **show link state group 1** command:

```
Switch> show link state group 1

Link State Group: 1      Status: Enabled, Down
```

This is an example of output from the **show link state group detail** command:

```
Switch> show link state group detail

(Up):Interface up   (Dwn):Interface Down   (Dis):Interface disabled

Link State Group: 1 Status: Enabled, Down
Upstream Interfaces : Fa1/7(Dwn) Fa1/8(Dwn)
Downstream Interfaces : Fa1/3(Dis) Fa1/4(Dis) Fa1/5(Dis) Fa1/6(Dis)

Link State Group: 2 Status: Enabled, Down
Upstream Interfaces : Fa1/6(Dwn) Fa1/7(Dwn) Fa1/8(Dwn)
Downstream Interfaces : Fa1/2(Dis) Fa1/3(Dis) Fa1/4(Dis) Fa1/5(Dis)
```

```
(Up):Interface up (Dwn):Interface Down (Dis):Interface disabled
```

Creating a Link State Group: Example

This example shows how to create a link state group and configure the interfaces:

```
Switch# configure terminal
Switch(config)# link state track 1
Switch(config)# interface range GigabitEthernet1/17 -2
Switch(config-if)# link state group 1 upstream
Switch(config-if)# interface GigabitEthernet1/17
Switch(config-if)# link state group 1 downstream
Switch(config-if)# interface GigabitEthernet1/17
Switch(config-if)# link state group 1 downstream
Switch(config-if)# interface GigabitEthernet1/18
Switch(config-if)# link state group 1 downstream
Switch(config-if)# end
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*
EtherChannel configuration	Configuring EtherChannels, page 1069

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Configuring IP Multicast Routing

This chapter describes how to configure IP multicast routing on the Cisco Industrial Ethernet switch, hereafter referred to as *switch*. IP multicasting is a more efficient way to use network resources, especially for bandwidth-intensive services such as audio and video. IP multicast routing enables a host (source) to send packets to a group of hosts (receivers) anywhere within the IP network by using a special form of IP address called the IP *multicast group address.* The sending host inserts the multicast group address into the IP destination address field of the packet, and IP multicast routers and multilayer switches forward incoming IP multicast packets out all interfaces that lead to members of the multicast group. Any host, regardless of whether it is a member of a group, can send to a group. However, only the members of a group receive the message.

Note: For complete syntax and usage information for the commands used in this chapter, see the documents listed in the Related Documents, page 772.

This chapter includes the following sections:

Information About Cisco’s Implementation of IP Multicast Routing

The switch supports these protocols to implement IP multicast routing:

- Internet Group Management Protocol (IGMP) is used among hosts on a LAN and the routers (and multilayer switches) on that LAN to track the multicast groups of which hosts are members.
- Protocol-Independent Multicast (PIM) protocol is used among routers and multilayer switches to track which multicast packets to forward to each other and to their directly connected LANs.

According to IPv4 multicast standards, the MAC destination multicast address begins with 0100:5e and is appended by the last 23 bits of the IP address. On the switch, if the multicast packet does not match the switch multicast address, the packets are treated in this way:

- If the packet has a multicast IP address and a unicast MAC address, the packet is forwarded in software. This can occur because some protocols on legacy devices use unicast MAC addresses with multicast IP addresses.
- If the packet has a multicast IP address and an unmatched multicast MAC address, the packet is dropped.

This section includes the following topics:

Information About IGMP

To participate in IP multicasting, multicast hosts, routers, and multilayer switches must have the IGMP operating. This protocol defines the querier and host roles:

- A querier is a network device that sends query messages to discover which network devices are members of a given multicast group.
- A host is a receiver that sends report messages (in response to query messages) to inform a querier of a host membership.

A set of queriers and hosts that receive multicast data streams from the same source is called a multicast group. Queriers and hosts use IGMP messages to join and leave multicast groups.

Any host, regardless of whether it is a member of a group, can send to a group. However, only the members of a group receive the message. Membership in a multicast group is dynamic; hosts can join and leave at any time. There is no restriction on the location or number of members in a multicast group. A host can be a member of more than one multicast group at a time. How active a multicast group is and what members it has can vary from group to group and from time to time. A multicast group can be active for a long time, or it can be very short-lived. Membership in a group can constantly change. A group that has members can have no activity.

IP multicast traffic uses group addresses, which are class D addresses. The high-order bits of a Class D address are 1110. Therefore, host group addresses can be in the range 224.0.0.0 through 239.255.255.255. Multicast addresses in the range 224.0.0.0 to 24.0.0.255 are reserved for use by routing protocols and other network control traffic. The address 224.0.0.0 is guaranteed not to be assigned to any group.

IGMP packets are sent using these IP multicast group addresses:

- IGMP general queries are destined to the address 224.0.0.1 (all systems on a subnet).
- IGMP group-specific queries are destined to the group IP address for which the switch is querying.
- IGMP group membership reports are destined to the group IP address for which the switch is reporting.
- IGMP Version 2 (IGMPv2) leave messages are destined to the address 224.0.0.2 (all-multicast-routers on a subnet). In some old host IP stacks, leave messages might be destined to the group IP address rather than to the all-routers address.

IGMP Version 1

IGMP Version 1 (IGMPv1) primarily uses a query-response model that enables the multicast router and multilayer switch to find which multicast groups are active (have one or more hosts interested in a multicast group) on the local subnet. IGMPv1 has other processes that enable a host to join and leave a multicast group. For more information, see RFC 1112.

IGMP Version 2

IGMPv2 extends IGMP functionality by providing such features as the IGMP leave process to reduce leave latency, group-specific queries, and an explicit maximum query response time. IGMPv2 also adds the capability for routers to elect the IGMP querier without depending on the multicast protocol to perform this task. For more information, see RFC 2236.

Information About PIM

PIM is called *protocol-independent*: regardless of the unicast routing protocols used to populate the unicast routing table, PIM uses this information to perform multicast forwarding instead of maintaining a separate multicast routing table.

PIM is defined in RFC 2362, *Protocol-Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification*. PIM is defined in these Internet Engineering Task Force (IETF) Internet drafts:

- *Protocol Independent Multicast (PIM): Motivation and Architecture*
- *Protocol Independent Multicast (PIM), Dense Mode Protocol Specification*
- *Protocol Independent Multicast (PIM), Sparse Mode Protocol Specification*
- *draft-ietf-idmr-igmp-v2-06.txt, Internet Group Management Protocol, Version 2*
- *draft-ietf-pim-v2-dm-03.txt, PIM Version 2 Dense Mode*

This section includes the following topics:

PIM Versions

PIMv2 includes these improvements over PIMv1:

- A single, active rendezvous point (RP) exists per multicast group, with multiple backup RPs. This single RP compares to multiple active RPs for the same group in PIMv1.
- A bootstrap router (BSR) provides a fault-tolerant, automated RP discovery and distribution mechanism that enables routers and multilayer switches to dynamically learn the group-to-RP mappings.
- Sparse mode and dense mode are properties of a group, as opposed to an interface. We strongly recommend sparse-dense mode, as opposed to either sparse mode or dense mode only.
- PIM join and prune messages have more flexible encoding for multiple address families.
- A more flexible hello packet format replaces the query packet to encode current and future capability options.
- Register messages to an RP specify whether they are sent by a border router or a designated router.

- PIM packets are no longer inside IGMP packets; they are standalone packets.

PIM Modes

PIM can operate in dense mode (DM), sparse mode (SM), or in sparse-dense mode (PIM DM-SM), which handles both sparse groups and dense groups at the same time.

PIM DM

PIM DM builds source-based multicast distribution trees. In dense mode, a PIM DM router or multilayer switch assumes that all other routers or multilayer switches forward multicast packets for a group. If a PIM DM device receives a multicast packet and has no directly connected members or PIM neighbors present, a prune message is sent back to the source to stop unwanted multicast traffic. Subsequent multicast packets are not flooded to this router or switch on this pruned branch because branches without receivers are pruned from the distribution tree, leaving only branches that contain receivers.

When a new receiver on a previously pruned branch of the tree joins a multicast group, the PIM DM device detects the new receiver and immediately sends a graft message up the distribution tree toward the source. When the upstream PIM DM device receives the graft message, it immediately puts the interface on which the graft was received into the forwarding state so that the multicast traffic begins flowing to the receiver.

PIM SM

PIM SM uses shared trees and shortest-path-trees (SPTs) to distribute multicast traffic to multicast receivers in the network. In PIM SM, a router or multilayer switch assumes that other routers or switches do not forward multicast packets for a group, unless there is an explicit request for the traffic (join message). When a host joins a multicast group using IGMP, its directly connected PIM SM device sends PIM join messages toward the root, also known as the RP. This join message travels router-by-router toward the root, constructing a branch of the shared tree as it goes.

The RP keeps track of multicast receivers. It also registers sources through register messages received from the source’s first-hop router (*designated router* [DR]) to complete the shared tree path from the source to the receiver. When using a shared tree, sources must send their traffic to the RP so that the traffic reaches all receivers.

Prune messages are sent up the distribution tree to prune multicast group traffic. This action permits branches of the shared tree or SPT that were created with explicit join messages to be torn down when they are no longer needed.

PIM Stub Routing

The PIM stub routing feature reduces resource usage by moving routed traffic closer to the end user.

In a network using PIM stub routing, the only allowable route for IP traffic to the user is through a switch that is configured with PIM stub routing. PIM passive interfaces are connected to Layer 2 access domains, such as VLANs, or to interfaces that are connected to other Layer 2 devices. Only directly connected multicast (IGMP) receivers and sources are allowed in the Layer 2 access domains. The PIM passive interfaces do not send or process any received PIM control packets.

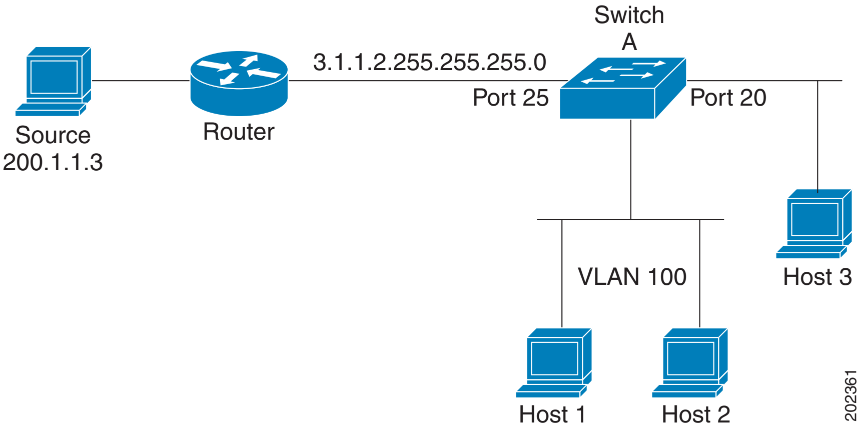
When using PIM stub routing, you should configure the distribution and remote routers to use IP multicast routing and configure only the switch as a PIM stub router. The switch does not route transit traffic between distribution routers. You also need to configure a routed uplink port on the switch. The switch uplink port cannot be used with SVIs. If you need PIM for an SVI uplink port, you should upgrade to the IP services feature set.

You must also configure EIGRP stub routing when configuring PIM stub routing on the switch.

The redundant PIM stub router topology is not supported. The redundant topology exists when there is more than one PIM router forwarding multicast traffic to a single access domain. PIM messages are blocked, and the PIM assert and designated router election mechanisms are not supported on the PIM passive interfaces. Only the nonredundant access router topology is supported by the PIM stub feature. By using a nonredundant topology, the PIM passive interface assumes that it is the only interface and designated router on that access domain.

In Figure 88 on page 721, Switch A routed uplink port 25 is connected to the router and PIM stub routing is enabled on the VLAN 100 interfaces and on Host 3. This configuration allows the directly connected hosts to receive traffic from multicast source 200.1.1.3. See Configuring PIM Stub Routing, page 733 for more information.

Figure 88 PIM Stub Router Configuration



IGMP Helper

PIM stub routing moves routed traffic closer to the end user and reduces network traffic. You can also reduce traffic by configuring a stub router (switch) with the IGMP helper feature.

You can configure a stub router (switch) with the **igmp helper help-address** interface configuration command to enable the switch to send reports to the next-hop interface. Hosts that are not directly connected to a downstream router can then join a multicast group sourced from an upstream network. The IGMP packets from a host wanting to join a multicast stream are forwarded upstream to the next-hop device when this feature is configured. When the upstream central router receives the helper IGMP reports or leaves, it adds or removes the interfaces from its outgoing interface list for that group.

For complete syntax and usage information for the **ip igmp helper-address** command, see the *Cisco IOS IP Multicast Command Reference*.

Auto-RP

This proprietary feature eliminates the need to manually configure the RP information in every router and multilayer switch in the network. For Auto-RP to work, you configure a Cisco router or multilayer switch as the mapping agent. It uses IP multicast to learn which routers or switches in the network are possible candidate RPs to receive candidate RP announcements. Candidate RPs periodically send multicast RP-announce messages to a particular group or group range to announce their availability.

Mapping agents listen to these candidate RP announcements and use the information to create entries in their Group-to-RP mapping caches. Only one mapping cache entry is created for any Group-to-RP range received, even if multiple candidate RPs are sending RP announcements for the same range. As the RP-announce messages arrive, the mapping agent selects the router or switch with the highest IP address as the active RP and stores this RP address in the Group-to-RP mapping cache.

Mapping agents periodically multicast the contents of their Group-to-RP mapping cache. Thus, all routers and switches automatically discover which RP to use for the groups they support. If a router or switch fails to receive RP-discovery messages and the Group-to-RP mapping information expires, it switches to a statically configured RP that was defined with the **ip pim rp-address** global configuration command. If no statically configured RP exists, the router or switch changes the group to dense-mode operation.

Multiple RPs serve different group ranges or serve as hot backups of each other.

Bootstrap Router

PIMv2 BSR is another method to distribute group-to-RP mapping information to all PIM routers and multilayer switches in the network. It eliminates the need to manually configure RP information in every router and switch in the network. However, instead of using IP multicast to distribute group-to-RP mapping information, BSR uses hop-by-hop flooding of special BSR messages to distribute the mapping information.

The BSR is elected from a set of candidate routers and switches in the domain that have been configured to function as BSRs. The election mechanism is similar to the root-bridge election mechanism used in bridged LANs. The BSR election is based on the BSR priority of the device contained in the BSR messages that are sent hop-by-hop through the network. Each BSR device examines the message and forwards out all interfaces only the message that has either a higher BSR priority than its BSR priority or the same BSR priority, but with a higher BSR IP address. Using this method, the BSR is elected.

The elected BSR sends BSR messages with a TTL of 1. Neighboring PIMv2 routers or multilayer switches receive the BSR message and multicast it out all other interfaces (except the one on which it was received) with a TTL of 1. In this way, BSR messages travel hop-by-hop throughout the PIM domain. Because BSR messages contain the IP address of the current BSR, the flooding mechanism enables candidate RPs to automatically learn which device is the elected BSR.

Candidate RPs send candidate RP advertisements showing the group range for which they are responsible to the BSR, which stores this information in its local candidate-RP cache. The BSR periodically advertises the contents of this cache in BSR messages to all other PIM devices in the domain. These messages travel hop-by-hop through the network to all routers and switches, which store the RP information in the BSR message in their local RP cache. The routers and switches select the same RP for a given group because they all use a common RP hashing algorithm.

Multicast Forwarding and Reverse Path Check

With unicast routing, routers and multilayer switches forward traffic through the network along a single path from the source to the destination host whose IP address appears in the destination address field of the IP packet. Each router and switch along the way makes a unicast forwarding decision, using the destination IP address in the packet, by looking up the destination address in the unicast routing table and forwarding the packet through the specified interface to the next hop toward the destination.

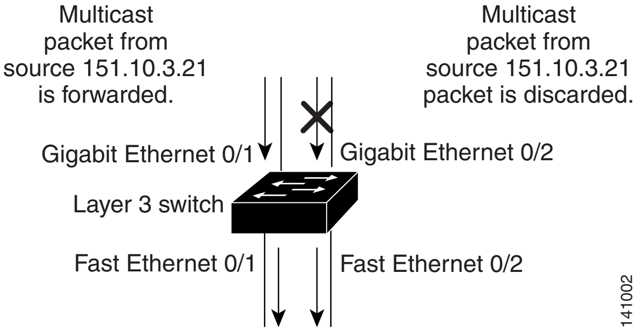
With multicasting, the source is sending traffic to an arbitrary group of hosts represented by a multicast group address in the destination address field of the IP packet. To decide whether to forward or drop an incoming multicast packet, the router or multilayer switch uses a reverse path forwarding (RPF) check on the packet as follows and shown in Figure 89 on page 723:

1. The router or multilayer switch examines the source address of the arriving multicast packet to decide whether the packet arrived on an interface that is on the reverse path back to the source.
2. If the packet arrives on the interface leading back to the source, the RPF check is successful and the packet is forwarded to all interfaces in the outgoing interface list (which might not be all interfaces on the router).
3. If the RPF check fails, the packet is discarded.

Some multicast routing protocols maintain a separate multicast routing table and use it for the RPF check. However, PIM uses the unicast routing table to perform the RPF check.

Figure 89 on page 723 shows port 2 receiving a multicast packet from source 151.10.3.21. Table 1 shows that the port on the reverse path to the source is port 1, not port 2. Because the RPF check fails, the multilayer switch discards the packet. Another multicast packet from source 151.10.3.21 is received on port 1, and the routing table shows this port is on the reverse path to the source. Because the RPF check passes, the switch forwards the packet to all ports in the outgoing port list.

Figure 89 RPF Check


Multicast packet from source 151.10.3.21 is forwarded.
Multicast packet from source 151.10.3.21 packet is discarded.
Gigabit Ethernet 0/1
Gigabit Ethernet 0/2
Layer 3 switch
Fast Ethernet 0/1
Fast Ethernet 0/2
141002

Network	Port
151.10.0.0/16	Gigabit Ethernet 0/1
198.14.32.0/32	Fast Ethernet 0/1
204.1.16.0/24	Fast Ethernet 0/2

PIM uses both source trees and RP-rooted shared trees to forward datagrams (described in the PIM DM, page 720 and the PIM SM, page 720). The RPF check is performed differently for each:

- If a PIM router or multilayer switch has a source-tree state (that is, an (S,G) entry is present in the multicast routing table), it performs the RPF check against the IP address of the source of the multicast packet.
- If a PIM router or multilayer switch has a shared-tree state (and no explicit source-tree state), it performs the RPF check on the RP address (which is known when members join the group).

Sparse-mode PIM uses the RPF lookup function to decide where it needs to send joins and prunes:

- (S,G) joins (which are source-tree states) are sent toward the source.
- (*,G) joins (which are shared-tree states) are sent toward the RP.

Dense-mode PIM uses only source trees and uses RPF as previously described.

Information About Source-Specific Multicast

The Source-Specific Multicast (SSM) feature is an extension of IP multicast in which datagram traffic is forwarded to receivers from only those multicast sources that the receivers have explicitly joined. For multicast groups configured for SSM, only SSM distribution trees (no shared trees) are created.

SSM Components Overview

SSM is a datagram delivery model that best supports one-to-many applications, also known as broadcast applications. SSM is a core networking technology for the Cisco implementation of IP multicast solutions targeted for audio and video broadcast application environments. The switch supports these components that support the implementation of SSM:

- Protocol independent multicast source-specific mode (PIM-SSM)

  PIM-SSM is the routing protocol that supports the implementation of SSM and is derived from PIM sparse mode (PIM-SM).

- Internet Group Management Protocol version 3 (IGMPv3)

To run SSM with IGMPv3, SSM must be supported in the Cisco IOS router, the host where the application is running, and the application itself.

How SSM Differs from Internet Standard Multicast

The current IP multicast infrastructure in the Internet and many enterprise intranets is based on the PIM-SM protocol and Multicast Source Discovery Protocol (MSDP). These protocols have the limitations of the Internet Standard Multicast (ISM) service model. For example, with ISM, the network must maintain knowledge about which hosts in the network are actively sending multicast traffic.

The ISM service consists of the delivery of IP datagrams from any source to a group of receivers called the multicast host group. The datagram traffic for the multicast host group consists of datagrams with an arbitrary IP unicast source address S and the multicast group address G as the IP destination address. Systems receive this traffic by becoming members of the host group.

Membership in a host group simply requires signalling the host group through IGMP version 1, 2, or 3. In SSM, delivery of datagrams is based on (S, G) channels. In both SSM and ISM, no signalling is required to become a source. However, in SSM, receivers must subscribe or unsubscribe to (S, G) channels to receive or not receive traffic from specific sources. In other words, receivers can receive traffic only from (S, G) channels to which they are subscribed, whereas in ISM, receivers need not know the IP addresses of sources from which they receive their traffic. The proposed standard approach for channel subscription signalling use IGMP include mode membership reports, which are supported only in IGMP version 3.

SSM IP Address Range

SSM can coexist with the ISM service by applying the SSM delivery model to a configured subset of the IP multicast group address range. Cisco IOS software allows SSM configuration for the IP multicast address range of 224.0.0.0 through 239.255.255.255. When an SSM range is defined, existing IP multicast receiver applications do not receive any traffic when they try to use an address in the SSM range (unless the application is modified to use an explicit (S, G) channel subscription).

SSM Operations

An established network, in which IP multicast service is based on PIM-SM, can support SSM services. SSM can also be deployed alone in a network without the full range of protocols that are required for interdomain PIM-SM (for example, MSDP, Auto-RP, or bootstrap router [BSR]) if only SSM service is needed.

If SSM is deployed in a network already configured for PIM-SM, only the last-hop routers support SSM. Routers that are not directly connected to receivers do not require support for SSM. In general, these not-last-hop routers must only run PIM-SM in the SSM range and might need additional access control configuration to suppress MSDP signalling, registering, or PIM-SM shared tree operations from occurring within the SSM range.

Use the **ip pim ssm** global configuration command to configure the SSM range and to enable SSM. This configuration has the following effects:

- For groups within the SSM range, (S, G) channel subscriptions are accepted through IGMPv3 include-mode membership reports.
- PIM operations within the SSM range of addresses change to PIM-SSM, a mode derived from PIM-SM. In this mode, only PIM (S, G) join and prune messages are generated by the router, and no (S, G) rendezvous point tree (RPT) or (*, G) RPT messages are generated. Incoming messages related to RPT operations are ignored or rejected, and incoming PIM register messages are immediately answered with register-stop messages. PIM-SSM is backward-compatible with PIM-SM unless a router is a last-hop router. Therefore, routers that are not last-hop routers can run PIM-SM for SSM groups (for example, if they do not yet support SSM).
- No MSDP source-active (SA) messages within the SSM range are accepted, generated, or forwarded.

IGMPv3 Host Signalling

In IGMPv3, hosts signal membership to last hop routers of multicast groups. Hosts can signal group membership with filtering capabilities with respect to sources. A host can either signal that it wants to receive traffic from all sources sending to a group except for some specific sources (called exclude mode), or that it wants to receive traffic only from some specific sources sending to the group (called include mode).

IGMPv3 can operate with both ISM and SSM. In ISM, both exclude and include mode reports are applicable. In SSM, only include mode reports are accepted by the last-hop router. Exclude mode reports are ignored.

Information About Source Specific Multicast Mapping

The Source Specific Multicast (SSM) mapping feature supports SSM transition when supporting SSM on the end system is impossible or unwanted due to administrative or technical reasons. You can use SSM mapping to leverage SSM for video delivery to legacy STBs that do not support IGMPv3 or for applications that do not use the IGMPv3 host stack.

In a typical STB deployment, each TV channel uses one separate IP multicast group and has one active server host sending the TV channel. A single server can send multiple TV channels, but each to a different group. In this network environment, if a router receives an IGMPv1 or IGMPv2 membership report for a particular group, the report addresses the well-known TV server for the TV channel associated with the multicast group.

When SSM mapping is configured, if a router receives an IGMPv1 or IGMPv2 membership report for a particular group, the router translates this report into one or more channel memberships for the well-known sources associated with this group.

When the router receives an IGMPv1 or IGMPv2 membership report for a group, the router uses SSM mapping to determine one or more source IP addresses for the group. SSM mapping then translates the membership report as an IGMPv3 report and continues as if it had received an IGMPv3 report. The router then sends PIM joins and continues to be joined to these groups as long as it continues to receive the IGMPv1 or IGMPv2 membership reports, and the SSM mapping for the group remains the same.

SSM mapping enables the last hop router to determine the source addresses either by a statically configured table on the router or through a DNS server. When the statically configured table or the DNS mapping changes, the router leaves the current sources associated with the joined groups.

Static SSM Mapping

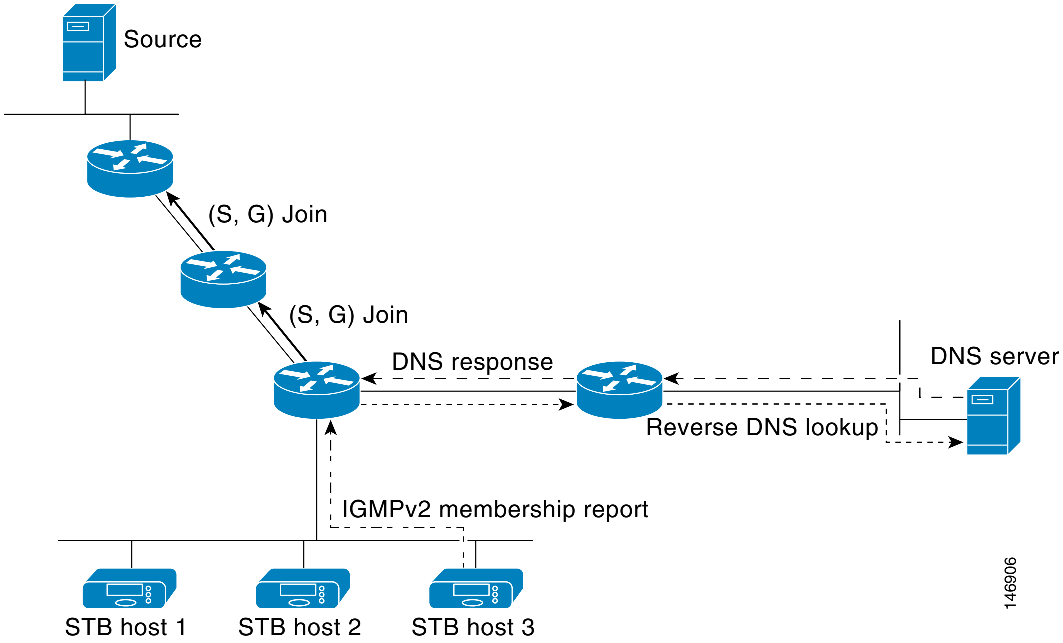
With static SSM mapping, you can configure the last hop router to use a static map to determine the sources that are sending to groups. Static SSM mapping requires that you configure ACLs to define group ranges. Then you can map the groups permitted by those ACLs to sources by using the **ip igmp static ssm-map** global configuration command.

You can configure static SSM mapping in smaller networks when a DNS is not needed or to locally override DNS mappings. When configured, static SSM mappings take precedence over DNS mappings.

DNS-Based SSM Mapping

You can use DNS-based SSM mapping to configure the last hop router to perform a reverse DNS lookup to determine sources sending to groups. When DNS-based SSM mapping is configured, the router constructs a domain name that includes the group address and performs a reverse lookup into the DNS. The router looks up IP address resource records and uses them as the source addresses associated with this group. SSM mapping supports up to 20 sources for each group. The router joins all sources configured for a group (see Figure 90 on page 726).

Figure 90 DNS-Based SSM-Mapping



The SSM mapping mechanism that enables the last hop router to join multiple sources for a group can provide source redundancy for a TV broadcast. In this context, the last hop router provides redundancy using SSM mapping to simultaneously join two video sources for the same TV channel. However, to prevent the last hop router from duplicating the video traffic, the video sources must use a server-side switchover mechanism. One video source is active, and the other backup video source is passive. The passive source waits until an active source failure is detected before sending the video traffic for the TV channel. Thus, the server-side switchover mechanism ensures that only one of the servers is actively sending video traffic for the TV channel.

To look up one or more source addresses for a group that includes G1, G2, G3, and G4, you must configure these DNS records on the DNS server:

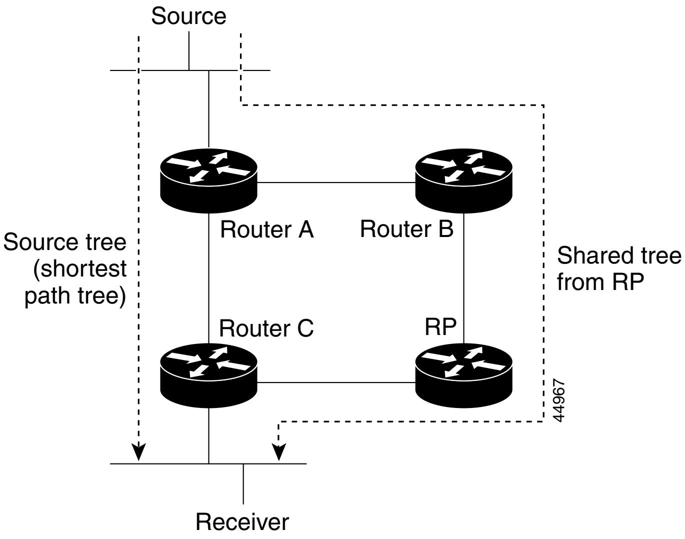
```
G4.G3.G2.G1 [multicast-domain] [timeout]IN A source-address-1
    IN A source-address-2
    IN A source-address-n
```

Refer to your DNS server documentation for more information about configuring DNS resource records.

Information About PIM Shared Tree and Source Tree

By default, members of a group receive data from senders to the group across a single data-distribution tree rooted at the RP. Figure 91 on page 727 shows this type of shared-distribution tree. Data from senders is delivered to the RP for distribution to group members joined to the shared tree.

Figure 91 Shared Tree and Source Tree (Shortest-Path Tree)



If the data rate warrants, leaf routers (routers without any downstream connections) on the shared tree can use the data distribution tree rooted at the source. This type of distribution tree is called a shortest-path tree or source tree. By default, the software switches to a source tree upon receiving the first data packet from a source.

This process describes the move from a shared tree to a source tree:

1. A receiver joins a group; leaf Router C sends a join message toward the RP.
2. The RP puts a link to Router C in its outgoing interface list.
3. A source sends data; Router A encapsulates the data in a register message and sends it to the RP.
4. The RP forwards the data down the shared tree to Router C and sends a join message toward the source. At this point, data might arrive twice at Router C, once encapsulated and once natively.
5. When data arrives natively (unencapsulated) at the RP, it sends a register-stop message to Router A.
6. By default, reception of the first data packet prompts Router C to send a join message toward the source.
7. When Router C receives data on (S,G), it sends a prune message for the source up the shared tree.
8. The RP deletes the link to Router C from the outgoing interface of (S,G). The RP triggers a prune message toward the source.

Join and prune messages are sent for sources and RPs. They are sent hop-by-hop and are processed by each PIM device along the path to the source or RP. Register and register-stop messages are not sent hop-by-hop. They are sent by the designated router that is directly connected to a source and are received by the RP for the group.

Multiple sources sending to groups use the shared tree.

You can configure the PIM device to stay on the shared tree. For more information, see Delaying the Use of PIM Shortest-Path Tree, page 752.

Prerequisites

- To use multicast routing, the switch must be running the IP services image.

- Be familiar with the information in the Information About Cisco's Implementation of IP Multicast Routing, page 717 and Guidelines and Limitations, page 728.

Guidelines and Limitations

PIMv1 and PIMv2 Interoperability

The Cisco PIMv2 implementation provides interoperability and transition between Version 1 and Version 2, although there might be some minor problems.

You can upgrade to PIMv2 incrementally. PIM Versions 1 and 2 can be configured on different routers and multilayer switches within one network. Internally, all routers and multilayer switches on a shared media network must run the same PIM version. Therefore, if a PIMv2 device detects a PIMv1 device, the Version 2 device downgrades itself to Version 1 until all Version 1 devices have been shut down or upgraded.

PIMv2 uses the BSR to discover and announce RP-set information for each group prefix to all the routers and multilayer switches in a PIM domain. PIMv1, together with the Auto-RP feature, can perform the same tasks as the PIMv2 BSR. However, Auto-RP is a standalone protocol, separate from PIMv1, and is a proprietary Cisco protocol. PIMv2 is a standards track protocol in the IETF. We recommend that you use PIMv2. The BSR mechanism interoperates with Auto-RP on Cisco routers and multilayer switches. For more information, see Auto-RP and BSR Configuration Guidelines, page 728.

When PIMv2 devices interoperate with PIMv1 devices, Auto-RP should have already been deployed. A PIMv2 BSR that is also an Auto-RP mapping agent automatically advertises the RP elected by Auto-RP. That is, Auto-RP sets its single RP on every router or multilayer switch in the group. Not all routers and switches in the domain use the PIMv2 hash function to select multiple RPs.

Dense-mode groups in a mixed PIMv1 and PIMv2 region need no special configuration; they automatically interoperate.

Sparse-mode groups in a mixed PIMv1 and PIMv2 region are possible because the Auto-RP feature in PIMv1 interoperates with the PIMv2 RP feature. Although all PIMv2 devices can also use PIMv1, we recommend that the RPs be upgraded to PIMv2. To ease the transition to PIMv2, we have these recommendations:

- Use Auto-RP throughout the region.
- Configure sparse-dense mode throughout the region.

If Auto-RP is not already configured in the PIMv1 regions, configure Auto-RP. For more information, see Configuring Auto-RP, page 741.

Auto-RP and BSR Configuration Guidelines

There are two approaches to using PIMv2. You can use Version 2 exclusively in your network or migrate to Version 2 by employing a mixed PIM version environment.

- If your network is all Cisco routers and multilayer switches, you can use either Auto-RP or BSR.
- If you have non-Cisco routers in your network, you must use BSR.
- If you have Cisco PIMv1 and PIMv2 routers and multilayer switches and non-Cisco routers, you must use both Auto-RP and BSR. If your network includes routers from other vendors, configure the Auto-RP mapping agent and the BSR on a Cisco PIMv2 device. Ensure that no PIMv1 device is located in the path a between the BSR and a non-Cisco PIMv2 device.
- Because bootstrap messages are sent hop-by-hop, a PIMv1 device prevents these messages from reaching all routers and multilayer switches in your network. Therefore, if your network has a PIMv1 device in it and only Cisco routers and multilayer switches, it is best to use Auto-RP.
- If you have a network that includes non-Cisco routers, configure the Auto-RP mapping agent and the BSR on a Cisco PIMv2 router or multilayer switch. Ensure that no PIMv1 device is on the path between the BSR and a non-Cisco PIMv2 router.

- If you have non-Cisco PIMv2 routers that need to interoperate with Cisco PIMv1 routers and multilayer switches, both Auto-RP and a BSR are required. We recommend that a Cisco PIMv2 device be both the Auto-RP mapping agent and the BSR. For more information, see Using Auto-RP and a BSR, page 751.

PIM Stub Routing Configuration Guidelines

Guidelines and limitations for PIM stub routing are as follows:

- Before configuring PIM stub routing, you must have IP multicast routing configured on both the stub router and the central router. You must also have PIM mode (dense-mode, sparse-mode, or dense-sparse-mode) configured on the uplink interface of the stub router.
- The PIM stub router does not route the transit traffic between the distribution routers. Unicast (EIGRP) stub routing enforces this behavior. You must configure unicast stub routing to assist the PIM stub router behavior.
- Only directly connected multicast (IGMP) receivers and sources are allowed in the Layer 2 access domains. The PIM protocol is not supported in access domains.
- The redundant PIM stub router topology is not supported.

Restrictions for Legacy Applications Within the SSM Range

Existing applications in a network predating Source-Specific Multicast (SSM) do not work within the SSM range unless they are modified to support (S, G) channel subscriptions. Therefore, enabling SSM in a network can cause problems for existing applications if they use addresses within the designated SSM range.

Address Management Restrictions

Address management is still necessary to some degree when SSM is used with Layer 2 switching mechanisms. Cisco Group Management Protocol (CGMP), IGMP snooping, or Router-Port Group Management Protocol (RGMP) support only group-specific filtering, not (S, G) channel-specific filtering. If different receivers in a switched network request different (S, G) channels sharing the same group, they do not benefit from these existing mechanisms. Instead, both receivers receive all (S, G) channel traffic and filter out the unwanted traffic on input. Because SSM can re-use the group addresses in the SSM range for many independent applications, this situation can lead to decreased traffic filtering in a switched network. For this reason, it is important to use random IP addresses from the SSM range for an application to minimize the chance for re-use of a single address within the SSM range between different applications. For example, an application service providing a set of television channels should, even with SSM, use a different group for each television (S, G) channel. This setup guarantees that multiple receivers to different channels within the same application service never experience traffic aliasing in networks that include Layer 2 switches.

IGMP Snooping and CGMP Limitations

IGMPv3 uses new membership report messages that might not be correctly recognized by older IGMP snooping switches.

For more information about switching issues related to IGMP (especially with CGMP), refer to the “Configuring IGMP Version 3” section of the “Configuring IP Multicast Routing” chapter.

State Maintenance Limitations

In PIM-SSM, the last hop router continues to periodically send (S, G) join messages if appropriate (S, G) subscriptions are on the interfaces. Therefore, as long as receivers send (S, G) subscriptions, the shortest path tree (SPT) state from the receivers to the source is maintained, even if the source does not send traffic for longer periods of time (or even never).

This case is opposite to PIM-SM, where (S, G) state is maintained only if the source is sending traffic and receivers are joining the group. If a source stops sending traffic for more than 3 minutes in PIM-SM, the (S, G) state is deleted and only re-established after packets from the source arrive again through the RPT. Because no mechanism in PIM-SSM notifies a receiver that a source is active, the network must maintain the (S, G) state in PIM-SSM as long as receivers are requesting receipt of that channel.

SSM Mapping Configuration Guidelines

Guidelines and limitations for SSM mapping:

- The SSM mapping feature does not have all the benefits of full SSM. Because SSM mapping takes a group join from a host and identifies this group with an application associated with one or more sources, it can only support one such application per group. Full SSM applications can still share the same group as in SSM mapping.
- Enable IGMPv3 with care on the last hop router when you rely solely on SSM mapping as a transition solution for full SSM. When you enable both SSM mapping and IGMPv3 and the hosts already support IGMPv3 (but not SSM), the hosts send IGMPv3 group reports. SSM mapping does not support these IGMPv3 group reports, and the router does not correctly associate sources with these reports.

Default Settings

Feature	Default Setting
Multicast routing	Disabled on all interfaces.
PIM version	Version 2.
PIM mode	No mode is defined.
PIM RP address	None configured.
PIM domain border	Disabled.
PIM multicast boundary	None.
Candidate BSRs	Disabled.
Candidate RPs	Disabled.
Shortest-path tree threshold rate	0 kbps.
PIM router query message interval	30 seconds.

Configuring IP Multicast Routing

This section includes the following topics:

- Configuring Basic Multicast Routing, page 731 (required)
- Configuring PIM Stub Routing, page 733 (optional)
- Configuring Source-Specific Multicast, page 734
- Configuring SSM Mapping, page 735
- Configuring a Rendezvous Point, page 739 (required if the interface is in sparse-dense mode, and you want to treat the group as a sparse group)
- Using Auto-RP and a BSR, page 751 (required for non-Cisco PIMv2 devices to interoperate with Cisco PIM v1 devices))
- Monitoring the RP Mapping Information, page 751 (optional)
- Troubleshooting PIMv1 and PIMv2 Interoperability Problems, page 752 (optional)

Configuring Basic Multicast Routing

You must enable IP multicast routing and configure the PIM version and the PIM mode. Then the software can forward multicast packets, and the switch can populate its multicast routing table.

Note: To enable IP multicast routing, the switch must be running the IP services image.

You can configure an interface to be in PIM dense mode, sparse mode, or sparse-dense mode. The switch populates its multicast routing table and forwards multicast packets it receives from its directly connected LANs according to the mode setting. You must enable PIM in one of these modes for an interface to perform IP multicast routing. Enabling PIM on an interface also enables IGMP operation on that interface.

Note: If you enable PIM on multiple interfaces and most of these interfaces are not part of the outgoing interface list, when IGMP snooping is disabled the outgoing interface might not be able to sustain line rate for multicast traffic because of the extra, unnecessary replication.

In populating the multicast routing table, dense-mode interfaces are always added to the table. Sparse-mode interfaces are added to the table only when periodic join messages are received from downstream devices or when there is a directly connected member on the interface. When forwarding from a LAN, sparse-mode operation occurs if there is an RP known for the group. If so, the packets are encapsulated and sent toward the RP. When no RP is known, the packet is flooded in a dense-mode fashion. If the multicast traffic from a specific source is sufficient, the receiver's first-hop router might send join messages toward the source to build a source-based distribution tree.

By default, multicast routing is disabled, and there is no default mode setting. Follow this procedure to enable IP multicasting, to configure a PIM version, and to configure a PIM mode. This procedure is required.

BEFORE YOU BEGIN

- Decide which PIM mode to use.
- Ensure that the interface on which you are enabling multicast routing has an IP address assigned to it.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip multicast-routing distributed**	Enable IP multicast distributed switching.
3.	**interface** *interface-id*	Specify the Layer 3 interface on which you want to enable multicast routing, and enter interface configuration mode. The specified interface must be one of the following: ■ A routed port: a physical port that has been configured as a Layer 3 port by entering the **no switchport** interface configuration command. ■ An SVI: a VLAN interface created by using the **interface vlan** *vlan-id* global configuration command.
4.	**no shutdown**	Enable the port, if necessary. By default, user network interfaces (UNIs) and enhanced network interfaces (ENIs) are disabled, and network node interfaces (NNIs) are enabled.
5.	**ip pim version** [**1** | **2**]	Configure the PIM version on the interface. By default, Version 2 is enabled and is the recommended setting. An interface in PIMv2 mode automatically downgrades to PIMv1 mode if that interface has a PIMv1 neighbor. The interface returns to Version 2 mode after all Version 1 neighbors are shut down or upgraded. For more information, see PIMv1 and PIMv2 Interoperability, page 728.
6.	**ip pim** {**dense-mode** | **sparse-mode** | **sparse-dense-mode**}	Enable a PIM mode on the interface. By default, no mode is configured. The keywords have these meanings: ■ **dense-mode**–Enables dense mode of operation. ■ **sparse-mode**–Enables sparse mode of operation. If you configure sparse-mode, you must also configure an RP. For more information, see Configuring a Rendezvous Point, page 739. ■ **sparse-dense-mode**–Causes the interface to be treated in the mode in which the group belongs. Sparse-dense-mode is the recommended setting.
7.	**end**	Return to privileged EXEC mode.
8.	**show running-config**	Verify your entries.
9.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable multicasting, use the **no ip multicast-routing distributed** global configuration command. To return to the default PIM version, use the **no ip pim version** interface configuration command. To disable PIM on an interface, use the **no ip pim** interface configuration command.

EXAMPLE

This example enables IP multicast distributed switching and specifies the PIM mode:

```
Switch# configure terminal
Switch(config)# ip multicast-routing distributed
Switch(config)# interface Gigabitethernet 1/0/0
Switch(config-if)# ip pim sparse-dense-mode
Switch(config-if)# end
```

Configuring PIM Stub Routing

The PIM Stub routing feature supports multicast routing between the distribution layer and the access layer. It supports two types of PIM interfaces, uplink PIM interfaces, and PIM passive interfaces. A routed interface configured with the PIM passive mode does not pass or forward PIM control traffic, it only passes and forwards IGMP traffic.

This procedure is optional.

BEFORE YOU BEGIN

- You must have IP multicast routing configured on both the stub router and the central router.
- You must have PIM mode (dense-mode, sparse-mode, or dense-sparse-mode) configured on the uplink interface of the stub router.
- You must configure EIGRP stub routing to assist the PIM stub router behavior.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Specify the interface on which you want to enable PIM stub routing, and enter interface configuration mode.
3.	**ip pim passive**	Configure the PIM stub feature on the interface.
4.	**end**	Return to privileged EXEC mode.
5.	**show ip pim interface**	Display the PIM stub that is enabled on each interface.
6.	**show running-config**	Verify your entries.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable PIM stub routing on an interface, use the **no ip pim passive** interface configuration command.

EXAMPLE

In this example, IP multicast routing is enabled, Switch A PIM uplink port 25 is configured as a routed uplink port with **spare-dense-mode enabled.** PIM stub routing is enabled on the VLAN 100 interfaces and on Gigabit Ethernet port 20 in Figure 88 on page 721:

```
Switch(config)# ip multicast-routing distributed
Switch(config)# interface GigabitEthernet0/25
Switch(config-if)# no switchport
Switch(config-if)# ip address 3.1.1.2 255.255.255.0
Switch(config-if)# ip pim sparse-dense-mode
Switch(config-if)# exit
Switch(config)# interface vlan100
Switch(config-if)# ip pim passive
Switch(config-if)# exit
Switch(config)# interface GigabitEthernet0/20
Switch(config-if)# ip pim passive
Switch(config-if)# exit
Switch(config)# interface vlan100
Switch(config-if)# ip address 100.1.1.1 255.255.255.0
Switch(config-if)# ip pim passive
Switch(config-if)# exit
Switch(config)# interface GigabitEthernet0/20
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.1.1.1 255.255.255.0
Switch(config-if)# ip pim passive
Switch(config-if)# end
```

To verify that PIM stub is enabled for each interface, use the **show ip pim interface** privileged EXEC command:

```
Switch# show ip pim interface
Address Interface Ver/ Nbr Query DR DR
Mode Count Intvl Prior
3.1.1.2 GigabitEthernet0/25 v2/SD 1 30 1 3.1.1.2
100.1.1.1 Vlan100 v2/P 0 30 1 100.1.1.1
10.1.1.1 GigabitEthernet0/20 v2/P 0 30 1 10.1.1.1
```

Use these privileged EXEC commands to display information about PIM stub configuration and status:

- **show ip pim interface** displays the PIM stub that is enabled on each interface.
- **show ip igmp detail** displays the interested clients that have joined the specific multicast source group.
- **show ip igmp mroute** verifies that the multicast stream forwards from the source to the interested clients.

Configuring Source-Specific Multicast

This section describes how to configure source-specific multicast (SSM).

BEFORE YOU BEGIN

See Information About Source-Specific Multicast, page 723 and Guidelines and Limitations, page 728.

DETAILED STEPS

	Command	Purpose
1.	**ip pim ssm** [**default** | **range** *access-list*]	Define the SSM range of IP multicast addresses.
2.	**interface** *type number*	Select an interface that is connected to hosts on which IGMPv3 can be enabled, and enter the interface configuration mode.
3.	**ip pim** {**sparse-mode** | **sparse-dense-mode**}	Enable PIM on an interface. You must use either **sparse mode** or **sparse-dense mode**.
4.	**ip igmp version 3**	Enable IGMPv3 on this interface. The default version of IGMP is set to Version 2.

EXAMPLE

The following example shows how to configure a device (running IGMPv3) for SSM:

```
ip multicast-routing
ip pim ssm default
!
```

```
interface GigabitEthernet3/1/0
 ip address 172.21.200.203 255.255.255.0
 description backbone interface
    ip pim sparse-mode
!
interface GigabitEthernet3/2/0
    ip address 131.108.1.2 255.255.255.0
    ip pim sparse-mode
    description ethernet connected to hosts
    ip igmp version 3
!
```

Verifying SSM Configuration

Command	Purpose
show ip igmp groups detail	Display the (S, G) channel subscription through IGMPv3.
show ip mroute	Display whether a multicast group supports SSM service or whether a source-specific host report was received.

Configuring SSM Mapping

This section includes the following topics:

- Configuring Static SSM Mapping, page 735 (required)
- Configuring DNS-Based SSM Mapping, page 736 (required)
- Configuring Static Traffic Forwarding with SSM Mapping, page 738 (optional)

Configuring Static SSM Mapping

BEFORE YOU BEGIN

- See Information About Source Specific Multicast Mapping, page 725 and SSM Mapping Configuration Guidelines, page 730.
- Before you configure SSM mapping, enable IP multicast routing, enable PIM sparse mode, and configure SSM. For information on enabling IP multicast routing and PIM sparse mode, see Configuring Basic Multicast Routing, page 731.
- Before you configure static SSM mapping, you must configure access control lists (ACLs) that define the group ranges to be mapped to source addresses.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip igmp ssm-map enable**	Enable SSM mapping for groups in the configured SSM range. **Note:** By default, this command enables DNS-based SSM mapping.
3.	**no ip igmp ssm-map query dns**	(Optional) Disable DNS-based SSM mapping. **Note:** Disable DNS-based SSM mapping if you only want to rely on static SSM mapping. By default, the **ip igmp ssm-map** global configuration command enables DNS-based SSM mapping.
4.	**ip igmp ssm-map static** *access-list source-address*	Configure static SSM mapping. The ACL supplied for *access-list* defines the groups to be mapped to the source IP address entered for the *source-address*. **Note:** You can configure additional static SSM mappings. If additional SSM mappings are configured and the router receives an IGMPv1 or IGMPv2 membership report for a group in the SSM range, the switch determines the source addresses associated with the group by using each configured **ip igmp ssm-map static** command. The switch associates up to 20 sources per group.
5.	Repeat Step 4 to configure additional static SSM mappings, if required.	–
6.	**end**	Return to privileged EXEC mode.
7.	**show running-config**	Verify your entries.
8.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

The following example shows how to enable static SSM mapping. In this example, the router is configured to statically map groups that match ACL 11 to source address 172.16.8.11 and to statically map groups that match ACL 10 to source address 172.16.8.10.

```
Switch(config)# ip igmp ssm-map enable
Switch(config)# ip igmp ssm-map static 11 172.16.8.11
Switch(config)# ip igmp ssm-map static 10 172.16.8.10
Switch(config)# end
```

Configuring DNS-Based SSM Mapping

To configure DNS-based SSM mapping, you need to create a DNS server zone or add records to an existing zone. If the routers that are using DNS-based SSM mapping are also using DNS for other purposes, you should use a normally configured DNS server. If DNS-based SSM mapping is the only DNS implementation being used on the router, you can configure a false DNS setup with an empty root zone or a root zone that points back to itself.

BEFORE YOU BEGIN

- See Information About Source Specific Multicast Mapping, page 725 and SSM Mapping Configuration Guidelines, page 730.
- Before you configure SSM mapping, enable IP multicast routing, enable PIM sparse mode, and configure SSM. For information on enabling IP multicast routing and PIM sparse mode, see Configuring Basic Multicast Routing, page 731.
- Before you can configure and use SSM mapping with DNS lookups, you must be able to add records to a running DNS server. If you do not already have a DNS server running, you need to install one.

  You can use a product such as Cisco Network Registrar. Go to this URL for more information:

  http://www.cisco.com/en/US/products/sw/netmgtsw/ps1982/index.html

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip igmp ssm-map enable**	Enable SSM mapping for groups in a configured SSM range.
3.	**ip igmp ssm-map query dns**	(Optional) Enable DNS-based SSM mapping. By default, the **ip igmp ssm-map** command enables DNS-based SSM mapping. Only the **no** form of this command is saved to the running configuration. **Note:** Use this command to re-enable DNS-based SSM mapping if DNS-based SSM mapping is disabled.
4.	**ip domain multicast** *domain-prefix*	(Optional) Change the domain prefix used by the switch for DNS-based SSM mapping. By default, the switch uses the *ip-addr.arpa* domain prefix.
5.	**ip name-server** *server-address1* [*server-address2... server-address6*]	Specify the address of one or more name servers to use for name and address resolution.
6.	Repeat Step 5 to configure additional DNS servers for redundancy, if required.	–
7.	**end**	Return to privileged EXEC mode.
8.	**show running-config**	Verify your entries.
9.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

The following example shows how to configure DNS-based SSM mapping:

```
Switch(config)# ip igmp ssm-map enable
Switch(config)# ip name-server 10.0.0.0
Switch(config)# end
```

Configuring Static Traffic Forwarding with SSM Mapping

Use static traffic forwarding with SSM mapping to statically forward SSM traffic for certain groups. When static traffic forwarding with SSM mapping is configured, the last hop router uses Domain Name System (DNS)-based SSM mapping to determine the sources associated with a group. The resulting (S, G) channels are then statically forwarded.

BEFORE YOU BEGIN

Configure DNS-based SSM mapping as described in the Configuring DNS-Based SSM Mapping, page 736.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *type number*	Select an interface on which to statically forward traffic for a multicast group using SSM mapping, and enter interface configuration mode. **Note:** Static forwarding of traffic with SSM mapping works with either DNS-based SSM mapping or statically configured SSM mapping.
3.	**ip igmp static-group** *group-address* **source ssm-map**	Configure SSM mapping to statically forward a (S, G) channel from the interface. Use this command if you want to statically forward SSM traffic for certain groups. Use DNS-based SSM mapping to determine the source addresses of the channels.
4.	**show running-config**	Verify your entries.
5.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

The following example shows how to configure group address 239.1.2.1 to use SSM mapping for statically forwarded groups on Ethernet interface 0:

```
interface ethernet 0
 ip igmp static-group 239.1.2.1 source ssm-map
```

Verifying SSM Mapping Configuration

Command	Purpose
show ip igmp ssm-mapping	Display information about SSM mapping.
show ip igmp ssm-mapping *group-address*	Display the sources that SSM mapping uses for a particular group.
show ip igmp groups [*group-name* | *group-address* | *interface-type interface-number*] [**detail**]	Display the multicast groups with receivers that are directly connected to the router and that were learned through IGMP.
show host	Display the default domain name, the style of name lookup service, a list of name server hosts, and the cached list of hostnames and addresses.
debug ip igmp *group-address*	Display the IGMP packets received and sent and IGMP host-related events.

Configuring a Rendezvous Point

You must have an RP if the interface is in sparse-dense mode and if you want to treat the group as a sparse group. You can use several methods, as described in these sections:

- Manually Assigning an RP to Multicast Groups, page 739
- Configuring Auto-RP, page 741 (a standalone, Cisco-proprietary protocol separate from PIMv1)
- Configuring PIMv2 BSR, page 746 (a standards track protocol in the Internet Engineering Task Force (IETF)

You can use Auto-RP, BSR, or a combination of both, depending on the PIM version you are running and the types of routers in your network. For more information, see PIMv1 and PIMv2 Interoperability, page 728 and the Auto-RP and BSR Configuration Guidelines, page 728.

Manually Assigning an RP to Multicast Groups

This section explains how to manually configure an RP. If the RP for a group is learned through a dynamic mechanism (such as Auto-RP or BSR), you need not perform this task for that RP.

Senders of multicast traffic announce their existence through register messages received from the source’s first-hop router (designated router) and forwarded to the RP. Receivers of multicast packets use RPs to join a multicast group by using explicit join messages. RPs are not members of the multicast group; rather, they serve as a *meeting place* for multicast sources and group members.

You can configure a single RP for multiple groups defined by an access list. If there is no RP configured for a group, the multilayer switch treats the group as dense and uses the dense-mode PIM techniques. This procedure is optional.

BEFORE YOU BEGIN

Review the Information About PIM, page 719 and Guidelines and Limitations, page 728.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip pim rp-address** *ip-address* [*access-list-number*] [**override**]	Configure the address of a PIM RP. By default, no PIM RP address is configured. You must configure the IP address of RPs on all routers and multilayer switches (including the RP). If there is no RP configured for a group, the switch treats the group as dense, using the dense-mode PIM techniques. A PIM device can be an RP for more than one group. Only one RP address can be used at a time within a PIM domain. The access-list conditions specify for which groups the device is an RP. ■ For *ip-address*, enter the unicast address of the RP in dotted-decimal notation. ■ (Optional) For *access-list-number*, enter an IP standard access list number from 1 to 99. If no access list is configured, the RP is used for all groups. ■ (Optional) The **override** keyword means that if there is a conflict between the RP configured with this command and one learned by Auto-RP or BSR, the RP configured with this command prevails.
3.	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]	Create a standard access list, repeating the command as many times as necessary. ■ For *access-list-number*, enter the access list number specified in Step 2. ■ The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. ■ For *source*, enter the multicast group address for which the RP should be used. ■ (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
4.	**end**	Return to privileged EXEC mode.
5.	**show running-config**	Verify your entries.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove an RP address, use the **no ip pim rp-address** *ip-address* [*access-list-number*] [**override**] global configuration command.

EXAMPLE

This example shows how to configure the address of the RP to 147.106.6.22 for multicast group 225.2.2.2 only:

```
Switch(config)# access-list 1 permit 225.2.2.2 0.0.0.0
Switch(config)# ip pim rp-address 147.106.6.22 1
```

Configuring Auto-RP

Auto-RP uses IP multicast to automate the distribution of group-to-RP mappings to all Cisco routers and multilayer switches in a PIM network. It has these benefits:

- It is easy to use multiple RPs within a network to serve different group ranges.
- It provides load splitting among different RPs and arrangement of RPs according to the location of group participants.
- It avoids inconsistent, manual RP configurations on every router and multilayer switch in a PIM network, which can cause connectivity problems.

Note: If you configure PIM in sparse mode or sparse-dense mode and do not configure Auto-RP, you must manually configure an RP as described in the Manually Assigning an RP to Multicast Groups, page 739.

Note: If routed interfaces are configured in sparse mode, Auto-RP can still be used if all devices are configured with a manual RP address for the Auto-RP groups.

These sections describe how to configure Auto-RP:

- Setting up Auto-RP in a New Internetwork, page 741 (optional)
- Adding Auto-RP to an Existing Sparse-Mode Cloud, page 741 (optional)
- Preventing Join Messages to False RPs, page 743 (optional)
- Filtering Incoming RP Announcement Messages, page 743 (optional)

Setting up Auto-RP in a New Internetwork

If you are setting up Auto-RP in a new internetwork, you do not need a default RP because you configure all the interfaces for sparse-dense mode. Follow the process described in the Adding Auto-RP to an Existing Sparse-Mode Cloud, page 741. However, omit Step 3 if you want to configure a PIM router as the RP for the local group.

Adding Auto-RP to an Existing Sparse-Mode Cloud

This section contains some suggestions for the initial deployment of Auto-RP into an existing sparse-mode cloud to minimize disruption of the existing multicast infrastructure. This procedure is optional.

BEFORE YOU BEGIN

- Review the Auto-RP, page 721 and Guidelines and Limitations, page 728.
- Configure a default RP as described in the Manually Assigning an RP to Multicast Groups, page 739.

DETAILED STEPS

	Command	Purpose
1.	**show running-config**	Verify that a default RP is already configured on all PIM devices and the RP in the sparse-mode network. It was previously configured with the **ip pim rp-address** global configuration command. This step is not required for spare-dense-mode environments. The selected RP should have good connectivity and be available across the network. Use this RP for the global groups (for example 224.x.x.x and other global groups). Do not reconfigure the group address range that this RP serves. RPs dynamically discovered through Auto-RP take precedence over statically configured RPs. Assume that it is desirable to use a second RP for the local groups.
2.	**configure terminal**	Enter global configuration mode.
3.	**ip pim send-rp-announce** *interface-id* **scope** *ttl* **group-list** *access-list-number* **interval** *seconds*	Configure another PIM device to be the candidate RP for local groups. ■ For *interface-id*, enter the interface type and number that identifies the RP address. Valid interfaces include physical ports, port channels, and VLANs. ■ For **scope** *ttl*, specify the time-to-live value in hops. Enter a hop count that is high enough so that the RP-announce messages reach all mapping agents in the network. There is no default setting. The range is 1 to 255. ■ For **group-list** *access-list-number*, enter an IP standard access list number from 1 to 99. If no access list is configured, the RP is used for all groups. ■ For **interval** *seconds*, specify how often the announcement messages must be sent. The default is 60 seconds. The range is 1 to 16383.
4.	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]	Create a standard access list, repeating the command as many times as necessary. ■ For *access-list-number*, enter the access list number specified in Step 3. ■ The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. ■ For *source*, enter the multicast group address range for which the RP should be used. ■ (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.

	Command	Purpose
5.	**ip pim send-rp-discovery scope** *ttl*	Find a switch whose connectivity is not likely to be interrupted, and assign it the role of RP-mapping agent. For **scope** *ttl*, specify the time-to-live value in hops to limit the RP discovery packets. All devices within the hop count from the source device receive the Auto-RP discovery messages. These messages tell other devices which group-to-RP mapping to use to avoid conflicts (such as overlapping group-to-RP ranges). There is no default setting. The range is 1 to 255.
6.	**end**	Return to privileged EXEC mode.
7.	**show running-config** **show ip pim rp mapping** **show ip pim rp**	Verify your entries. Display active RPs that are cached with associated multicast routing entries. Display the information cached in the routing table.
8.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the PIM device configured as the candidate RP, use the **no ip pim send-rp-announce** *interface-id* global configuration command. To remove the switch as the RP-mapping agent, use the **no ip pim send-rp-discovery** global configuration command.

EXAMPLE

This example shows how to send RP announcements out all PIM-enabled interfaces for a maximum of 31 hops. The IP address of port 1 is the RP. Access list 5 describes the group for which this switch serves as RP:

```
Switch(config)# ip pim send-rp-announce gigabitethernet0/1 scope 31 group-list 5
Switch(config)# access-list 5 permit 224.0.0.0 15.255.255.255
```

Preventing Join Messages to False RPs

Find whether the **ip pim accept-rp** command was previously configured throughout the network by using the **show running-config** privileged EXEC command. If the **ip pim accept-rp** command is not configured on any device, this problem can be addressed later. In those routers or multilayer switches already configured with the **ip pim accept-rp** command, you must enter the command again to accept the newly advertised RP.

To accept all RPs advertised with Auto-RP and reject all other RPs by default, use the **ip pim accept-rp auto-rp** global configuration command. This procedure is optional.

If all interfaces are in sparse mode, use a default-configured RP to support the two well-known groups 224.0.1.39 and 224.0.1.40. Auto-RP uses these two well-known groups to collect and distribute RP-mapping information. When this is the case and the **ip pim accept-rp auto-rp** command is configured, another **ip pim accept-rp** command accepting the RP must be configured as follows:

```
Switch(config)# ip pim accept-rp 172.10.20.1 1
Switch(config)# access-list 1 permit 224.0.1.39
Switch(config)# access-list 1 permit 224.0.1.40
```

Filtering Incoming RP Announcement Messages

You can add configuration commands to the mapping agents to prevent a maliciously configured router from masquerading as a candidate RP and causing problems. This procedure is optional.

BEFORE YOU BEGIN

- This command should only be configured on RP mapping agents.
- If you use more than one RP-mapping agent, you must configure the same filters on all mapping agents to avoid inconsistencies in Auto-RP operations.
- An improperly configured **ip pim rp-announce-filter** command may result in RP announcements being ignored. In addition, the **ip pim rp-announce-filter** command should only be configured on the mapping agent; if not, the command will fail because non-mapping agents do not listen to group 224.0.1.39 and do not know how to distribute the necessary group-to-RP mappings.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip pim rp-announce-filter rp-list** *access-list-number* **group-list** *access-list-number*	Filter incoming RP announcement messages. Enter this command on each mapping agent in the network. Without this command, all incoming RP-announce messages are accepted by default. For **rp-list** *access-list-number*, configure an access list of candidate RP addresses that, if permitted, is accepted for the group ranges supplied in the **group-list** *access-list-number* variable. If this variable is omitted, the filter applies to all multicast groups. If more than one mapping agent is used, the filters must be consistent across all mapping agents to ensure that no conflicts occur in the Group-to-RP mapping information.
3.	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]	Create a standard access list, repeating the command as many times as necessary. ■ For *access-list-number*, enter the access list number specified in Step 2. ■ The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. ■ Create an access list that specifies from which routers and multilayer switches the mapping agent accepts candidate RP announcements (rp-list ACL). ■ Create an access list that specifies the range of multicast groups from which to accept or deny (group-list ACL). ■ For *source*, enter the multicast group address range for which the RP should be used. ■ (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
4.	**end**	Return to privileged EXEC mode.
5.	**show running-config**	Verify your entries.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove a filter on incoming RP announcement messages, use the **no ip pim rp-announce-filter rp-list** *access-list-number* [**group-list** *access-list-number*] global configuration command.

EXAMPLE

This example shows a sample configuration on an Auto-RP mapping agent that is used to prevent candidate RP announcements from being accepted from unauthorized candidate RPs:

```
Switch(config)# ip pim rp-announce-filter rp-list 10 group-list 20
Switch(config)# access-list 10 permit host 172.16.5.1
Switch(config)# access-list 10 permit host 172.16.2.1
Switch(config)# access-list 20 deny 239.0.0.0 0.0.255.255
Switch(config)# access-list 20 permit 224.0.0.0 15.255.255.255
```

In this example, the mapping agent accepts candidate RP announcements from only two devices, 172.16.5.1 and 172.16.2.1. The mapping agent accepts candidate RP announcements from these two devices only for multicast groups that fall in the group range of 224.0.0.0 to 239.255.255.255. The mapping agent does not accept candidate RP announcements from any other devices in the network. Furthermore, the mapping agent does not accept candidate RP announcements from 172.16.5.1 or 172.16.2.1 if the announcements are for any groups in the 239.0.0.0 through 239.255.255.255 range. This range is the administratively scoped address range.

Configuring PIMv2 BSR

These sections describe how to set up BSR in your PIMv2 network:

- Defining the PIM Domain Border, page 746 (optional)
- Defining the IP Multicast Boundary, page 747 (optional)
- Configuring Candidate BSRs, page 748 (optional)
- Configuring Candidate RPs, page 749 (optional)

Defining the PIM Domain Border

As IP multicast becomes more widespread, the chance of one PIMv2 domain bordering another PIMv2 domain is increasing. Because these two domains probably do not share the same set of RPs, BSR, candidate RPs, and candidate BSRs, you need to constrain PIMv2 BSR messages from flowing into or out of the domain. Allowing these messages to leak across the domain borders could adversely affect the normal BSR election mechanism and elect a single BSR across all bordering domains and co-mingle candidate RP advertisements, resulting in the election of RPs in the wrong domain. This procedure is optional.

BEFORE YOU BEGIN

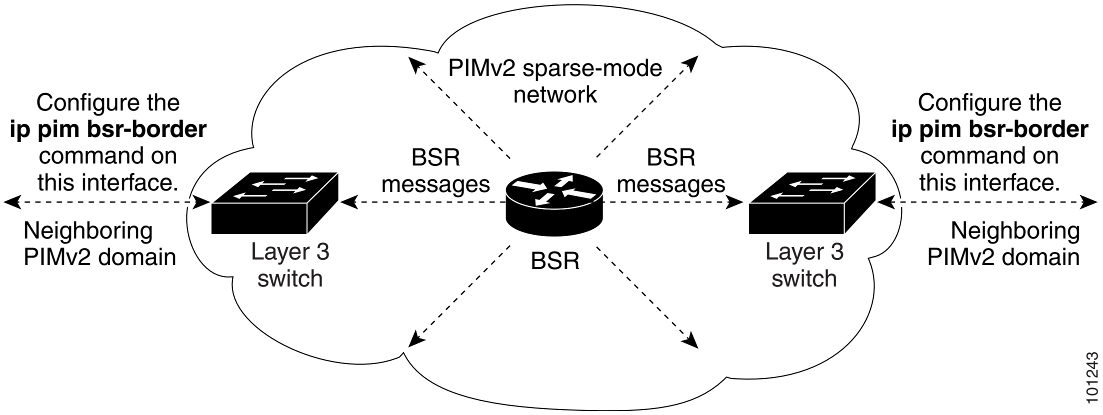
Review the Bootstrap Router, page 722 and Guidelines and Limitations, page 728.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
3.	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
4.	**ip pim bsr-border**	Define a PIM bootstrap message boundary for the PIM domain. Enter this command on each interface that connects to other bordering PIM domains. This command instructs the switch to neither send or receive PIMv2 BSR messages on this interface as shown in Figure 92 on page 747.
5.	**end**	Return to privileged EXEC mode.
6.	**show running-config**	Verify your entries.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the PIM border, use the **no ip pim bsr-border** interface configuration command.

Figure 92 Constraining PIMv2 BSR Messages


PIMv2 sparse-mode network
Configure the **ip pim bsr-border** command on this interface.
Neighboring PIMv2 domain
Layer 3 switch
BSR messages
BSR
BSR messages
Layer 3 switch
Configure the **ip pim bsr-border** command on this interface.
Neighboring PIMv2 domain
101243

EXAMPLE

The following example configures the interface to be the PIM domain border:

```
interface ethernet 1
ip pim bsr-border
```

Defining the IP Multicast Boundary

You define a multicast boundary to prevent Auto-RP messages from entering the PIM domain. You create an access list to deny packets destined for 224.0.1.39 and 224.0.1.40, which carry Auto-RP information. This procedure is optional.

BEFORE YOU BEGIN

Review the Information About PIM, page 719 and the Guidelines and Limitations, page 728.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**access-list** *access-list-number* **deny** *source* [*source-wildcard*]	Create a standard access list, repeating the command as many times as necessary. ■ For *access-list-number*, the range is 1 to 99. ■ The **deny** keyword denies access if the conditions are matched. ■ For *source*, enter multicast addresses 224.0.1.39 and 224.0.1.40, which carry Auto-RP information. ■ (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
3.	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
4.	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
5.	**ip multicast boundary** *access-list-number*	Configure the boundary, specifying the access list you created in Step 2.
6.	**end**	Return to privileged EXEC mode.
7.	**show running-config**	Verify your entries.
8.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the boundary, use the **no ip multicast boundary** interface configuration command.

EXAMPLE

This example shows a portion of an IP multicast boundary configuration that denies Auto-RP information:

```
Switch(config)# access-list 1 deny 224.0.1.39
Switch(config)# access-list 1 deny 224.0.1.40
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip multicast boundary 1
```

Configuring Candidate BSRs

You can configure one or more candidate BSRs. The devices serving as candidate BSRs should have good connectivity to other devices and be in the backbone portion of the network. This procedure is optional.

BEFORE YOU BEGIN

Enable PIM on the interface using the **ip pim** command as described in the Configuring Basic Multicast Routing, page 731.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip pim bsr-candidate** *interface-id hash-mask-length* [*priority*]	Configure your switch to be a candidate BSR. ■ For *interface-id,* enter the interface on this switch from which the BSR address is derived to make it a candidate. This interface must be enabled with PIM. Valid interfaces include physical ports, port channels, and VLANs. ■ For *hash-mask-length*, specify the mask length (32 bits maximum) that is to be ANDed with the group address before the hash function is called. All groups with the same seed hash correspond to the same RP. For example, if this value is 24, only the first 24 bits of the group addresses matter. ■ (Optional) For *priority*, enter a number from 0 to 255. The BSR with the larger priority is preferred. If the priority values are the same, the device with the highest IP address is selected as the BSR. The default is 0.
3.	**end**	Return to privileged EXEC mode.
4.	**show running-config**	Verify your entries.
5.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove this device as a candidate BSR, use the **no ip pim bsr-candidate** global configuration command.

EXAMPLE

This example shows how to configure a candidate BSR, which uses the IP address 172.21.24.18 on a port as the advertised BSR address, uses 30 bits as the hash-mask-length, and has a priority of 10:

```
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# ip address 172.21.24.18 255.255.255.0
Switch(config-if)# ip pim sparse-dense-mode
Switch(config-if)# ip pim bsr-candidate gigabitethernet0/2 30 10
```

Configuring Candidate RPs

You can configure one or more candidate RPs. Similar to BSRs, the RPs should also have good connectivity to other devices and be in the backbone portion of the network. An RP can serve the entire IP multicast address space or a portion of it. Candidate RPs send candidate RP advertisements to the BSR. When deciding which devices should be RPs, consider these options:

- In a network of Cisco routers and multilayer switches where only Auto-RP is used, any device can be configured as an RP.
- In a network that includes only Cisco PIMv2 routers and multilayer switches and with routers from other vendors, any device can be used as an RP.
- In a network of Cisco PIMv1 routers, Cisco PIMv2 routers, and routers from other vendors, configure only Cisco PIMv2 routers and multilayer switches as RPs.

This procedure is optional.

Configuring IP Multicast Routing

BEFORE YOU BEGIN

Enable PIM on the interface using the **ip pim** command as described in the Configuring Basic Multicast Routing, page 731.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip pim rp-candidate** *interface-id* [**group-list** *access-list-number*]	Configure your switch to be a candidate RP. ■ For *interface-id,* specify the interface whose associated IP address is advertised as a candidate RP address. Valid interfaces include physical ports, port channels, and VLANs. ■ (Optional) For **group-list** *access-list-number*, enter an IP standard access list number from 1 to 99. If no group-list is specified, the switch is a candidate RP for all groups.
3.	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]	Create a standard access list, repeating the command as many times as necessary. ■ For *access-list-number*, enter the access list number specified in Step 2. ■ The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. ■ For *source*, enter the number of the network or host from which the packet is being sent. ■ (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
4.	**end**	Return to privileged EXEC mode.
5.	**show running-config**	Verify your entries.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove this device as a candidate RP, use the **no ip pim rp-candidate** *interface-id* global configuration command.

EXAMPLE

This example shows how to configure the switch to advertise itself as a candidate RP to the BSR in its PIM domain. Standard access list number 4 specifies the group prefix associated with the RP that has the address identified by a port. That RP is responsible for the groups with the prefix 239.

```
Switch(config)# ip pim rp-candidate gigabitethernet0/2 group-list 4
Switch(config)# access-list 4 permit 239.0.0.0 0.255.255.255
```

Using Auto-RP and a BSR

If there are only Cisco devices in you network (no routers from other vendors), there is no need to configure a BSR. Configure Auto-RP in a network that is running both PIMv1 and PIMv2.

If you have non-Cisco PIMv2 routers that need to interoperate with Cisco PIMv1 routers and multilayer switches, both Auto-RP and a BSR are required. We recommend that a Cisco PIMv2 router or multilayer switch be both the Auto-RP mapping agent and the BSR.

If you must have one or more BSRs, we have these recommendations:

- Configure the candidate BSRs as the RP-mapping agents for Auto-RP. For more information, see Configuring Auto-RP, page 741 and Configuring Candidate BSRs, page 748.
- For group prefixes advertised through Auto-RP, the PIMv2 BSR mechanism should not advertise a subrange of these group prefixes served by a different set of RPs. In a mixed PIMv1 and PIMv2 domain, have backup RPs serve the same group prefixes. This prevents the PIMv2 DRs from selecting a different RP from those PIMv1 DRs, due to the longest match lookup in the RP-mapping database.

Follow this procedure to verify the consistency of group-to-RP mappings. This procedure is optional.

BEFORE YOU BEGIN

Review the Auto-RP and BSR Configuration Guidelines, page 728.

DETAILED STEPS

	Command	Purpose
1.	**show ip pim rp** [[*group-name* | *group-address*] | **mapping**]	On any Cisco device, display the available RP mappings. ■ (Optional) For *group-name*, specify the name of the group about which to display RPs. ■ (Optional) For *group-address*, specify the address of the group about which to display RPs. ■ (Optional) Use the **mapping** keyword to display all group-to-RP mappings of which the Cisco device is aware (either configured or learned from Auto-RP).
2.	**show ip pim rp-hash** *group*	On a PIMv2 router or multilayer switch, confirm that the same RP is the one that a PIMv1 system chooses. For *group*, enter the group address for which to display RP information.

Monitoring the RP Mapping Information

To monitor the RP mapping information, use these commands in privileged EXEC mode:

- **show ip pim bsr** displays information about the elected BSR.
- **show ip pim rp-hash** *group* displays the RP that was selected for the specified group.
- **show ip pim rp** [*group-name* | *group-address* | **mapping**] displays how the switch learns of the RP (through the BSR or the Auto-RP mechanism).

Troubleshooting PIMv1 and PIMv2 Interoperability Problems

When debugging interoperability problems between PIMv1 and PIMv2, check these in the order shown:

1. Verify RP mapping with the **show ip pim rp-hash** privileged EXEC command, making sure that all systems agree on the same RP for the same group.
2. Verify interoperability between different versions of DRs and RPs. Make sure the RPs are interacting with the DRs properly (by responding with register-stops and forwarding decapsulated data packets from registers).

Configuring Advanced PIM Features

This section includes the following topics:

- Delaying the Use of PIM Shortest-Path Tree, page 752 (optional)
- Modifying the PIM Router-Query Message Interval, page 754 (optional)

Delaying the Use of PIM Shortest-Path Tree

The change from shared to source tree happens when the first data packet arrives at the last-hop router (Router C in Figure 91 on page 727 in the Information About PIM Shared Tree and Source Tree, page 726). This change occurs because the **ip pim spt-threshold** global configuration command controls that timing.

The shortest-path tree requires more memory than the shared tree but reduces delay. You might want to postpone its use. Instead of allowing the leaf router to immediately move to the shortest-path tree, you can specify that the traffic must first reach a threshold.

You can configure when a PIM leaf router should join the shortest-path tree for a specified group. If a source sends at a rate greater than or equal to the specified kbps rate, the multilayer switch triggers a PIM join message toward the source to construct a source tree (shortest-path tree). If the traffic rate from the source drops below the threshold value, the leaf router switches back to the shared tree and sends a prune message toward the source.

You can specify to which groups the shortest-path tree threshold applies by using a group list (a standard access list). If a value of 0 is specified or if the group list is not used, the threshold applies to all groups.

This procedure is optional.

BEFORE YOU BEGIN

Review the Information About PIM Shared Tree and Source Tree, page 726.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]	Create a standard access list. ■ For *access-list-number*, the range is 1 to 99. ■ The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. ■ For *source*, specify the multicast group to which the threshold will apply. ■ (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
3.	**ip pim spt-threshold** {*kbps* | **infinity**} [**group-list** *access-list-number*]	Specify the threshold that must be reached before moving to shortest-path tree (spt). ■ For *kbps*, specify the traffic rate in kilobits per second. The default is 0 kbps. **Note:** Because of switch hardware limitations, 0 kbps is the only valid entry even though the range is 0 to 4294967. ■ Specify **infinity** if you want all sources for the specified group to use the shared tree, never switching to the source tree. ■ (Optional) For **group-list** *access-list-number*, specify the access list created in Step 2. If the value is 0 or if the group-list is not used, the threshold applies to all groups.
4.	**end**	Return to privileged EXEC mode.
5.	**show running-config**	Verify your entries.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no ip pim spt-threshold** {*kbps* | **infinity**} global configuration command.

EXAMPLE

The following example shows how to set a threshold of 4 kbps. If the traffic rate exceeds this threshold, the traffic to a group from a source causes the router to switch to the shortest path tree to that source.

```
Switch# configure terminal
Switch(config)# ip pim spt-threshold 4
```

Modifying the PIM Router-Query Message Interval

PIM routers and multilayer switches send PIM router-query messages to find which device will be the DR for each LAN segment (subnet). The DR is responsible for sending IGMP host-query messages to all hosts on the directly connected LAN.

With PIM DM operation, the DR has meaning only if IGMPv1 is in use. IGMPv1 does not have an IGMP querier election process, so the elected DR functions as the IGMP querier. With PIM SM operation, the DR is the device that is directly connected to the multicast source. It sends PIM register messages to notify the RP that multicast traffic from a source needs to be forwarded down the shared tree. In this case, the DR is the device with the highest IP address.

Follow this procedure to modify the router-query message interval. This procedure is optional.

BEFORE YOU BEGIN

Review the Information About PIM, page 719.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
3.	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
4.	**ip pim query-interval** *seconds*	Configure the frequency at which the switch sends PIM router-query messages. The default is 30 seconds. The range is 1 to 65535.
5.	**end**	Return to privileged EXEC mode.
6.	**show ip igmp interface** [*interface-id*]	Verify your entries.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no ip pim query-interval** [*seconds*] interface configuration command.

EXAMPLE

The following example shows how to set the PIM hello interval to 45 seconds:

```
interface FastEthernet0/1
 ip pim query-interval 45
```

Configuring Optional IGMP Features

This section includes the following topics:

- Default IGMP Configuration, page 755
- Configuring the Switch as a Member of a Group, page 755 (optional)
- Controlling Access to IP Multicast Groups, page 756 (optional)

- Changing the IGMP Version, page 757 (optional)
- Modifying the IGMP Host-Query Message Interval, page 758 (optional)
- Changing the IGMP Query Timeout for IGMPv2, page 760 (optional)
- Changing the Maximum Query Response Time for IGMPv2, page 761 (optional)
- Configuring the Switch as a Statically Connected Member, page 762 (optional)

Default IGMP Configuration

Feature	Default Setting
Multilayer switch as a member of a multicast group	No group memberships are defined.
Access to multicast groups	All groups are allowed on an interface.
IGMP version	Version 2 on all interfaces.
IGMP host-query message interval	60 seconds on all interfaces.
IGMP query timeout	60 seconds on all interfaces.
IGMP maximum query response time	10 seconds on all interfaces.
Multilayer switch as a statically connected member	Disabled.

Configuring the Switch as a Member of a Group

You can configure the switch as a member of a multicast group and discover multicast reachability in a network. If all the multicast-capable routers and multilayer switches that you administer are members of a multicast group, pinging that group causes all these devices to respond. The devices respond to IGMP echo-request packets addressed to a group of which they are members. Another example is the multicast trace-route tools provided in the software.

This procedure is optional.

BEFORE YOU BEGIN

Caution: Performing this procedure might impact the CPU performance because the CPU will receive all data traffic for the group address.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
3.	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
4.	**ip igmp join-group** *group-address*	Configure the switch to join a multicast group. By default, no group memberships are defined. For *group-address*, specify the multicast IP address in dotted decimal notation.
5.	**end**	Return to privileged EXEC mode.
6.	**show ip igmp interface** [*interface-id*]	Verify your entries.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To cancel membership in a group, use the **no ip igmp join-group** *group-address* interface configuration command.

EXAMPLE

This example shows how to enable the switch to join multicast group 255.2.2.2:

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip igmp join-group 255.2.2.2
```

Controlling Access to IP Multicast Groups

The switch sends IGMP host-query messages to find which multicast groups have members on attached local networks. The switch then forwards to these group members all packets addressed to the multicast group. You can place a filter on each interface to restrict the multicast groups that hosts on the subnet serviced by the interface can join.

This procedure is optional.

BEFORE YOU BEGIN

Review the Information About IGMP, page 718.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
3.	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
4.	**ip igmp access-group** *access-list-number*	Specify the multicast groups that hosts on the subnet serviced by an interface can join. By default, all groups are allowed on an interface. For *access-list-number*, specify an IP standard access list number. The range is 1 to 99.
5.	**exit**	Return to global configuration mode.
6.	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]	Create a standard access list. ■ For *access-list-number*, specify the access list created in Step 3. ■ The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. ■ For *source*, specify the multicast group that hosts on the subnet can join. ■ (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
7.	**end**	Return to privileged EXEC mode.
8.	**show ip igmp interface** [*interface-id*]	Verify your entries.
9.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable groups on an interface, use the **no ip igmp access-group** interface configuration command.

EXAMPLE

This example shows how to configure hosts attached to a port as able to join only group 255.2.2.2:

```
Switch(config)# access-list 1 255.2.2.2 0.0.0.0
Switch(config-if)# interface gigabitethernet0/1
Switch(config-if)# ip igmp access-group 1
```

Changing the IGMP Version

By default, the switch uses IGMP Version 2, which provides features such as the IGMP query timeout and the maximum query response time.

All systems on the subnet must support the same version. The switch does not automatically detect Version 1 systems and switch to Version 1. You can mix Version 1 and Version 2 hosts on the subnet because Version 2 routers or switches always work correctly with IGMPv1 hosts.

Configure the switch for Version 1 if your hosts do not support Version 2.

This procedure is optional.

BEFORE YOU BEGIN

Review the Information About IGMP, page 718.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
3.	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
4.	**ip igmp version** {**1** | **2**}	Specify the IGMP version that the switch uses. **Note:** If you change to Version 1, you cannot configure the **ip igmp query-interval** or the **ip igmp query-max-response-time** interface configuration commands.
5.	**end**	Return to privileged EXEC mode.
6.	**show ip igmp interface** [*interface-id*]	Verify your entries.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no ip igmp version** interface configuration command.

EXAMPLE

The following example configures the router to use IGMP Version 2:

```
ip igmp version 2
```

Modifying the IGMP Host-Query Message Interval

The switch periodically sends IGMP host-query messages to discover which multicast groups are present on attached networks. These messages are sent to the all-hosts multicast group (224.0.0.1) with a time-to-live (TTL) of 1. The switch sends host-query messages to refresh its knowledge of memberships present on the network. If, after some number of queries, the software discovers that no local hosts are members of a multicast group, the software stops forwarding multicast packets to the local network from remote origins for that group and sends a prune message upstream toward the source.

The switch elects a PIM designated router (DR) for the LAN (subnet). The DR is the router or multilayer switch with the highest IP address for IGMPv2. For IGMPv1, the DR is elected according to the multicast routing protocol that runs on the LAN. The designated router is responsible for sending IGMP host-query messages to all hosts on the LAN. In sparse mode, the designated router also sends PIM register and PIM join messages toward the RP router.

This procedure is optional.

BEFORE YOU BEGIN

We recommend that you do not modify the IGMP query interval and IGMP querier timeout values. However, if you configure the appropriate commands to change the query interval and querier timeout default values, the following conditions apply:

- If you use the **ip igmp query-interval** command to configure the query interval, the timeout value is automatically adjusted to two times the query interval; the adjusted timeout value, however, is not reflected in the interface configuration.

Note: To confirm that the timeout value adjusted to two times the modified query interval, use the **show ip igmp interface** command to display the query interval and timeout values being used for the interface.

- Conversely, if you use the **ip igmp querier-timeout** command to configure the timeout value, the query interval does not automatically adjust to half of the modified timeout value, so it is possible to override the default timeout period of two times the query interval. If you must configure the timeout period, we recommend that you configure the timeout value in proportion to the query interval value.
- The query interval must be greater than the IGMP maximum query response time. Use the **ip igmp query-max-response-time** command to change the maximum query response time value from the default (10 seconds) to a specified length of time, if required.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
3.	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
4.	**ip igmp query-interval** *seconds*	Configure the frequency at which the designated router sends IGMP host-query messages. By default, the designated router sends IGMP host-query messages every 60 seconds to keep the IGMP overhead very low on hosts and networks. The range is 1 to 65535.
5.	**end**	Return to privileged EXEC mode.
6.	**show ip igmp interface** [*interface-id*]	Verify your entries.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no ip igmp query-interval** interface configuration command.

EXAMPLE

The following example shows how to configure the switch to wait 240 seconds from the time it received the last query before it triggers the IGMP election process. In this example, the timeout period is manually modified in proportion to the IGMP query interval by using the **ip igmp querier-timeout** command.

```
interface GigabitEthernet1/17
 ip igmp query-interval 120
 ip igmp querier-timeout 240
```

The following example shows how to configure the switch to wait 250 seconds from the time it received the last query until the time that it triggers the IGMP election process. When the timeout value is explicitly configured, the query interval does not automatically adjust. Because the query interval was not explicitly configured to change the default value (60 seconds), the default timeout period of two times the query interval, or 120 seconds, is overridden by the specified value.

```
interface GigabitEthernet0/1
 ip igmp querier-timeout 250
```

Changing the IGMP Query Timeout for IGMPv2

If you are using IGMPv2, you can specify the period of time before the switch takes over as the querier for the interface. By default, the switch waits twice the query interval controlled by the **ip igmp query-interval** interface configuration command. After that time, if the switch has received no queries, it becomes the querier.

You can configure the query interval by entering the **show ip igmp interface** *interface-id* privileged EXEC command. This procedure is optional.

BEFORE YOU BEGIN

We recommend that you do not modify the IGMP query interval and IGMP querier timeout values. However, if you configure the appropriate commands to change the query interval and querier timeout default values, the following conditions apply:

- If you use the **ip igmp query-interval** command to configure the query interval, the timeout value is automatically adjusted to two times the query interval; the adjusted timeout value, however, is not reflected in the interface configuration.

Note: To confirm that the timeout value adjusted to two times the modified query interval, use the **show ip igmp interface** command to display the query interval and timeout values being used for the interface.

- Conversely, if you use the **ip igmp querier-timeout** command to configure the timeout value, the query interval does not automatically adjust to half of the modified timeout value, so it is possible to override the default timeout period of two times the query interval. If you must configure the timeout period, we recommend that you configure the timeout value in proportion to the query interval value.
- The query interval must be greater than the IGMP maximum query response time. Use the **ip igmp query-max-response-time** command to change the maximum query response time value from the default (10 seconds) to a specified length of time, if required.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
3.	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
4.	**ip igmp querier-timeout** *seconds*	Specify the IGMP query timeout. The default is 60 seconds (twice the query interval). The range is 60 to 300.
5.	**end**	Return to privileged EXEC mode.
6.	**show ip igmp interface** [*interface-id*]	Verify your entries.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no ip igmp querier-timeout** interface configuration command.

EXAMPLE

The following example shows how to configure the switch to wait 240 seconds from the time it received the last query before it triggers the IGMP election process. In this example, the timeout period is manually modified in proportion to the IGMP query interval by using the **ip igmp querier-timeout** command.

```
interface GigabitEthernet1/17
 ip igmp query-interval 120
 ip igmp querier-timeout 240
```

The following example shows how to configure the switch to wait 250 seconds from the time it received the last query until the time that it triggers the IGMP election process. When the timeout value is explicitly configured, the query interval does not automatically adjust. Because the query interval was not explicitly configured to change the default value (60 seconds), the default timeout period of two times the query interval, or 120 seconds, is overridden by the specified value.

```
interface GigabitEthernet0/1
 ip igmp querier-timeout 250
```

Changing the Maximum Query Response Time for IGMPv2

If you are using IGMPv2, you can change the maximum query response time advertised in IGMP queries. The maximum query response time enables the switch to quickly detect that there are no more directly connected group members on a LAN. Decreasing the value enables the switch to prune groups faster.

This procedure is optional.

BEFORE YOU BEGIN

The query interval (see the Modifying the IGMP Host-Query Message Interval, page 758) must be greater than the IGMP maximum query response time.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
3.	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
4.	**ip igmp query-max-response-time** *seconds*	Change the maximum query response time advertised in IGMP queries. The default is 10 seconds. The range is 1 to 25.
5.	**end**	Return to privileged EXEC mode.
6.	**show ip igmp interface** [*interface-id*]	Verify your entries.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no ip igmp query-max-response-time** interface configuration command.

EXAMPLE

The following example configures a maximum response time of 8 seconds:

```
ip igmp query-max-response-time 8
```

Configuring the Switch as a Statically Connected Member

Sometimes there is either no group member on a network segment or a host cannot report its group membership by using IGMP. However, you might want multicast traffic to go to that network segment. These are ways to pull multicast traffic down to a network segment:

- Use the **ip igmp join-group** interface configuration command. With this method, the switch accepts the multicast packets in addition to forwarding them. Accepting the multicast packets prevents the switch from fast switching.
- Use the **ip igmp static-group** interface configuration command. With this method, the switch does not accept the packets itself, but only forwards them. This method enables fast switching. The outgoing interface appears in the IGMP cache, but the switch itself is not a member, as evidenced by lack of an *L* (local) flag in the multicast route entry.

This procedure is optional.

BEFORE YOU BEGIN

If you configure the **ip igmp join-group** command for the same group address as the **ip igmp static-group** command, the **ip igmp join-group** command takes precedence, and the group behaves like a locally joined group.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
3.	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
4.	**ip igmp static-group** *group-address*	Configure the switch as a statically connected member of a group. By default, this feature is disabled.
5.	**end**	Return to privileged EXEC mode.
6.	**show ip igmp interface** [*interface-id*]	Verify your entries.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the switch as a member of the group, use the **no ip igmp static-group** *group-address* interface configuration command.

EXAMPLE

The following example shows how to configure group address 239.100.100.101 on Ethernet interface 0:

```
interface ethernet 0
 ip igmp static-group 239.100.100.101
```

Configuring Optional Multicast Routing Features

This section includes the following topics:

- Configuring Session Directory Announcement Support, page 763 (optional)—for MBONE multimedia conference session and set up
- Configuring an IP Multicast Boundary, page 765 (optional)—to control bandwidth utilization.

Configuring Session Directory Announcement Support

The MBONE (multicast backbone of the Internet) is the small subset of Internet routers and hosts that are interconnected and capable of forwarding IP multicast traffic. Other multimedia content is often broadcast over the MBONE. Before you can join a multimedia session, you need to know what multicast group address and port are being used for the session, when the session is going to be active, and what sort of applications (audio, video, and so forth) are required on your workstation. The MBONE Session Directory Version 2 (sdr) tool provides this information. This freeware application can be downloaded from several sites on the World Wide Web, one of which is http://www.video.ja.net/mice/index.html.

SDR is a multicast application that listens to a well-known multicast group address and port for Session Announcement Protocol (SAP) multicast packets from SAP clients, which announce their conference sessions. These SAP packets contain a session description, the time the session is active, its IP multicast group addresses, media format, contact person, and other information about the advertised multimedia session. The information in the SAP packet is displayed in the SDR Session Announcement window.

Enabling Listening to Session Directory Announcements

By default, the switch does not listen to session directory advertisements. Follow this procedure to enable the switch to join the default session directory group (224.2.127.254) on the interface and listen to session directory advertisements. This procedure is optional.

BEFORE YOU BEGIN

Enable multicast routing on the interface as described in the Configuring Basic Multicast Routing, page 731.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Specify the interface on which the well-known session directory groups can receive and store session announcements, and enter interface configuration mode.
3.	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
4.	**ip sap listen**	Enable the switch to listen to session directory announcements.
5.	**end**	Return to privileged EXEC mode.
6.	**show running-config**	Verify your entries.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable listening to session directory announcements, use the **no ip sap listen** interface configuration command.

EXAMPLE

The following example shows how to enable the switch to listen to session directory announcements:

```
ip routing
interface loopback 0
 ip address 10.0.0.51 255.255.255.0
 ip pim sparse-dense mode
 ip sap listen
```

Limiting How Long an SAP Cache Entry Exists

You can limit how long an SAP entry remains active so that if a source stops advertising SAP information, old advertisements are not needlessly kept. This procedure is optional.

BEFORE YOU BEGIN

Setting the cache timeout to a value less than 30 minutes is not recommended.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip sap cache-timeout** *minutes*	Limit how long an SAP cache entry stays active in the cache. By default, session announcements remain for 1440 minutes (24 hours) in the cache. For *minutes*, the range is 1 to 4294967295.
3.	**end**	Return to privileged EXEC mode.
4.	**show running-config**	Verify your entries.
5.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no ip sap cache-timeout** global configuration command. To delete the entire cache, use the **clear ip sap** privileged EXEC command.

To display the session directory cache, use the **show ip sap** privileged EXEC command.

EXAMPLE

The following example causes SAP cache entries to remain in the cache for 30 minutes:

```
ip sap cache-timeout 30
```

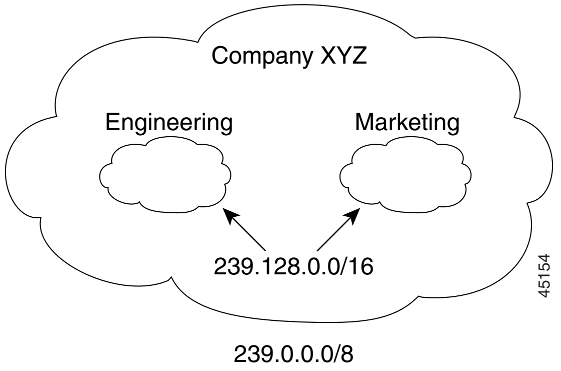
Configuring an IP Multicast Boundary

Administratively-scoped boundaries can be used to limit the forwarding of multicast traffic outside of a domain or subdomain. This approach uses a special range of multicast addresses, called *administratively-scoped addresses*, as the boundary mechanism. If you configure an administratively-scoped boundary on a routed interface, multicast traffic whose multicast group addresses fall in this range cannot enter or exit this interface, thereby providing a firewall for multicast traffic in this address range.

Note: Multicast boundaries and TTL thresholds control the scoping of multicast domains; however, TTL thresholds are not supported by the switch. You should use multicast boundaries instead of TTL thresholds to limit the forwarding of multicast traffic outside of a domain or a subdomain.

Figure 93 on page 766 shows that Company XYZ has an administratively-scoped boundary set for the multicast address range 239.0.0.0/8 on all routed interfaces at the perimeter of its network. This boundary prevents any multicast traffic in the range 239.0.0.0 through 239.255.255.255 from entering or leaving the network. Similarly, the engineering and marketing departments have an administratively-scoped boundary of 239.128.0.0/16 around the perimeter of their networks. This boundary prevents multicast traffic in the range of 239.128.0.0 through 239.128.255.255 from entering or leaving their respective networks.

Figure 93 Administratively-Scoped Boundaries



You can define an administratively-scoped boundary on a routed interface for multicast group addresses. A standard access list defines the range of addresses affected. When a boundary is defined, no multicast data packets are allowed to flow across the boundary from either direction. The boundary allows the same multicast group address to be reused in different administrative domains.

The IANA has designated the multicast address range 239.0.0.0 to 239.255.255.255 as the administratively-scoped addresses. This range of addresses can then be reused in domains administered by different organizations. The addresses would be considered local, not globally unique.

This procedure is optional.

BEFORE YOU BEGIN

Enable multicast routing on the interface as described in the Configuring Basic Multicast Routing, page 731.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]	Create a standard access list, repeating the command as many times as necessary. ■ For *access-list-number*, the range is 1 to 99. ■ The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. ■ For *source*, enter the number of the network or host from which the packet is being sent. ■ (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
3.	**interface** *interface-id*	Specify the interface to be configured, and enter interface configuration mode.
4.	**no shutdown**	Enable the port, if necessary. By default, UNIs and ENIs are disabled, and NNIs are enabled.
5.	**ip multicast boundary** *access-list-number*	Configure the boundary, specifying the access list you created in Step 2.
6.	**end**	Return to privileged EXEC mode.
7.	**show running-config**	Verify your entries.
8.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the boundary, use the **no ip multicast boundary** interface configuration command.

EXAMPLE

This example shows how to set up a boundary for all administratively-scoped addresses:

```
Switch(config)# access-list 1 deny 239.0.0.0 0.255.255.255
Switch(config)# access-list 1 permit 224.0.0.0 15.255.255.255
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip multicast boundary 1
```

Verifying Configuration

This section includes the following topics:

- Clearing Caches, Tables, and Databases, page 768
- Displaying System and Network Statistics, page 768
- Monitoring IP Multicast Routing, page 769

Clearing Caches, Tables, and Databases

You can remove all contents of a particular cache, table, or database. Clearing a cache, table, or database might be necessary when the contents of the particular structure are or suspected to be invalid.

Command	Purpose
clear ip igmp group [*group-name* | *group-address* | *interface*]	Delete entries from the IGMP cache.
clear ip mroute {***** | *group* [*source*]}	Delete entries from the IP multicast routing table.
clear ip pim auto-rp *rp-address*	Clear the Auto-RP cache.
clear ip sdr [*group-address* | **"***session-name***"**]	Delete the Session Directory Protocol Version 2 cache or an sdr cache entry.

Displaying System and Network Statistics

You can display specific statistics, such as the contents of IP routing tables, caches, and databases.

Note: This release does not support per-route statistics.

You can display information to learn resource utilization and solve network problems. You can also display information about node reachability and discover the routing path your device's packets are taking through the network.

Command	Purpose
ping [*group-name* | *group-address*]	Send an ICMP Echo Request to a multicast group address.
show ip igmp groups [*group-name* | *group-address* | *type number*]	Display the multicast groups that are directly connected to the switch and that were learned through IGMP.
show ip igmp interface [*type number*]	Display multicast-related information about an interface.
show ip mcache [*group* [*source*]]	Display the contents of the IP fast-switching cache.
show ip mpacket [*source-address* | *name*] [*group-address* | *name*] [**detail**]	Display the contents of the circular cache-header buffer.
show ip mroute [*group-name* | *group-address*] [*source*] [**summary**] [**count**] [**active** *kbps*]	Display the contents of the IP multicast routing table.
show ip pim interface [*type number*] [**count**]	Display information about interfaces configured for PIM.
show ip pim neighbor [*type number*]	List the PIM neighbors discovered by the switch.
show ip pim rp [*group-name* | *group-address*]	Display the RP routers associated with a sparse-mode multicast group.
show ip rpf {*source-address* | *name*}	Display how the switch is doing Reverse-Path Forwarding (that is, from the unicast routing table or static mroutes).
show ip sap [*group* | **"***session-name***"** | **detail**]	Display the Session Directory Protocol Version 2 cache.

Monitoring IP Multicast Routing

Command	Purpose
mrinfo [*hostname* | *address*] [*source-address* | *interface*]	Query a multicast router or multilayer switch about which neighboring multicast devices are peering with it.
mstat *source* [*destination*] [*group*]	Display IP multicast packet rate and loss information.
mtrace *source* [*destination*] [*group*]	Trace the path from a source to a destination branch for a multicast distribution tree for a given group.

Configuration Example

This example enables IP multicast distributed switching and specifies the PIM mode:

```
Switch# configure terminal
Switch(config)# ip multicast-routing distributed
Switch(config)# interface Gigabitethernet 1/0/0
Switch(config-if)# ip pim sparse-dense-mode
Switch(config-if)# end
```

In this example, IP multicast routing is enabled, Switch A PIM uplink port 25 is configured as a routed uplink port with **spare-dense-mode enabled.** PIM stub routing is enabled on the VLAN 100 interfaces and on Gigabit Ethernet port 20 in Figure 88 on page 721:

```
Switch(config)# ip multicast-routing distributed
Switch(config)# interface GigabitEthernet0/25
Switch(config-if)# no switchport
Switch(config-if)# ip address 3.1.1.2 255.255.255.0
Switch(config-if)# ip pim sparse-dense-mode
Switch(config-if)# exit
Switch(config)# interface vlan100
Switch(config-if)# ip pim passive
Switch(config-if)# exit
Switch(config)# interface GigabitEthernet0/20
Switch(config-if)# ip pim passive
Switch(config-if)# exit
Switch(config)# interface vlan100
Switch(config-if)# ip address 100.1.1.1 255.255.255.0
Switch(config-if)# ip pim passive
Switch(config-if)# exit
Switch(config)# interface GigabitEthernet0/20
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.1.1.1 255.255.255.0
Switch(config-if)# ip pim passive
Switch(config-if)# end
```

To verify that PIM stub is enabled for each interface, use the **show ip pim interface** privileged EXEC command:

```
Switch# show ip pim interface
Address Interface Ver/ Nbr Query DR DR
Mode Count Intvl Prior
3.1.1.2 GigabitEthernet0/25 v2/SD 1 30 1 3.1.1.2
100.1.1.1 Vlan100 v2/P 0 30 1 100.1.1.1
10.1.1.1 GigabitEthernet0/20 v2/P 0 30 1 10.1.1.1
```

The following example shows how to configure a device (running IGMPv3) for SSM:

```
ip multicast-routing
ip pim ssm default
!
interface GigabitEthernet3/1/0
 ip address 172.21.200.203 255.255.255.0
 description backbone interface
    ip pim sparse-mode
!
interface GigabitEthernet3/2/0
    ip address 131.108.1.2 255.255.255.0
    ip pim sparse-mode
    description ethernet connected to hosts
    ip igmp version 3
!
```

The following example shows how to enable static SSM mapping. In this example, the router is configured to statically map groups that match ACL 11 to source address 172.16.8.11 and to statically map groups that match ACL 10 to source address 172.16.8.10.

```
Switch(config)# ip igmp ssm-map enable
Switch(config)# ip igmp ssm-map static 11 172.16.8.11
Switch(config)# ip igmp ssm-map static 10 172.16.8.10
Switch(config)# end
```

The following example shows how to configure DNS-based SSM mapping:

```
Switch(config)# ip igmp ssm-map enable
Switch(config)# ip name-server 10.0.0.0
Switch(config)# end
```

The following example shows how to configure group address 239.1.2.1 to use SSM mapping for statically forwarded groups on Ethernet interface 0:

```
interface ethernet 0
 ip igmp static-group 239.1.2.1 source ssm-map
```

This example shows how to configure the address of the RP to 147.106.6.22 for multicast group 225.2.2.2 only:

```
Switch(config)# access-list 1 permit 225.2.2.2 0.0.0.0
Switch(config)# ip pim rp-address 147.106.6.22 1
```

This example shows how to send RP announcements out all PIM-enabled interfaces for a maximum of 31 hops. The IP address of port 1 is the RP. Access list 5 describes the group for which this switch serves as RP:

```
Switch(config)# ip pim send-rp-announce gigabitethernet0/1 scope 31 group-list 5
Switch(config)# access-list 5 permit 224.0.0.0 15.255.255.255
```

This example shows a sample configuration on an Auto-RP mapping agent that is used to prevent candidate RP announcements from being accepted from unauthorized candidate RPs. In this example, the mapping agent accepts candidate RP announcements from only two devices, 172.16.5.1 and 172.16.2.1. The mapping agent accepts candidate RP announcements from these two devices only for multicast groups that fall in the group range of 224.0.0.0 to 239.255.255.255. The mapping agent does not accept candidate RP announcements from any other devices in the network. Furthermore, the mapping agent does not accept candidate RP announcements from 172.16.5.1 or 172.16.2.1 if the announcements are for any groups in the 239.0.0.0 through 239.255.255.255 range. This range is the administratively scoped address range.

```
Switch(config)# ip pim rp-announce-filter rp-list 10 group-list 20
Switch(config)# access-list 10 permit host 172.16.5.1
Switch(config)# access-list 10 permit host 172.16.2.1
Switch(config)# access-list 20 deny 239.0.0.0 0.0.255.255
Switch(config)# access-list 20 permit 224.0.0.0 15.255.255.255
```

The following example configures the interface to be the PIM domain border:

```
interface ethernet 1
ip pim bsr-border
```

This example shows a portion of an IP multicast boundary configuration that denies Auto-RP information:

```
Switch(config)# access-list 1 deny 224.0.1.39
Switch(config)# access-list 1 deny 224.0.1.40
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip multicast boundary 1
```

This example shows how to configure a candidate BSR, which uses the IP address 172.21.24.18 on a port as the advertised BSR address, uses 30 bits as the hash-mask-length, and has a priority of 10:

```
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# ip address 172.21.24.18 255.255.255.0
Switch(config-if)# ip pim sparse-dense-mode
Switch(config-if)# ip pim bsr-candidate gigabitethernet0/2 30 10
```

This example shows how to configure the switch to advertise itself as a candidate RP to the BSR in its PIM domain. Standard access list number 4 specifies the group prefix associated with the RP that has the address identified by a port. That RP is responsible for the groups with the prefix 239.

```
Switch(config)# ip pim rp-candidate gigabitethernet0/2 group-list 4
Switch(config)# access-list 4 permit 239.0.0.0 0.255.255.255
```

The following example shows how to set a threshold of 4 kbps. If the traffic rate exceeds this threshold, the traffic to a group from a source causes the router to switch to the shortest path tree to that source.

```
Switch# configure terminal
Switch(config)# ip pim spt-threshold 4
```

The following example shows how to set the PIM hello interval to 45 seconds:

```
interface FastEthernet0/1
 ip pim query-interval 45
```

This example shows how to enable the switch to join multicast group 255.2.2.2:

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip igmp join-group 255.2.2.2
```

The following example configures the router to use IGMP Version 2:

```
ip igmp version 2
```

The following example shows how to configure the switch to wait 240 seconds from the time it received the last query before it triggers the IGMP election process. In this example, the timeout period is manually modified in proportion to the IGMP query interval by using the **ip igmp querier-timeout** command.

```
interface GigabitEthernet1/17
 ip igmp query-interval 120
 ip igmp querier-timeout 240
```

The following example shows how to configure the switch to wait 250 seconds from the time it received the last query until the time that it triggers the IGMP election process. When the timeout value is explicitly configured, the query interval does not automatically adjust. Because the query interval was not explicitly configured to change the default value (60 seconds), the default timeout period of two times the query interval, or 120 seconds, is overridden by the specified value.

```
interface GigabitEthernet0/1
 ip igmp querier-timeout 250
```

The following example configures a maximum response time of 8 seconds:

```
ip igmp query-max-response-time 8
```

The following example shows how to configure group address 239.100.100.101 on Ethernet interface 0:

```
interface ethernet 0
 ip igmp static-group 239.100.100.101
```

The following example shows how to enable the switch to listen to session directory announcements:

```
ip routing
interface loopback 0
 ip address 10.0.0.51 255.255.255.0
 ip pim sparse-dense mode
 ip sap listen
```

The following example causes SAP cache entries to remain in the cache for 30 minutes:

```
ip sap cache-timeout 30
```

This example shows how to set up a boundary for all administratively-scoped addresses:

```
Switch(config)# access-list 1 deny 239.0.0.0 0.255.255.255
Switch(config)# access-list 1 permit 224.0.0.0 15.255.255.255
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip multicast boundary 1
```

Related Documents

- Cisco IOS IP Multicast Command Reference
- IP Multicast Configuration Guide Library, Cisco IOS Release 15M&T
- Cisco IOS Master Command List, All Releases

Configuring MSDP

This chapter describes how to configure Multicast Source Discovery Protocol (MSDP) on the Cisco Industrial Ethernet Switches, hereafter referred to as *switch*. MSDP connects multiple Protocol-Independent Multicast sparse-mode (PIM-SM) domains.

MSDP is not fully supported in this software release because of a lack of support for Multicast Border Gateway Protocol (MBGP), which works closely with MSDP. However, it is possible to create default peers that MSDP can operate with if MBGP is not running.

To use this feature, the switch must be running the IP services image.

Note: For complete syntax and usage information for the commands used in this chapter, see the documents listed in the Related Documents, page 794.

This chapter includes the following sections:

Information About MSDP

MSDP allows multicast sources for a group to be known to all rendezvous points (RPs) in different domains. Each PIM-SM domain uses its own RPs and does not depend on RPs in other domains. An RP runs MSDP over the Transmission Control Protocol (TCP) to discover multicast sources in other domains.

An RP in a PIM-SM domain has an MSDP peering relationship with MSDP-enabled devices in another domain. The peering relationship occurs over a TCP connection, primarily exchanging a list of sources sending to multicast groups. The TCP connections between RPs are achieved by the underlying routing system. The receiving RP uses the source lists to establish a source path.

The purpose of this topology is to have domains discover multicast sources in other domains. If the multicast sources are of interest to a domain that has receivers, multicast data is delivered over the normal, source-tree building mechanism in PIM-SM. MSDP is also used to announce sources sending to a group. These announcements must originate at the domain’s RP.

MSDP depends heavily on the Border Gateway Protocol (BGP) or MBGP for interdomain operation. We recommend that you run MSDP in RPs in your domain that are RPs for sources sending to global groups to be announced to the Internet.

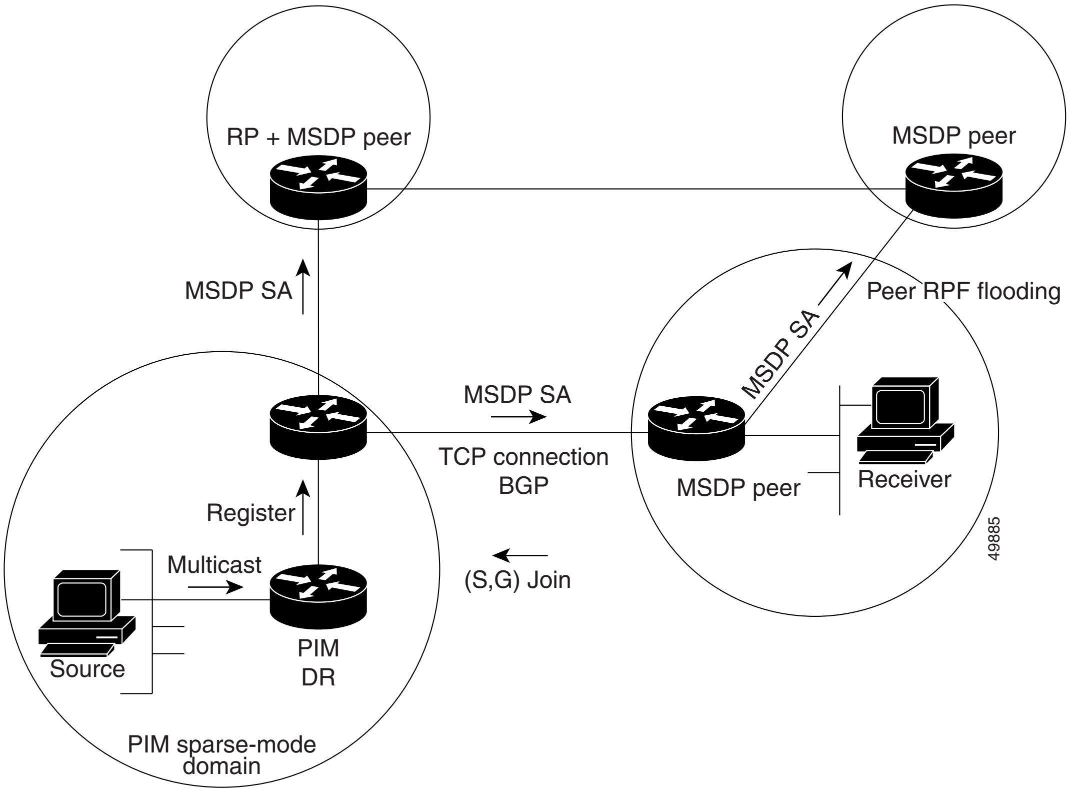
MSDP Operation

Figure 94 on page 774 shows MSDP operating between two MSDP peers. PIM uses MSDP as the standard mechanism to register a source with the RP of a domain. When MSDP is configured, this sequence occurs.

When a source sends its first multicast packet, the first-hop router (*designated router* or RP) directly connected to the source sends a PIM register message to the RP. The RP uses the register message to register the active source and to forward the multicast packet down the shared tree in the local domain. With MSDP configured, the RP also forwards a source-active (SA) message to all MSDP peers. The SA message identifies the source, the group the source is sending to, and the address of the RP or the originator ID (the IP address of the interface used as the RP address), if configured.

Each MSDP peer receives and forwards the SA message away from the originating RP to achieve peer reverse-path flooding (RPF). The MSDP device examines the BGP or MBGP routing table to discover which peer is the next hop toward the originating RP of the SA message. Such a peer is called an *RPF peer* (reverse-path forwarding peer). The MSDP device forwards the message to all MSDP peers other than the RPF peer. For information on how to configure an MSDP peer when BGP and MBGP are not supported, see Configuring a Default MSDP Peer, page 776.

Figure 94 MSDP Running Between RP Peers



If the MSDP peer receives the same SA message from a non-RPF peer toward the originating RP, it drops the message. Otherwise, it forwards the message to all its MSDP peers.

The RP for a domain receives the SA message from an MSDP peer. If the RP has any join requests for the group the SA message describes and if the (*,G) entry exists with a nonempty outgoing interface list, the domain is interested in the group, and the RP triggers an (S,G) join toward the source. After the (S,G) join reaches the source's DR, a branch of the source tree has been built from the source to the RP in the remote domain. Multicast traffic can now flow from the source across the source tree to the RP and then down the shared tree in the remote domain to the receiver.

MSDP Benefits

MSDP has these benefits:

- It breaks up the shared multicast distribution tree. You can make the shared tree local to your domain. Your local members join the local tree, and join messages for the shared tree never need to leave your domain.
- PIM sparse-mode domains can rely only on their own RPs, decreasing reliance on RPs in another domain. This increases security because you can prevent your sources from being known outside your domain.
- Domains with only receivers can receive data without globally advertising group membership.
- Global source multicast routing table state is not required, saving memory.

Prerequisites

- The switch is running the IP services image.
- You have enabled IP multicast routing and configured PIM for the networks where you want to configure MSDP.

Guidelines and Limitations

MSDP is not fully supported in this software release because of a lack of support for Multicast Border Gateway Protocol (MBGP), which works closely with MSDP. However, it is possible to create default peers that MSDP can operate with if MBGP is not running.

Default Settings

MSDP is not enabled, and no default MSDP peer exists.

Configuring MSDP

This section includes the following topics:

- Configuring a Default MSDP Peer, page 776 (required)
- Caching Source-Active State, page 778 (optional)
- Requesting Source Information from an MSDP Peer, page 780 (optional)
- Controlling Source Information that Your Switch Originates, page 780 (optional)
- Controlling Source Information that Your Switch Forwards, page 784 (optional)
- Controlling Source Information that Your Switch Receives, page 786 (optional)
- Configuring an MSDP Mesh Group, page 789 (optional)
- Shutting Down an MSDP Peer, page 790 (optional)
- Including a Bordering PIM Dense-Mode Region in MSDP, page 790 (optional)
- Configuring an Originating Address other than the RP Address, page 791 (optional)

Configuring a Default MSDP Peer

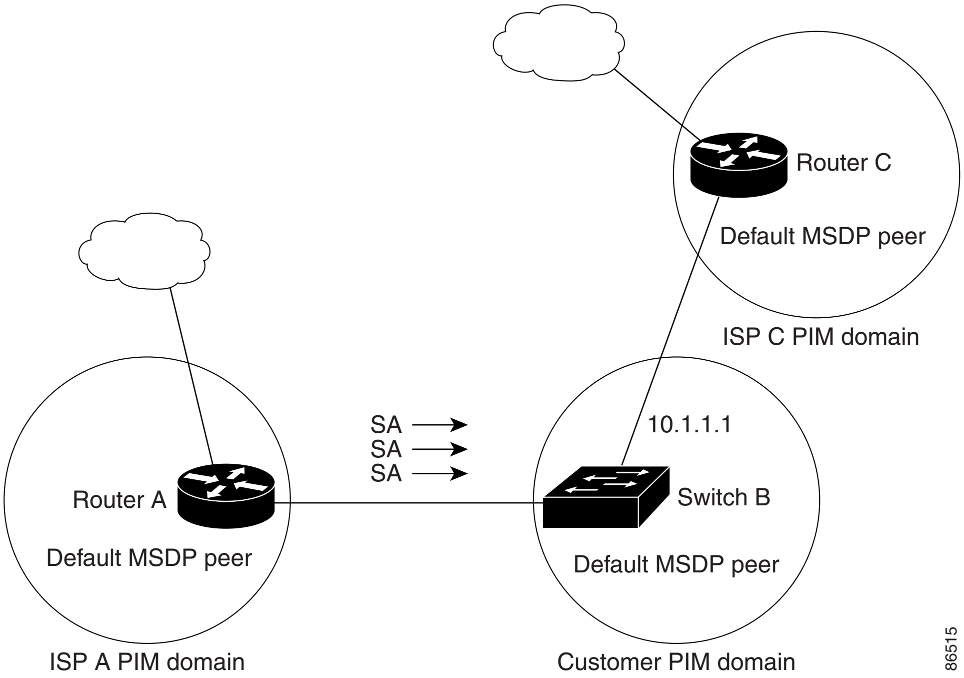
In this software release, because BGP and MBGP are not supported, you cannot configure an MSDP peer on the local switch by using the **ip msdp peer** global configuration command. Instead, you define a default MSDP peer (by using the **ip msdp default-peer** global configuration command) from which to accept all SA messages for the switch. The default MSDP peer must be a previously configured MSDP peer. Configure a default MSDP peer when the switch is not BGP- or MBGP-peering with an MSDP peer. If a single MSDP peer is configured, the switch always accepts all SA messages from that peer.

Figure 95 on page 776 shows a network in which default MSDP peers might be used. In Figure 95 on page 776, a customer who owns Switch B is connected to the Internet through two Internet service providers (ISPs), one owning Router A and the other owning Router C. They are not running BGP or MBGP between them. To learn about sources in the ISP's domain or in other domains, Switch B at the customer site identifies Router A as its default MSDP peer. Switch B advertises SA messages to both Router A and Router C but accepts SA messages only from Router A or only from Router C. If Router A is first in the configuration file, it is used if it is running. If Router A is not running, only then does Switch B accept SA messages from Router C. This is the default behavior without a prefix list.

If you specify a prefix list, the peer is a default peer only for the prefixes in the list. You can have multiple active default peers when you have a prefix list associated with each. When you do not have any prefix lists, you can configure multiple default peers, but only the first one is the active default peer as long as the router has connectivity to this peer and the peer is alive. If the first configured peer fails or the connectivity to this peer fails, the second configured peer becomes the active default, and so on.

The ISP probably uses a prefix list to define which prefixes it accepts from the customer's router.

Figure 95 Default MSDP Peer Network



Follow this procedure to specify a default MSDP peer. This procedure is required.

BEFORE YOU BEGIN

An MSDP default peer must be a previously configured MSDP peer. Before configuring a default MSDP peer, you must first configure an MSDP peer.

DETAILED STEPS

Table 0-1

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip msdp default-peer** *ip-address* | *name* [**prefix-list** *list*]	Define a default peer from which to accept all MSDP SA messages. ■ For *ip-address* | *name*, enter the IP address or Domain Name System (DNS) server name of the MSDP default peer. ■ (Optional) For **prefix-list** *list*, enter the list name that specifies the peer to be the default peer only for the listed prefixes. You can have multiple active default peers when you have a prefix list associated with each. When you enter multiple **ip msdp default-peer** commands with the **prefix-list** keyword, you use all the default peers at the same time for different RP prefixes. This syntax is typically used in a service provider cloud that connects stub site clouds. When you enter multiple **ip msdp default-peer** commands without the **prefix-list** keyword, a single active peer accepts all SA messages. If that peer fails, the next configured default peer accepts all SA messages. This syntax is typically used at a stub site.
3.	**ip prefix-list** *name* [**description** *string*] | **seq** *number* {**permit** | **deny**} *network length*	(Optional) Create a prefix list using the name specified in Step 2. ■ (Optional) For **description** *string*, enter a description of up to 80 characters to describe this prefix list. ■ For **seq** *number,* enter the sequence number of the entry. The range is 1 to 4294967294. ■ The **deny** keyword denies access to matching conditions. ■ The **permit** keyword permits access to matching conditions. ■ For *network length*, specify the network number and length (in bits) of the network mask that is permitted or denied.
4.	**ip msdp description** {*peer-name* | *peer-address*} *text*	(Optional) Configure a description for the specified peer to make it easier to identify in a configuration or in **show** command output. By default, no description is associated with an MSDP peer.
5.	**end**	Return to privileged EXEC mode.
6.	**show running-config**	Verify your entries.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the default peer, use the **no ip msdp default-peer** *ip-address* | *name* global configuration command.

EXAMPLE

This example shows a partial configuration of Router A and Router C in Figure 95 on page 776. Each of these ISPs have more than one customer (like the customer in Figure 95 on page 776) who use default peering (no BGP or MBGP). In that case, they might have similar configurations. That is, they accept SAs only from a default peer if the SA is permitted by the corresponding prefix list.

Router A

```
Router(config)# ip msdp default-peer 10.1.1.1
Router(config)# ip msdp default-peer 10.1.1.1 prefix-list site-a
Router(config)# ip prefix-list site-b permit 10.0.0.0/1
```

Router C

```
Router(config)# ip msdp default-peer 10.1.1.1 prefix-list site-a
Router(config)# ip prefix-list site-b permit 10.0.0.0/1
```

Caching Source-Active State

By default, the switch does not cache source/group pairs from received SA messages. When the switch forwards the MSDP SA information, it does not store it in memory. Therefore, if a member joins a group soon after a SA message is received by the local RP, that member needs to wait until the next SA message to hear about the source. This delay is known as join latency.

If you want to sacrifice some memory in exchange for reducing the latency of the source information, you can configure the switch to cache SA messages. This procedure is optional.

Note: An alternative to this command is the **ip msdp sa-request** global configuration command, which causes the switch to send an SA request message to the MSDP peer when a new member for a group becomes active. For more information, see the next section.

DETAILED STEPS

Table 0-2

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip msdp cache-sa-state** [**list** *access-list-number*]	Enable the caching of source/group pairs (create an SA state). Those pairs that pass the access list are cached. For **list** *access-list-number*, the range is 100 to 199.
3.	**access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard*	Create an IP extended access list, repeating the command as many times as necessary. ■ For *access-list-number*, the range is 100 to 199. Enter the same number created in Step 2. ■ The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. ■ For *protocol*, enter **ip** as the protocol name. ■ For *source*, enter the number of the network or host from which the packet is being sent. ■ For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. ■ For *destination*, enter the number of the network or host to which the packet is being sent. ■ For *destination-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the destination. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
4.	**end**	Return to privileged EXEC mode.
5.	**show running-config**	Verify your entries.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting (no SA state is created), use the **no ip msdp cache-sa-state** global configuration command.

EXAMPLE

This example shows how to enable the cache state for all sources in 171.69.0.0/16 sending to groups 224.2.0.0/16:

```
Switch(config)# ip msdp cache-sa-state 100
Switch(config)# access-list 100 permit ip 171.69.0.0 0.0.255.255 224.2.0.0 0.0.255.255
```

Requesting Source Information from an MSDP Peer

Local RPs can send SA requests and get immediate responses for all active sources for a given group. By default, the switch does not send any SA request messages to its MSDP peers when a new member joins a group and wants to receive multicast traffic. The new member waits to receive the next periodic SA message.

If you want a new member of a group to learn the active multicast sources in a connected PIM sparse-mode domain that are sending to a group, configure the switch to send SA request messages to the specified MSDP peer when a new member joins a group. The peer replies with the information in its SA cache. If the peer does not have a cache configured, this command has no result. Configuring this feature reduces join latency but sacrifices memory.

Follow this procedure to configure the switch to send SA request messages to the MSDP peer when a new member joins a group and wants to receive multicast traffic. This procedure is optional.

DETAILED STEPS

Table 0-3

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip msdp sa-request** {*ip-address* | *name*}	Configure the switch to send SA request messages to the specified MSDP peer. For *ip-address* | *name*, enter the IP address or name of the MSDP peer from which the local switch requests SA messages when a new member for a group becomes active. Repeat the command for each MSDP peer that you want to supply with SA messages.
3.	**end**	Return to privileged EXEC mode.
4.	**show running-config**	Verify your entries.
5.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no ip msdp sa-request** {*ip-address* | *name*} global configuration command.

EXAMPLE

This example shows how to configure the switch to send SA request messages to the MSDP peer at 171.69.1.1:

```
Switch(config)# ip msdp sa-request 171.69.1.1
```

Controlling Source Information that Your Switch Originates

You can control the multicast source information that originates with your switch:

- Sources you advertise (based on your sources)
- Receivers of source information (based on knowing the requestor)

For more information, see Redistributing Sources, page 780 and Filtering Source-Active Request Messages, page 782.

Redistributing Sources

SA messages originate on RPs to which sources have registered. By default, any source that registers with an RP is advertised. The *A flag* is set in the RP when a source is registered, which means the source is advertised in an SA unless it is filtered. Follow this procedure to further restrict which registered sources are advertised. This procedure is optional.

Configuring MSDP

BEFORE YOU BEGIN

For best practice information related to configuring MSDP SA message filters, see the Multicast Source Discovery Protocol SA Filter Recommendations tech note.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip msdp redistribute** [**list** *access-list-name*] [**asn** *aspath-access-list-number*] [**route-map** *map*]	Configure which (S,G) entries from the multicast routing table are advertised in SA messages. By default, only sources within the local domain are advertised. ■ (Optional) For **list** *access-list-name*, enter the name or number of an IP standard or extended access list. The range is 1 to 99 for standard access lists and 100 to 199 for extended lists. The access list controls which local sources are advertised and to which groups they send. ■ (Optional) For **asn** *aspath-access-list-number*, enter the IP standard or extended access list number in the range 1 to 199. This access list number must also be configured in the **ip as-path access-list** command. ■ (Optional) For **route-map** *map*, enter the IP standard or extended access list number in the range 1 to 199. This access list number must also be configured in the **ip as-path access-list** command. The switch advertises (S,G) pairs according to the access list or autonomous system path access list.

	Command	Purpose
3.	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] **or** **access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard*	Create an IP standard access list, repeating the command as many times as necessary. or Create an IP extended access list, repeating the command as many times as necessary. ■ For *access-list-number*, the range is 1 to 99 for standard access lists and 100 to 199 for extended lists. Enter the same number created in Step 2. ■ The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. ■ For *protocol*, enter **ip** as the protocol name. ■ For *source*, enter the number of the network or host from which the packet is being sent. ■ For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. ■ For *destination*, enter the number of the network or host to which the packet is being sent. ■ For *destination-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the destination. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
4.	**end**	Return to privileged EXEC mode.
5.	**show running-config**	Verify your entries.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the filter, use the **no ip msdp redistribute** global configuration command.

EXAMPLE

The following example shows how to configure which (S, G) entries from the mroute table are advertised in SA messages originated from AS 64512:

```
Switch(config)# ip msdp redistribute route-map customer-sources
Switch(config)# route-map customer-sources permit
Switch(config)# match as-path 100
Switch(config)# ip as-path access-list 100 permit ^64512$
```

Filtering Source-Active Request Messages

By default, only switches that are caching SA information can respond to SA requests. By default, such a switch honors all SA request messages from its MSDP peers and supplies the IP addresses of the active sources.

However, you can configure the switch to ignore all SA requests from an MSDP peer. You can also honor only those SA request messages from a peer for groups described by a standard access list. If the groups in the access list pass, SA request messages are accepted. All other such messages from the peer for other groups are ignored.

Follow this procedure to configure one of these options. This procedure is optional.

DETAILED STEPS

Table 0-4

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip msdp filter-sa-request** *ip-address* | *name* or **ip msdp filter-sa-request** {*ip-address* | *name*} **list** *access-list-number*	Filter all SA request messages from the specified MSDP peer. or Filter SA request messages from the specified MSDP peer for groups that pass the standard access list. The access list describes a multicast group address. The range for the access-list-number is 1 to 99.
3.	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]	Create an IP standard access list, repeating the command as many times as necessary. ■ For *access-list-number*, the range is 1 to 99. ■ The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. ■ For *source*, enter the number of the network or host from which the packet is being sent. ■ (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
4.	**end**	Return to privileged EXEC mode.
5.	**show running-config**	Verify your entries.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no ip msdp filter-sa-request** {*ip-address* | *name*} global configuration command.

EXAMPLE

This example shows how to configure the switch to filter SA request messages from the MSDP peer at 171.69.2.2. SA request messages from sources on network 192.4.22.0 pass access list 1 and are accepted; all others are ignored.

```
Switch(config)# ip msdp filter sa-request 171.69.2.2 list 1
Switch(config)# access-list 1 permit 192.4.22.0 0.0.0.255
```

Controlling Source Information that Your Switch Forwards

By default, the switch forwards all SA messages it receives to all its MSDP peers. However, you can prevent outgoing messages from being forwarded to a peer by using a filter or by setting a time-to-live (TTL) value. These methods are described in the next sections.

Using a Filter

By creating a filter, you can perform one of these actions:

- Filter all source/group pairs
- Specify an IP extended access list to pass only certain source/group pairs
- Filter based on match criteria in a route map

Follow this procedure to apply a filter. This procedure is optional.

BEFORE YOU BEGIN

For best practice information related to configuring MSDP SA message filters, see the Multicast Source Discovery Protocol SA Filter Recommendations tech note.

Configuring MSDP

DETAILED STEPS

Table 0-5

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip msdp sa-filter out** *ip-address* | *name* or **ip msdp sa-filter out** {*ip-address* | *name*} **list** *access-list-number* or **ip msdp sa-filter out** {*ip-address* | *name*} **route-map** *map-tag*	Filter all SA messages to the specified MSDP peer. or To the specified peer, pass only those SA messages that pass the IP extended access list. The range for the extended *access-list-number* is 100 to 199. If both the **list** and the **route-map** keywords are used, all conditions must be true to pass any (S,G) pair in outgoing SA messages. or To the specified MSDP peer, pass only those SA messages that meet the match criteria in the route map *map-tag*. If all match criteria are true, a **permit** from the route map passes routes through the filter. A **deny** filters routes.
3.	**access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard*	(Optional) Create an IP extended access list, repeating the command as many times as necessary. ■ For *access-list-number*, enter the number specified in Step 2. ■ The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. ■ For *protocol*, enter **ip** as the protocol name. ■ For *source*, enter the number of the network or host from which the packet is being sent. ■ For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. ■ For *destination*, enter the number of the network or host to which the packet is being sent. ■ For *destination-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the destination. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
4.	**end**	Return to privileged EXEC mode.
5.	**show running-config**	Verify your entries.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the filter, use the **no ip msdp sa-filter out** {*ip-address* | *name*} [**list** *access-list-number*] [**route-map** *map-tag*] global configuration command.

EXAMPLE

This example shows how to allow only (S,G) pairs that pass access list 100 to be forwarded in an SA message to the peer named *switch.cisco.com*:

```
Switch(config)# ip msdp peer switch.cisco.com connect-source gigabitethernet0/1
Switch(config)# ip msdp sa-filter out switch.cisco.com list 100
Switch(config)# access-list 100 permit ip 171.69.0.0 0.0.255.255 224.20 0 0.0.255.255
```

Using TTL to Limit the Multicast Data Sent in SA Messages

You can use a TTL value to control what data is encapsulated in the first SA message for every source. Only multicast packets with an IP-header TTL greater than or equal to the *ttl* argument are sent to the specified MSDP peer. For example, you can limit internal traffic to a TTL of 8. If you want other groups to go to external locations, you must send those packets with a TTL greater than 8.

Follow this procedure to establish a TTL threshold. This procedure is optional.

DETAILED STEPS

Table 0-6

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip msdp ttl-threshold** {*ip-address* | *name*} *ttl*	Limit which multicast data is encapsulated in the first SA message to the specified MSDP peer. ■ For *ip-address* | *name*, enter the IP address or name of the MSDP peer to which the TTL limitation applies. ■ For *ttl*, enter the TTL value. The default is 0, which means all multicast data packets are forwarded to the peer until the TTL is exhausted. The range is 0 to 255.
3.	**end**	Return to privileged EXEC mode.
4.	**show running-config**	Verify your entries.
5.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no ip msdp ttl-threshold** {*ip-address* | *name*} global configuration command.

EXAMPLE

The following example shows how to configure a TTL threshold of 8 hops:

```
Switch(config)# ip msdp ttl-threshold 192.168.1.5 8
```

Controlling Source Information that Your Switch Receives

By default, the switch receives all SA messages that its MSDP RPF peers send to it. However, you can control the source information that you receive from MSDP peers by filtering incoming SA messages. In other words, you can configure the switch to not accept them.

You can perform one of these actions:

- Filter all incoming SA messages from an MSDP peer
- Specify an IP extended access list to pass certain source/group pairs
- Filter based on match criteria in a route map

Follow this procedure to apply a filter. This procedure is optional.

BEFORE YOU BEGIN

For best practice information related to configuring MSDP SA message filters, see the Multicast Source Discovery Protocol SA Filter Recommendations tech note.

DETAILED STEPS

Table 0-7

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip msdp sa-filter in** *ip-address* | *name* or **ip msdp sa-filter in** {*ip-address* | *name*} **list** *access-list-number* or **ip msdp sa-filter in** {*ip-address* | *name*} **route-map** *map-tag*	Filter all SA messages from the specified MSDP peer. or From the specified peer, pass only those SA messages that pass the IP extended access list. The range for the extended *access-list-number* is 100 to 199. If both the **list** and the **route-map** keywords are used, all conditions must be true to pass any (S,G) pair in incoming SA messages. or From the specified MSDP peer, pass only those SA messages that meet the match criteria in the route map *map-tag*. If all match criteria are true, a **permit** from the route map passes routes through the filter. A **deny** will filter routes.
3.	**access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard*	(Optional) Create an IP extended access list, repeating the command as many times as necessary. ■ For *access-list-number*, enter the number specified in Step 2. ■ The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. ■ For *protocol*, enter **ip** as the protocol name. ■ For *source*, enter the number of the network or host from which the packet is being sent. ■ For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. ■ For *destination*, enter the number of the network or host to which the packet is being sent. ■ For *destination-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the destination. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
4.	**end**	Return to privileged EXEC mode.
5.	**show running-config**	Verify your entries.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the filter, use the **no ip msdp sa-filter in** {*ip-address* | *name*} [**list** *access-list-number*] [**route-map** *map-tag*] global configuration command.

EXAMPLE

This example shows how to filter all SA messages from the peer named *switch.cisco.com*:

```
Switch(config)# ip msdp peer switch.cisco.com connect-source gigabitethernet0/1
Switch(config)# ip msdp sa-filter in switch.cisco.com
```

Configuring an MSDP Mesh Group

An MSDP mesh group is a group of MSDP speakers that have fully meshed MSDP connectivity among one another. Any SA messages received from a peer in a mesh group are not forwarded to other peers in the same mesh group. Thus, you reduce SA message flooding and simplify peer-RPF flooding. Use the **ip msdp mesh-group** global configuration command when there are multiple RPs within a domain. It is especially used to send SA messages across a domain. You can configure multiple mesh groups (with different names) in a single switch. This procedure is optional.

DETAILED STEPS

Table 0-8

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip msdp mesh-group** *name* {*ip-address* | *name*}	Configure an MSDP mesh group, and specify the MSDP peer belonging to that mesh group. By default, the MSDP peers do not belong to a mesh group. ■ For *name*, enter the name of the mesh group. ■ For *ip-address* | *name*, enter the IP address or name of the MSDP peer to be a member of the mesh group.
3.	**end**	Return to privileged EXEC mode.
4.	**show running-config**	Verify your entries.
5.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.
6.		Repeat this procedure on each MSDP peer in the group.

To remove an MSDP peer from a mesh group, use the **no ip msdp mesh-group** *name* {*ip-address* | *name*} global configuration command.

EXAMPLE

The following example shows how to configure the MSDP peer at address 192.168.1.3 to be a member of the mesh group named internal:

```
Switch(config)# ip msdp mesh-group internal 192.168.1.3
```

Shutting Down an MSDP Peer

If you want to configure many MSDP commands for the same peer and you do not want the peer to become active, you can shut down the peer, configure it, and later bring it up. When a peer is shut down, the TCP connection is terminated and is not restarted. You can also shut down an MSDP session without losing configuration information for the peer. This procedure is optional.

DETAILED STEPS

Table 0-9

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip msdp shutdown** {*peer-name* | *peer address*}	Administratively shut down the specified MSDP peer without losing configuration information. For *peer-name* | *peer address,* enter the IP address or name of the MSDP peer to shut down.
3.	**end**	Return to privileged EXEC mode.
4.	**show running-config**	Verify your entries.
5.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To bring the peer back up, use the **no ip msdp shutdown** {*peer-name* | *peer address*} global configuration command. The TCP connection is reestablished.

EXAMPLE

The following example shows how to shut down the MSDP peer at IP address 192.168.7.20:

```
Switch(config)# ip msdp shutdown 192.168.7.20
```

Including a Bordering PIM Dense-Mode Region in MSDP

You can configure MSDP on a switch that borders a PIM sparse-mode region with a dense-mode region. By default, active sources in the dense-mode region do not participate in MSDP.

Follow this procedure to configure the border router to send SA messages for sources active in the dense-mode region to the MSDP peers. This procedure is optional.

BEFORE YOU BEGIN

- We do not recommend using the **ip msdp border sa-address** global configuration command. It is better to configure the border router in the sparse-mode domain to proxy-register sources in the dense-mode domain to the RP of the sparse-mode domain and have the sparse-mode domain use standard MSDP procedures to advertise these sources.

- If you use the **ip msdp border sa-address** command, you must constrain the sources advertised by using the **ip msdp redistribute** command. Configure the **ip msdp redistribute** command to apply to only local sources. Be aware that this configuration can result in (S, G) state remaining long after a source in the dense mode domain has stopped sending.

- Note that the **ip msdp originator-id** global configuration command also identifies an interface to be used as the RP address. If both the **ip msdp border sa-address** and the **ip msdp originator-id** global configuration commands are configured, the address derived from the **ip msdp originator-id** command specifies the RP address.

DETAILED STEPS

Table 0-10

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip msdp border sa-address** *interface-id*	Configure the switch on the border between a dense-mode and sparse-mode region to send SA messages about active sources in the dense-mode region. For *interface-id*, specify the interface from which the IP address is derived and used as the RP address in SA messages. The IP address of the interface is used as the Originator-ID, which is the RP field in the SA message.
3.	**ip msdp redistribute** [**list** *access-list-name*] [**asn** *aspath-access-list-number*] [**route-map** *map*]	Configure which (S,G) entries from the multicast routing table are advertised in SA messages. For more information, see Redistributing Sources, page 780.
4.	**end**	Return to privileged EXEC mode.
5.	**show running-config**	Verify your entries.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting (active sources in the dense-mode region do not participate in MSDP), use the **no ip msdp border sa-address** *interface-id* global configuration command.

EXAMPLE

In the following example, the local router is not an RP. It borders a PIM sparse mode region with a dense mode region. It uses the IP address of Ethernet interface 0 as the “RP” address in SA messages.

```
Switch(config)# ip msdp border sa-address ethernet0
```

Configuring an Originating Address other than the RP Address

You can allow an MSDP speaker that originates an SA message to use the IP address of the interface as the RP address in the SA message by changing the Originator ID. You might change the Originator ID in one of these cases:

- If you configure a logical RP on multiple switches in an MSDP mesh group.
- If you have a switch that borders a PIM sparse-mode domain and a dense-mode domain. If a switch borders a dense-mode domain for a site, and sparse-mode is being used externally, you might want dense-mode sources to be known to the outside world. Because this switch is not an RP, it would not have an RP address to use in an SA message. Therefore, this command provides the RP address by specifying the address of the interface.

This procedure is optional.

BEFORE YOU BEGIN

If both the **ip msdp border sa-address** and the **ip msdp originator-id** global configuration commands are configured, the address derived from the **ip msdp originator-id** command specifies the address of the RP.

Configuring MSDP

DETAILED STEPS

Table 0-11

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip msdp originator-id** *interface-id*	Configures the RP address in SA messages to be the address of the originating device interface. For *interface-id*, specify the interface on the local switch.
3.	**end**	Return to privileged EXEC mode.
4.	**show running-config**	Verify your entries.
5.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To prevent the RP address from being derived in this way, use the **no ip msdp originator-id** *interface-id* global configuration command.

EXAMPLE

The following example shows how to configure the IP address of Ethernet interface 1 as the RP address in SA messages:

```
Switch(config)# ip msdp originator-id ethernet1
```

Verifying Configuration

Table 47-59

Command	Purpose
debug ip msdp [*peer-address* | *name*] [**detail**] [**routes**]	Debugs an MSDP activity.
debug ip msdp resets	Debugs MSDP peer reset reasons.
show ip msdp count [*autonomous-system-number*]	Displays the number of sources and groups originated in SA messages from each autonomous system. The **ip msdp cache-sa-state** command must be configured for this command to produce any output.
show ip msdp peer [*peer-address* | *name*]	Displays detailed information about an MSDP peer.
show ip msdp sa-cache [*group-address* | *source-address* | *group-name* | *source-name*] [*autonomous-system-number*]	Displays (S,G) state learned from MSDP peers.
show ip msdp summary	Displays MSDP peer status and SA message counts.

To clear MSDP connections, statistics, or SA cache entries, use the following privileged EXEC commands:

Table 47-60

Command	Purpose
clear ip msdp peer *peer-address* | *name*	Clears the TCP connection to the specified MSDP peer, resetting all MSDP message counters.
clear ip msdp statistics [*peer-address* | *name*]	Clears statistics counters for one or all the MSDP peers without resetting the sessions.
clear ip msdp sa-cache [*group-address* | *name*]	Clears the SA cache entries for all entries, all sources for a specific group, or all entries for a specific source/group pair.

Configuration Example

This example shows a partial configuration of Router A and Router C in Figure 95 on page 776. Each of these ISPs have more than one customer (like the customer in Figure 95 on page 776) who use default peering (no BGP or MBGP). In that case, they might have similar configurations. That is, they accept SAs only from a default peer if the SA is permitted by the corresponding prefix list.

Router A

```
Router(config)# ip msdp default-peer 10.1.1.1
Router(config)# ip msdp default-peer 10.1.1.1 prefix-list site-a
Router(config)# ip prefix-list site-b permit 10.0.0.0/1
```

Router C

```
Router(config)# ip msdp default-peer 10.1.1.1 prefix-list site-a
Router(config)# ip prefix-list site-b permit 10.0.0.0/1
```

This example shows how to enable the cache state for all sources in 171.69.0.0/16 sending to groups 224.2.0.0/16:

```
Switch(config)# ip msdp cache-sa-state 100
Switch(config)# access-list 100 permit ip 171.69.0.0 0.0.255.255 224.2.0.0 0.0.255.255
```

This example shows how to configure the switch to send SA request messages to the MSDP peer at 171.69.1.1:

```
Switch(config)# ip msdp sa-request 171.69.1.1
```

The following example shows how to configure which (S, G) entries from the mroute table are advertised in SA messages originated from AS 64512:

```
Switch(config)# ip msdp redistribute route-map customer-sources
Switch(config)# route-map customer-sources permit
Switch(config)# match as-path 100
Switch(config)# ip as-path access-list 100 permit ^64512$
```

This example shows how to configure the switch to filter SA request messages from the MSDP peer at 171.69.2.2. SA request messages from sources on network 192.4.22.0 pass access list 1 and are accepted; all others are ignored.

```
Switch(config)# ip msdp filter sa-request 171.69.2.2 list 1
Switch(config)# access-list 1 permit 192.4.22.0 0.0.0.255
```

This example shows how to allow only (S,G) pairs that pass access list 100 to be forwarded in an SA message to the peer named *switch.cisco.com*:

```
Switch(config)# ip msdp peer switch.cisco.com connect-source gigabitethernet0/1
Switch(config)# ip msdp sa-filter out switch.cisco.com list 100
Switch(config)# access-list 100 permit ip 171.69.0.0 0.0.255.255 224.20 0 0.0.255.255
```

The following example shows how to configure a TTL threshold of 8 hops:

```
Switch(config)# ip msdp ttl-threshold 192.168.1.5 8
```

This example shows how to filter all SA messages from the peer named *switch.cisco.com*:

```
Switch(config)# ip msdp peer switch.cisco.com connect-source gigabitethernet0/1
Switch(config)# ip msdp sa-filter in switch.cisco.com
```

The following example shows how to configure the MSDP peer at address 192.168.1.3 to be a member of the mesh group named internal:

```
Switch(config)# ip msdp mesh-group internal 192.168.1.3
```

The following example shows how to shut down the MSDP peer at IP address 192.168.7.20:

```
Switch(config)# ip msdp shutdown 192.168.7.20
```

In the following example, the local router is not an RP. It borders a PIM sparse mode region with a dense mode region. It uses the IP address of Ethernet interface 0 as the “RP” address in SA messages.

```
Switch(config)# ip msdp border sa-address ethernet0
```

The following example shows how to configure the IP address of Ethernet interface 1 as the RP address in SA messages:

```
Switch(config)# ip msdp originator-id ethernet1
```

Related Documents

- Cisco IOS IP Multicast Command Reference
- IP Multicast Configuration Guide Library, Cisco IOS Release 15M&T
- Cisco IOS Master Command List, All Releases

Configuring IPv6 MLD Snooping

This chapter describes how to configure Multicast Listener Discovery (MLD) snooping on the Cisco Industrial Ethernet Switches, hereafter referred to as *switch*. When the switch is running the IP services image, you can use MLD snooping to enable efficient distribution of IP version 6 (IPv6) multicast data to clients and routers in a switched network.

Note: To use IPv6, you must configure the dual IPv4 and IPv6 Switch Database Management (SDM) template on the switch. You select the template by entering the **sdm prefer dual-ipv4-and-ipv6** global configuration command.

Note: For complete syntax and usage information for the commands used in this chapter, see the documents listed in the Related Documents, page 807.

This chapter includes the following sections:

Information About MLD Snooping

In IP version 4 (IPv4), Layer 2 switches can use Internet Group Management Protocol (IGMP) snooping to limit the flooding of multicast traffic by dynamically configuring Layer 2 interfaces so that multicast traffic is forwarded to only those interfaces associated with IP multicast devices. In IPv6, MLD snooping performs a similar function. With MLD snooping, IPv6 multicast data is selectively forwarded to a list of ports that want to receive the data, instead of being flooded to all ports in a VLAN. This list is constructed by snooping IPv6 multicast control packets.

MLD is a protocol used by IPv6 multicast routers to discover the presence of multicast listeners (nodes wishing to receive IPv6 multicast packets) on its directly attached links and to discover which multicast packets are of interest to neighboring nodes. MLD is derived from IGMP; MLD version 1 (MLDv1) is equivalent to IGMPv2 and MLD version 2 (MLDv2) is equivalent to IGMPv3. MLD is a subprotocol of Internet Control Message Protocol version 6 (ICMPv6), and MLD messages are a subset of ICMPv6 messages, identified in IPv6 packets by a preceding Next Header value of 58.

The switch supports two versions of MLD snooping:

- MLDv1 snooping detects MLDv1 control packets and sets up traffic bridging based on IPv6 destination multicast addresses.
- MLDv2 basic snooping (MBSS) uses MLDv2 control packets to set up traffic forwarding based on IPv6 destination multicast addresses.

The switch can snoop on both MLDv1 and MLDv2 protocol packets and bridge IPv6 multicast data based on destination IPv6 multicast addresses.

Note: The switch does not support MLDv2 enhanced snooping (MESS), which sets up IPv6 source and destination multicast address-based forwarding.

MLD snooping can be enabled or disabled globally or per VLAN. When MLD snooping is enabled, a per-VLAN IPv6 multicast MAC address table is constructed in software and a per-VLAN IPv6 multicast address table is constructed in software and hardware. The switch then performs IPv6 multicast-address based bridging in hardware.

These sections describe some parameters of IPv6 MLD snooping:

- MLD Messages, page 796
- MLD Queries, page 796
- Multicast Client Aging Robustness, page 797
- Multicast Router Discovery, page 797
- MLD Reports, page 797
- MLD Done Messages and Immediate-Leave, page 798
- Topology Change Notification Processing, page 798

MLD Messages

MLDv1 supports three types of messages:

- Listener Queries are the equivalent of IGMPv2 queries and are either General Queries or Multicast-Address-Specific Queries (MASQs).
- Multicast Listener Reports are the equivalent of IGMPv2 reports.
- Multicast Listener Done messages are the equivalent of IGMPv2 leave messages.

MLDv2 supports MLDv2 queries and reports, as well as MLDv1 Report and Done messages.

Message timers and state transitions resulting from messages being sent or received are the same as those of IGMPv2 messages. MLD messages that do not have valid link-local IPv6 source addresses are ignored by MLD routers and switches.

MLD Queries

The switch sends out MLD queries, constructs an IPv6 multicast address database, and generates MLD group-specific and MLD group-and-source-specific queries in response to MLD Done messages. The switch also supports report suppression, report proxying, Immediate-Leave functionality, and static IPv6 multicast MAC-address configuration.

When MLD snooping is disabled, all MLD queries are flooded in the ingress VLAN.

When MLD snooping is enabled, received MLD queries are flooded in the ingress VLAN, and a copy of the query is sent to the CPU for processing. From the received query, MLD snooping builds the IPv6 multicast address database. It detects multicast router ports, maintains timers, sets report response time, learns the querier IP source address for the VLAN, learns the querier port in the VLAN, and maintains multicast-address aging.

Note: When the IPv6 multicast router is a Catalyst 6500 switch and you are using extended VLANs (in the range 1006 to 4094), IPv6 MLD snooping must be enabled on the extended VLAN on the Catalyst 6500 switch in order for this switch to receive queries on the VLAN. For normal-range VLANs (1 to 1005), it is not necessary to enable IPv6 MLD snooping on the VLAN on the Catalyst 6500 switch.

When a group exists in the MLD snooping database, the switch responds to a group-specific query by sending an MLDv1 report. When the group is unknown, the group-specific query is flooded to the ingress VLAN.

When a host wants to leave a multicast group, it can send out an MLD Done message (equivalent to IGMP Leave message). When the switch receives an MLDv1 Done message, if Immediate- Leave is not enabled, the switch sends an MASQ to the port from which the message was received to determine if other devices connected to the port should remain in the multicast group.

Multicast Client Aging Robustness

You can configure port membership removal from addresses based on the number of queries. A port is removed from membership to an address only when there are no reports to the address on the port for the configured number of queries. The default number is 2.

Multicast Router Discovery

Like IGMP snooping, MLD snooping performs multicast router discovery, with these characteristics:

- Ports configured by a user never age out.
- Dynamic port learning results from MLDv1 snooping queries and IPv6 PIMv2 packets.
- If there are multiple routers on the same Layer 2 interface, MLD snooping tracks a single multicast router on the port (the router that most recently sent a router control packet).
- Dynamic multicast router port aging is based on a default timer of 5 minutes; the multicast router is deleted from the router port list if no control packet is received on the port for 5 minutes.
- IPv6 multicast router discovery only takes place when MLD snooping is enabled on the switch.
- Received IPv6 multicast router control packets are always flooded to the ingress VLAN, whether or not MLD snooping is enabled on the switch.
- After the discovery of the first IPv6 multicast router port, unknown IPv6 multicast data is forwarded only to the discovered router ports (before that time, all IPv6 multicast data is flooded to the ingress VLAN).

MLD Reports

The processing of MLDv1 join messages is essentially the same as with IGMPv2. When no IPv6 multicast routers are detected in a VLAN, reports are not processed or forwarded from the switch. When IPv6 multicast routers are detected and an MLDv1 report is received, an IPv6 multicast group address and an IPv6 multicast MAC address are entered in the VLAN MLD database. Then all IPv6 multicast traffic to the group within the VLAN is forwarded using this address. When MLD snooping is disabled, reports are flooded in the ingress VLAN.

When MLD snooping is enabled, MLD report suppression, called listener message suppression, is automatically enabled. With report suppression, the switch forwards the first MLDv1 report received by a group to IPv6 multicast routers; subsequent reports for the group are not sent to the routers. When MLD snooping is disabled, report suppression is disabled, and all MLDv1 reports are flooded to the ingress VLAN.

The switch also supports MLDv1 proxy reporting. When an MLDv1 MASQ is received, the switch responds with MLDv1 reports for the address on which the query arrived if the group exists in the switch on another port and if the port on which the query arrived is not the last member port for the address.

MLD Done Messages and Immediate-Leave

When the Immediate-Leave feature is enabled and a host sends an MLDv1 Done message (equivalent to an IGMP leave message), the port on which the Done message was received is immediately deleted from the group.You enable Immediate-Leave on VLANs and (as with IGMP snooping), you should only use the feature on VLANs where a single host is connected to the port. If the port was the last member of a group, the group is also deleted, and the leave information is forwarded to the detected IPv6 multicast routers.

When Immediate Leave is not enabled in a VLAN (which would be the case when there are multiple clients for a group on the same port) and a Done message is received on a port, an MASQ is generated on that port. The user can control when a port membership is removed for an existing address in terms of the number of MASQs. A port is removed from membership to an address when there are no MLDv1 reports to the address on the port for the configured number of queries.

The number of MASQs generated is configured by using the **ipv6 mld snooping last-listener-query count** global configuration command. The default number is 2.

The MASQ is sent to the IPv6 multicast address for which the Done message was sent. If there are no reports sent to the IPv6 multicast address specified in the MASQ during the switch maximum response time, the port on which the MASQ was sent is deleted from the IPv6 multicast address database. The maximum response time is the time configured by using the **ipv6 mld snooping last-listener-query-interval** global configuration command. If the deleted port is the last member of the multicast address, the multicast address is also deleted, and the switch sends the address leave information to all detected multicast routers.

Topology Change Notification Processing

When topology change notification (TCN) solicitation is enabled by using the **ipv6 mld snooping tcn query solicit** global configuration command, MLDv1 snooping sets the VLAN to flood all IPv6 multicast traffic with a configured number of MLDv1 queries before it begins sending multicast data only to selected ports. You set this value by using the **ipv6 mld snooping tcn flood query count** global configuration command. The default is to send two queries. The switch also generates MLDv1 global Done messages with valid link-local IPv6 source addresses when the switch becomes the STP root in the VLAN or when it is configured by the user. This is same as done in IGMP snooping.

Prerequisites

Review the Information About MLD Snooping, page 795.

Guidelines and Limitations

- You can configure MLD snooping characteristics at any time, but you must globally enable MLD snooping by using the **ipv6 mld snooping** global configuration command for the configuration to take effect.
- When the IPv6 multicast router is a Catalyst 6500 switch and you are using extended VLANs (in the range 1006 to 4094), IPv6 MLD snooping must be enabled on the extended VLAN on the Catalyst 6500 switch in order for this switch to receive queries on the VLAN. For normal-range VLANs (1 to 1005), it is not necessary to enable IPv6 MLD snooping on the VLAN on the Catalyst 6500 switch.
- MLD snooping and IGMP snooping act independently of each other. You can enable both features at the same time on the switch.
- The maximum number of multicast entries allowed on the switch is determined by the configured SDM template.
- The maximum number of address entries allowed for the switch is 1000.

Default Settings

Table 48-61

Feature	Default Setting
MLD snooping (Global)	Disabled.
MLD snooping (per VLAN)	Enabled. MLD snooping must be globally enabled for VLAN MLD snooping to take place.
IPv6 Multicast addresses	None configured.
IPv6 Multicast router ports	None configured.
MLD snooping Immediate Leave	Disabled.
MLD snooping robustness variable	Global: 2; Per VLAN: 0. **Note:** The VLAN value overrides the global setting. When the VLAN value is 0, the VLAN uses the global count.
Last listener query count	Global: 2; Per VLAN: 0. **Note:** The VLAN value overrides the global setting. When the VLAN value is 0, the VLAN uses the global count.
Last listener query interval	Global: 1000 (1 second); VLAN: 0. **Note:** The VLAN value overrides the global setting. When the VLAN value is 0, the VLAN uses the global interval.
TCN query solicit	Disabled.
TCN query count	2.
MLD listener suppression	Enabled.

Configuring IPv6 MLD Snooping

These sections describe how to configure IPv6 MLD snooping:

- Enabling or Disabling MLD Snooping, page 799
- Configuring a Static Multicast Group, page 801
- Configuring a Multicast Router Port, page 801
- Enabling MLD Immediate Leave, page 802
- Configuring MLD Snooping Queries, page 803
- Disabling MLD Listener Message Suppression, page 805

Enabling or Disabling MLD Snooping

By default, IPv6 MLD snooping is globally disabled on the switch and enabled on all VLANs. When MLD snooping is globally disabled, it is also disabled on all VLANs. When you globally enable MLD snooping, the VLAN configuration overrides the global configuration. That is, MLD snooping is enabled only on VLAN interfaces in the default state (enabled).

Configuring IPv6 MLD Snooping

You can enable and disable MLD snooping on a per-VLAN basis or for a range of VLANs, but if you globally disable MLD snooping, it is disabled in all VLANs. If global snooping is enabled, you can enable or disable VLAN snooping.

Enabling MLD Snooping

DETAILED STEPS

Table 48-12

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ipv6 mld snooping**	Globally enable MLD snooping on the switch.
3.	**end**	Return to privileged EXEC mode.
4.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.
5.	**reload**	Reload the operating system.

To globally disable MLD snooping on the switch, use the **no ipv6 mld snooping** global configuration command.

EXAMPLE

This example shows how to enable MLD snooping globally:

```
Switch(config)# ipv6 mld snooping
```

Enabling MLD Snooping on a VLAN

DETAILED STEPS

Note: When the IPv6 multicast router is a Catalyst 6500 switch and you are using extended VLANs (in the range 1006 to 4094), IPv6 MLD snooping must be enabled on the extended VLAN on the Catalyst 6500 switch in order for this switch to receive queries on the VLAN. For normal-range VLANs (1 to 1005), it is not necessary to enable IPv6 MLD snooping on the VLAN on the Catalyst 6500 switch.

Table 48-13

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ipv6 mld snooping**	Globally enable MLD snooping on the switch.
3.	**ipv6 mld snooping vlan** *vlan-id*	Enable MLD snooping on the VLAN.The VLAN ID range is 1 to 1001 and 1006 to 4094. **Note:** MLD snooping must be globally enabled for VLAN snooping to be enabled.
4.	**end**	Return to privileged EXEC mode.
5.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable MLD snooping on a VLAN interface, use the **no ipv6 mld snooping vlan** *vlan-id* global configuration command for the specified VLAN number**.**

EXAMPLE

This example shows how to enable MLD snooping on a VLAN:

```
Switch(config)# ipv6 mld snooping vlan 100
```

Configuring a Static Multicast Group

Hosts or Layer 2 ports normally join multicast groups dynamically, but you can also statically configure an IPv6 multicast address and member ports for a VLAN. Follow this procedure to add a Layer 2 port as a member of a multicast group.

DETAILED STEPS

Table 48-14

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode
2.	**ipv6 mld snooping vlan** *vlan-id* **static** *ipv6_multicast_address* **interface** *interface-id*	Statically configure a multicast group with a Layer 2 port as a member of a multicast group: ■ *vlan-id* is the multicast group VLAN ID. The VLAN ID range is 1 to 1001 and 1006 to 4094. ■ *ipv6_multicast_address* is the 128-bit group IPv6 address. The address must be in the form specified in RFC 2373. ■ *interface-id* is the member port. It can be a physical interface or a port channel (1 to 10).
3.	**end**	Return to privileged EXEC mode.
4.	**show ipv6 mld snooping multicast-address user** or **show ipv6 mld snooping multicast-address vlan** *vlan-id* **user**	Verify the static member port and the IPv6 address.
5.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove a Layer 2 port from the multicast group, use the **no ipv6 mld snooping vlan** *vlan-id* **static** *mac-address* **interface** *interface-id* global configuration command. If all member ports are removed from a group, the group is deleted.

EXAMPLE

This example shows how to statically configure an IPv6 multicast group:

```
Switch# configure terminal
Switch(config)# ipv6 mld snooping vlan 2 static FF12::3 interface gigabitethernet0/1
Switch(config)# end
```

Configuring a Multicast Router Port

Although MLD snooping learns about router ports through MLD queries and PIMv6 queries, you can also use the command-line interface (CLI) to add a multicast router port to a VLAN. To add a multicast router port (add a static connection to a multicast router), use the **ipv6 mld snooping vlan mrouter** global configuration command on the switch.

BEFORE YOU BEGIN

Static connections to multicast routers are supported only on switch ports.

Configuring IPv6 MLD Snooping

DETAILED STEPS

Table 48-15

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ipv6 mld snooping vlan** *vlan-id* **mrouter interface** *interface-id*	Specify the multicast router VLAN ID, and specify the interface to the multicast router. ■ The VLAN ID range is 1 to 1001 and 1006 to 4094. ■ The interface can be a physical interface or a port channel. The port-channel range is 1 to 10.
3.	**end**	Return to privileged EXEC mode.
4.	**show ipv6 mld snooping mrouter** [**vlan** *vlan-id*]	Verify that IPv6 MLD snooping is enabled on the VLAN interface.
5.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove a multicast router port from the VLAN, use the **no ipv6 mld snooping vlan** *vlan-id* **mrouter interface** *interface-id* global configuration command.

EXAMPLE

This example shows how to add a multicast router port to VLAN 200:

```
Switch# configure terminal
Switch(config)# ipv6 mld snooping vlan 200 mrouter interface gigabitethernet0/2
Switch(config)# exit
```

Enabling MLD Immediate Leave

When you enable MLDv1 Immediate Leave, the switch immediately removes a port from a multicast group when it detects an MLD Done message on that port.

BEFORE YOU BEGIN

You should only use the Immediate-Leave feature when there is a single receiver present on every port in the VLAN. When there are multiple clients for a multicast group on the same port, you should not enable Immediate-Leave in a VLAN.

DETAILED STEPS

Table 48-16

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ipv6 mld snooping vlan** *vlan-id* **immediate-leave**	Enable MLD Immediate Leave on the VLAN interface.
3.	**end**	Return to privileged EXEC mode.
4.	**show ipv6 mld snooping vlan** *vlan-id*	Verify that Immediate Leave is enabled on the VLAN interface.
5.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable MLD Immediate Leave on a VLAN, use the **no ipv6 mld snooping vlan** *vlan-id* **immediate-leave** global configuration command.

EXAMPLE

This example shows how to enable MLD Immediate Leave on VLAN 130:

```
Switch# configure terminal
Switch(config)# ipv6 mld snooping vlan 130 immediate-leave
Switch(config)# exit
```

Configuring MLD Snooping Queries

When Immediate Leave is not enabled and a port receives an MLD Done message, the switch generates MASQs on the port and sends them to the IPv6 multicast address for which the Done message was sent. You can optionally configure the number of MASQs that are sent and the length of time the switch waits for a response before deleting the port from the multicast group.

Configuring IPv6 MLD Snooping

DETAILED STEPS

Table 48-17

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ipv6 mld snooping robustness-variable** *value*	(Optional) Set the number of queries that are sent before switch will deletes a listener (port) that does not respond to a general query. The range is 1 to 3; the default is 2.
3.	**ipv6 mld snooping vlan** *vlan-id* **robustness-variable** *value*	(Optional) Set the robustness variable on a VLAN basis, which determines the number of general queries that MLD snooping sends before aging out a multicast address when there is no MLD report response. The range is 1 to 3; the default is 0. When set to 0, the number used is the global robustness variable value.
4.	**ipv6 mld snooping last-listener-query-count** *count*	(Optional) Set the number of MASQs that the switch sends before aging out an MLD client. The range is 1 to 7; the default is 2. The queries are sent 1 second apart.
5.	**ipv6 mld snooping vlan** *vlan-id* **last-listener-query-count** *count*	(Optional) Set the last-listener query count on a VLAN basis. This value overrides the value configured globally. The range is 1 to 7; the default is 0. When set to 0, the global count value is used. Queries are sent 1 second apart.
6.	**ipv6 mld snooping last-listener-query-interval** *interval*	(Optional) Set the maximum response time that the switch waits after sending out a MASQ before deleting a port from the multicast group. The range is 100 to 32,768 thousands of a second. The default is 1000 (1 second).
7.	**ipv6 mld snooping vlan** *vlan-id* **last-listener-query-interval** *interval*	(Optional) Set the last-listener query interval on a VLAN basis. This value overrides the value configured globally. The range is 0 to 32,768 thousands of a second. The default is 0. When set to 0, the global last-listener query interval is used.
8.	**ipv6 mld snooping tcn query solicit**	(Optional) Enable topology change notification (TCN) solicitation, which means that VLANs flood all IPv6 multicast traffic for the configured number of queries before sending multicast data to only those ports requesting to receive it. The default is for TCN to be disabled.
9.	**ipv6 mld snooping tcn flood query count** *count*	(Optional) When TCN is enabled, specify the number of TCN queries to be sent. The range is from 1 to 10; the default is 2.
10.	**end**	Return to privileged EXEC mode.
11.	**show ipv6 mld snooping querier** [**vlan** *vlan-id*]	(Optional) Verify that the MLD snooping querier information for the switch or for the VLAN.
12.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

This example shows how to set the MLD snooping global robustness variable to 3:

```
Switch# configure terminal
Switch(config)# ipv6 mld snooping robustness-variable 3
Switch(config)# exit
```

This example shows how to set the MLD snooping last-listener query count for a VLAN to 3:

```
Switch# configure terminal
Switch(config)# ipv6 mld snooping vlan 200 last-listener-query-count 3
Switch(config)# exit
```

This example shows how to set the MLD snooping last-listener query interval (maximum response time) to 2000 (2 seconds):

```
Switch# configure terminal
Switch(config)# ipv6 mld snooping last-listener-query-interval 2000
Switch(config)# exit
```

Disabling MLD Listener Message Suppression

MLD snooping listener message suppression is enabled by default. When it is enabled, the switch forwards only one MLD report per multicast router query. When message suppression is disabled, multiple MLD reports could be forwarded to the multicast routers.

DETAILED STEPS

Table 48-18

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**no ipv6 mld snooping listener-message-suppression**	Disable MLD message suppression.
3.	**end**	Return to privileged EXEC mode.
4.	**show ipv6 mld snooping**	Verify that IPv6 MLD snooping report suppression is disabled.
5.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To re-enable MLD message suppression, use the **ipv6 mld snooping listener-message-suppression** global configuration command.

EXAMPLE

This example shows how to disable MLD message suppression:

```
Switch# configure terminal
Switch(config)# no ipv6 mld snooping listener-message-suppression
Switch(config)# end
```

Verifying Configuration

You can display MLD snooping information for dynamically learned and statically configured router ports and VLAN interfaces. You can also display MAC address multicast entries for a VLAN configured for MLD snooping.

Table 48-62

Command	Purpose
show ipv6 mld snooping [**vlan** *vlan-id*]	Display the MLD snooping configuration information for all VLANs on the switch or for a specified VLAN. (Optional) Enter **vlan** *vlan-id* to display information for a single VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4094.
show ipv6 mld snooping mrouter [**vlan** *vlan-id*]	Display information on dynamically learned and manually configured multicast router interfaces. When you enable MLD snooping, the switch automatically learns the interface to which a multicast router is connected. These are dynamically learned interfaces. (Optional) Enter **vlan** *vlan-id* to display information for a single VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4094.
show ipv6 mld snooping querier [**vlan** *vlan-id*]	Display information about the IPv6 address and incoming port for the most-recently received MLD query messages in the VLAN. (Optional) Enter **vlan** *vlan-id* to display information for a single VLAN.The VLAN ID range is 1 to 1001 and 1006 to 4094.
show ipv6 mld snooping multicast-address [**vlan** *vlan-id*] [**count** / **dynamic** / **user**]	Display all IPv6 multicast address information or specific IPv6 multicast address information for the switch or a VLAN. ■ Enter **count** to show the group count on the switch or in a VLAN. ■ Enter **dynamic** to display MLD snooping learned group information for the switch or for a VLAN. ■ Enter **user** to display MLD snooping user-configured group information for the switch or for a VLAN.
show ipv6 mld snooping multicast-address vlan *vlan-id* [*ipv6-multicast-address*]	Display MLD snooping for the specified VLAN and IPv6 multicast address.

Configuration Example

This example shows how to enable MLD snooping globally:

```
Switch(config)# ipv6 mld snooping
```

This example shows how to enable MLD snooping on a VLAN:

```
Switch(config)# ipv6 mld snooping vlan 100
```

This example shows how to statically configure an IPv6 multicast group:

```
Switch# configure terminal
Switch(config)# ipv6 mld snooping vlan 2 static FF12::3 interface gigabitethernet0/1
Switch(config)# end
```

This example shows how to add a multicast router port to VLAN 200:

```
Switch# configure terminal
Switch(config)# ipv6 mld snooping vlan 200 mrouter interface gigabitethernet0/2
Switch(config)# exit
```

This example shows how to enable MLD Immediate Leave on VLAN 130:

```
Switch# configure terminal
Switch(config)# ipv6 mld snooping vlan 130 immediate-leave
Switch(config)# exit
```

This example shows how to set the MLD snooping global robustness variable to 3:

```
Switch# configure terminal
Switch(config)# ipv6 mld snooping robustness-variable 3
Switch(config)# exit
```

This example shows how to set the MLD snooping last-listener query count for a VLAN to 3:

```
Switch# configure terminal
Switch(config)# ipv6 mld snooping vlan 200 last-listener-query-count 3
Switch(config)# exit
```

This example shows how to set the MLD snooping last-listener query interval (maximum response time) to 2000 (2 seconds):

```
Switch# configure terminal
Switch(config)# ipv6 mld snooping last-listener-query-interval 2000
Switch(config)# exit
```

This example shows how to disable MLD message suppression:

```
Switch# configure terminal
Switch(config)# no ipv6 mld snooping listener-message-suppression
Switch(config)# end
```

Related Documents

- Cisco IOS IPv6 Command Reference
- Cisco IOS Master Command List, All Releases

Configuring HSRP and VRRP

This chapter describes how to use Hot Standby Router Protocol (HSRP) to provide routing redundancy for routing IP traffic not dependent on the availability of any single router. HSRP for IPv4 is supported on switches running the IP services image.

You can also use a version of HSRP in Layer 2 mode to configure a redundant command switch to take over cluster management if the cluster command switch fails.

For complete syntax and usage information for the commands used in this chapter, see these documents:

- *Cisco IOS IP Command Reference, Volume 1 of 3: Addressing and Services, Release 12.2* at http://www.cisco.com/en/US/docs/ios/12_2/ipaddr/command/reference/fipras_r.html
- *Hot Standby Router Protocol Version 2* feature module at http://www.cisco.com/en/US/docs/ios/12_3t/12_3t4/feature/guide/gthsrpv2.html

This chapter consists of these sections:

Understanding HSRP

HSRP is Cisco’s standard method of providing high network availability by providing first-hop redundancy for IP hosts on an IEEE 802 LAN configured with a default gateway IP address. HSRP routes IP traffic without relying on the availability of any single router. It enables a set of router interfaces to work together to present the appearance of a single virtual router or default gateway to the hosts on a LAN. When HSRP is configured on a network or segment, it provides a virtual Media Access Control (MAC) address and an IP address that is shared among a group of configured routers. HSRP allows two or more HSRP-configured routers to use the MAC address and IP network address of a virtual router. The virtual router does not exist; it represents the common target for routers that are configured to provide backup to each other. One of the routers is selected to be the active router and another to be the standby router, which assumes control of the group MAC address and IP address should the designated active router fail.

Note Routers in an HSRP group can be any router interface that supports HSRP, including routed ports and switch virtual interfaces (SVIs).

HSRP provides high network availability by providing redundancy for IP traffic from hosts on networks. In a group of router interfaces, the active router is the router of choice for routing packets; the standby router is the router that takes over the routing duties when an active router fails or when preset conditions are met.

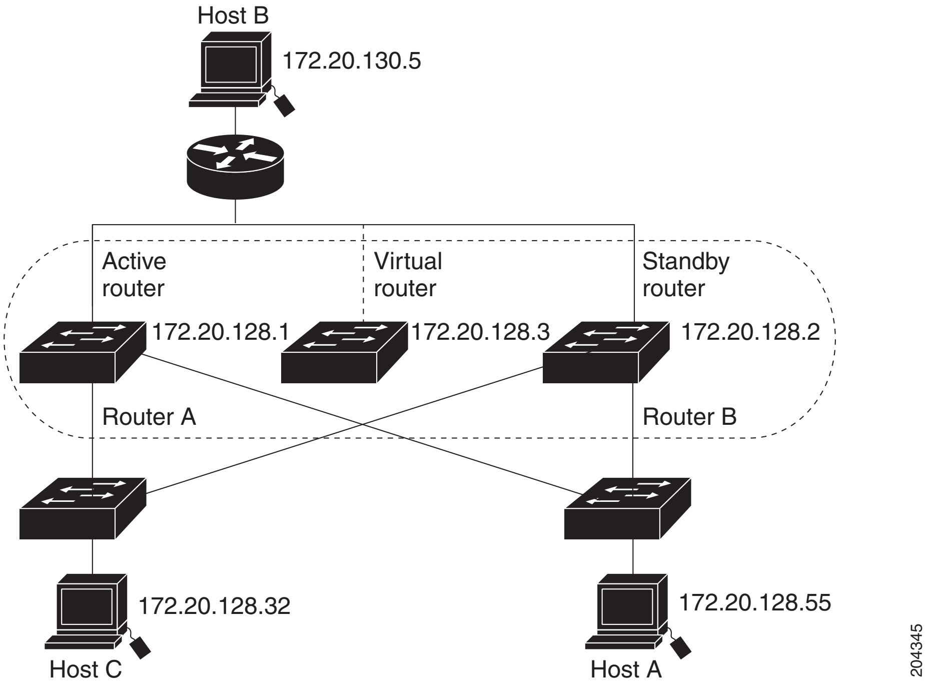
HSRP is useful for hosts that do not support a router discovery protocol and cannot switch to a new router when their selected router reloads or loses power. When HSRP is configured on a network segment, it provides a virtual MAC address and an IP address that is shared among router interfaces in a group of router interfaces running HSRP. The router selected by the protocol to be the active router receives and routes packets destined for the group's MAC address. For *n* routers running HSRP, there are *n +1* IP and MAC addresses assigned.

HSRP detects when the designated active router fails, and a selected standby router assumes control of the Hot Standby group's MAC and IP addresses. A new standby router is also selected at that time. Devices running HSRP send and receive multicast UDP-based hello packets to detect router failure and to designate active and standby routers. When HSRP is configured on an interface, Internet Control Message Protocol (ICMP) redirect messages are disabled by default for the interface.

You can configure multiple Hot Standby groups among switches that are operating in Layer 3 to make more use of the redundant routers. To do so, specify a group number for each Hot Standby command group you configure for an interface. For example, you might configure an interface on switch 1 as an active router and one on switch 2 as a standby router and also configure another interface on switch 2 as an active router with another interface on switch 1 as its standby router.

Figure 49-96 shows a segment of a network configured for HSRP. Each router is configured with the MAC address and IP network address of the virtual router. Instead of configuring hosts on the network with the IP address of Router A, you configure them with the IP address of the virtual router as their default router. When Host C sends packets to Host B, it sends them to the MAC address of the virtual router. If for any reason, Router A stops transferring packets, Router B responds to the virtual IP address and virtual MAC address and becomes the active router, assuming the active router duties. Host C continues to use the IP address of the virtual router to address packets destined for Host B, which Router B now receives and sends to Host B. Until Router A resumes operation, HSRP allows Router B to provide uninterrupted service to users on Host C's segment that need to communicate with users on Host B's segment and also continues to perform its normal function of handling packets between the Host A segment and Host B.

Figure 49-96 Typical HSRP Configuration



HSRP Versions

The switch supports these Hot Standby Redundancy Protocol (HSRP) versions:

- HSRPv1—Version 1 of the HSRP, the default version of HSRP. It has these features:
  - The HSRP group number can be from 0 to 255.
  - HSRPv1 uses the multicast address 224.0.0.2 to send hello packets, which can conflict with Cisco Group Management Protocol (CGMP) leave processing. You cannot enable HSRPv1 and CGMP at the same time; they are mutually exclusive.
- HSRPv2—Version 2 of the HSRP has these features:
  - To match the HSRP group number to the VLAN ID of a subinterface, HSRPv2 can use a group number from 0 to 4095 and a MAC address from 0000.0C9F.F000 to 0000.0C9F.FFFF.
  - HSRPv2 uses the multicast address 224.0.0.102 to send hello packets. HSRPv2 and CGMP leave processing are no longer mutually exclusive, and both can be enabled at the same time.
  - HSRPv2 has a different packet format than HRSPv1.

    A switch running HSRPv1 cannot identify the physical router that sent a hello packet because the source MAC address of the router is the virtual MAC address.

    HSRPv2 has a different packet format than HSRPv1. A HSRPv2 packet uses the type-length-value (TLV) format and has a 6-byte identifier field with the MAC address of the physical router that sent the packet.

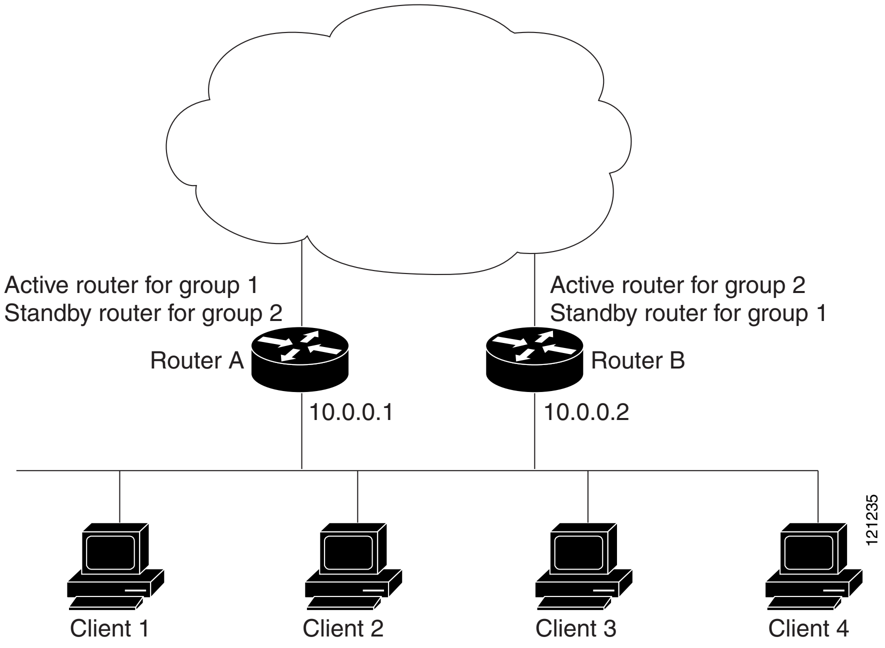
    If an interface running HSRPv1 gets an HSRPv2 packet, the type field is ignored.

Multiple HSRP

The switch supports Multiple HSRP (MHSRP), an extension of HSRP that allows load sharing between two or more HSRP groups. You can configure MHSRP to achieve load balancing and to use two or more standby groups (and paths) from a host network to a server network. In Figure 49-97, half the clients are configured for Router A, and half the clients are configured for Router B. Together, the configuration for Routers A and B establishes two HSRP groups. For group 1, Router A is the default active router because it has the assigned highest priority, and Router B is the standby router. For group 2, Router B is the default active router because it has the assigned highest priority, and Router A is the standby router. During normal operation, the two routers share the IP traffic load. When either router becomes unavailable, the other router becomes active and assumes the packet-transfer functions of the router that is unavailable.

Note For MHSRP, you need to enter the **standby preempt** interface configuration command on the HSRP interfaces so that if a router fails and then comes back up, preemption restores load sharing.

Figure 49-97 MHSRP Load Sharing


Active router for group 1
Standby router for group 2
Router A
Active router for group 2
Standby router for group 1
Router B
10.0.0.1
10.0.0.2
121235
Client 1
Client 2
Client 3
Client 4

Configuring HSRP

These sections contain this configuration information:

- Default HSRP Configuration, page -812
- HSRP Configuration Guidelines, page -813
- Enabling HSRP, page -814
- Configuring HSRP Priority, page -815
- Configuring MHSRP, page -817
- Configuring HSRP Authentication and Timers, page -818
- Enabling HSRP Support for ICMP Redirect Messages, page -819
- Configuring HSRP Groups and Clustering, page -819
- Troubleshooting HSRP, page -819

Default HSRP Configuration

Table 49-63 shows the default HSRP configuration.

Table 49-63 Default HSRP Configuration

Feature	Default Setting
HSRP version	Version 1
HSRP groups	None configured

Table 49-63 Default HSRP Configuration

Feature	Default Setting
Standby group number	0
Standby MAC address	System assigned as: 0000.0c07.acXX, where *XX* is the HSRP group number
Standby priority	100
Standby delay	0 (no delay)
Standby track interface priority	10
Standby hello time	3 seconds
Standby holdtime	10 seconds

HSRP Configuration Guidelines

Follow these guidelines when configuring HSRP:

- HSRP for IPv4 and HSRP for IPv6 are mutually exclusive. You cannot enable both at the same time.
- HSRPv2 and HSRPv1 are mutually exclusive. HSRPv2 is not interoperable with HSRPv1 on an interface and the reverse.
- You can configure up to 32 instances of HSRP groups.

  If you configure the same HSRP group number on multiple interfaces, the switch counts each interface as one instance:

  For example, if you configure HSRP group 0 on VLAN 1 and on port 1, the switch counts this as two instances.
- In the configuration procedures, the specified interface must be a Layer 3 interface:
  - Routed port: a physical port configured as a Layer 3 port by entering the **no switchport** interface configuration command.
  - SVI: a VLAN interface created by using the **interface vlan** *vlan_id* global configuration command and by default a Layer 3 interface.
  - EtherChannel port channel in Layer 3 mode: a port-channel logical interface created by using the **interface port-channel** *port-channel-number* global configuration command and binding the Ethernet interface into the channel group. For more information, see the “Configuring Layer 3 EtherChannels” section.
- All Layer 3 interfaces must have assigned IP addresses.
- Configure only one instance of an FHRP. The switches support HSRPv1, HSRPv2, and HSRP for IPv6.
- The version of an HSRP group can be changed from HSRPv2 to HSRPv1 only if the group number is less than 256.
- When configuring group numbers for HSRPv2 and HSRP for IPv6, you must use group numbers in ranges that are multiples of 256. Valid ranges are 0 to 255, 256 to 511, 512 to 767, 3840 to 4095, and so on.

  Examples of valid and invalid group numbers:
  - If you configure groups with the numbers 2, 150, and 225, you cannot configure another group with the number 3850. It is not in the range of 0 to 255.

   - If you configure groups with the numbers 520, 600, and 700, you cannot configure another group with the number 900. It is not in the range of 512 to 767.

- If you change the HSRP version on an interface, each HSRP group resets because it now has a new virtual MAC address.

Enabling HSRP

The **standby ip** interface configuration command activates HSRP on the configured interface. If an IP address is specified, that address is used as the designated address for the Hot Standby group. If no IP address is specified, the address is learned through the standby function. You must configure at least one Layer 3 port on the LAN with the designated address. Configuring an IP address always overrides another designated address currently in use.

When the **standby ip** command is enabled on an interface and proxy ARP is enabled, if the interface's Hot Standby state is active, proxy ARP requests are answered using the Hot Standby group MAC address. If the interface is in a different state, proxy ARP responses are suppressed.

Beginning in privileged EXEC mode, follow these steps to create or enable HSRP on a Layer 3 interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and enter the Layer 3 interface on which you want to enable HSRP.
Step 3	**standby version** {**1** | **2**}	(Optional) Configure the HSRP version on the interface. • 1— Select HSRPv1. • 2— Select HSRPv2. If you do not enter this command or do not specify a keyword, the interface runs the default HSRP version, HSRP v1.
Step 4	**standby** [*group-number*] **ip** [*ip-address* [**secondary**]]	Create (or enable) the HSRP group using its number and virtual IP address. • (Optional) *group-number*—The group number on the interface for which HSRP is being enabled. The range is 0 to 255; the default is 0. If there is only one HSRP group, you do not need to enter a group number. • (Optional on all but one interface) *ip-address*—The virtual IP address of the hot standby router interface. You must enter the virtual IP address for at least one of the interfaces; it can be learned on the other interfaces. • (Optional) **secondary**—The IP address is a secondary hot standby router interface. If neither router is designated as a secondary or standby router and no priorities are set, the primary IP addresses are compared and the higher IP address is the active router, with the next highest as the standby router.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show standby** [*interface-id* [*group*]]	Verify the configuration.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no standby** [*group-number*] **ip** [*ip-address*] interface configuration command to disable HSRP.

This example shows how to activate HSRP for group 1 on an interface. The IP address used by the hot standby group is learned by using HSRP.

Note This procedure is the minimum number of steps required to enable HSRP. Other configuration is optional.

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/1
Switch(config-if)# no switchport
Switch(config-if)# standby 1 ip
Switch(config-if)# end
Switch# show standby
```

Configuring HSRP Priority

The **standby priority**, **standby preempt**, and **standby track** interface configuration commands are all used to set characteristics for finding active and standby routers and behavior regarding when a new active router takes over.

When configuring HSRP priority, follow these guidelines:

- Assigning a priority allows you to select the active and standby routers. If preemption is enabled, the router with the highest priority becomes the active router. If priorities are equal, the current active router does not change.
- The highest number (1 to 255) represents the highest priority (most likely to become the active router).
- When setting the priority, preempt, or both, you must specify at least one keyword (**priority**, **preempt**, or both).
- The priority of the device can change dynamically if an interface is configured with the **standby track** command and another interface on the router goes down.
- The **standby track** interface configuration command ties the router hot standby priority to the availability of its interfaces and is useful for tracking interfaces that are not configured for HSRP. When a tracked interface fails, the hot standby priority on the device on which tracking has been configured decreases by 10. If an interface is not tracked, its state changes do not affect the hot standby priority of the configured device. For each interface configured for hot standby, you can configure a separate list of interfaces to be tracked.
- The **standby track** *interface-priority* interface configuration command specifies how much to decrement the hot standby priority when a tracked interface goes down. When the interface comes back up, the priority is incremented by the same amount.
- When multiple tracked interfaces are down and *interface-priority* values have been configured, the configured priority decrements are cumulative. If tracked interfaces that were not configured with priority values fail, the default decrement is 10, and it is noncumulative.
- When routing is first enabled for the interface, it does not have a complete routing table. If it is configured to preempt, it becomes the active router, even though it is unable to provide adequate routing services. To solve this problem, configure a delay time to allow the router to update its routing table.

Beginning in privileged EXEC mode, use one or more of these steps to configure HSRP priority characteristics on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and enter the HSRP interface on which you want to set priority.
Step 3	**standby** [*group-number*] **priority** *priority* [**preempt** [**delay** *delay*]]	Set a **priority** value used in choosing the active router. The range is 1 to 255; the default priority is 100. The highest number represents the highest priority. • (Optional) *group-number*—The group number to which the command applies. • (Optional) **preempt—**Select so that when the local router has a higher priority than the active router, it assumes control as the active router. • (Optional) **delay**—Set to cause the local router to postpone taking over the active role for the shown number of seconds. The range is 0 to 3600(1 hour); the default is 0 (no delay before taking over). Use the **no** form of the command to restore the default values.
Step 4	**standby** [*group-number*] [**priority** *priority*] **preempt** [**delay** *delay*]	Configure the router to **preempt**, which means that when the local router has a higher priority than the active router, it assumes control as the active router. • (Optional) *group-number*—The group number to which the command applies. • (Optional) **priority**—Enter to set or change the group priority. The range is 1 to 255; the default is 100. • (Optional) **delay**—Set to cause the local router to postpone taking over the active role for the number of seconds shown. The range is 0 to 3600 (1 hour); the default is 0 (no delay before taking over). Use the **no** form of the command to restore the default values.
Step 5	**standby** [*group-number*] **track** *type number* [*interface-priority*]	Configure an interface to track other interfaces so that if one of the other interfaces goes down, the device's Hot Standby priority is lowered. • (Optional) *group-number*—The group number to which the command applies. • *type*—Enter the interface type (combined with interface number) that is tracked. • *number*—Enter the interface number (combined with interface type) that is tracked. • (Optional) *interface-priority*—Enter the amount by which the hot standby priority for the router is decremented or incremented when the interface goes down or comes back up. The default value is 10.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running-config**	Verify the configuration of the standby groups.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no standby** [*group-number*] **priority** *priority* [**preempt** [**delay** *delay*]] and **no standby** [*group-number*] [**priority** *priority*] **preempt** [**delay** *delay*] interface configuration commands to restore default priority, preempt, and delay values.

Use the **no standby** [*group-number*] **track** *type number* [*interface-priority*] interface configuration command to remove the tracking.

This example activates a port, sets an IP address and a priority of 120 (higher than the default value), and waits for 300 seconds (5 minutes) before attempting to become the active router:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/1
Switch(config-if)# no switchport
Switch(config-if)# standby ip 172.20.128.3
Switch(config-if)# standby priority 120 preempt delay 300
Switch(config-if)# end
```

Configuring MHSRP

To enable MHSRP and load balancing, you configure two routers as active routers for their groups, with virtual routers as standby routers. This example shows how to enable the MHSRP configuration shown in Figure 49-97. You need to enter the **standby preempt** interface configuration command on each HSRP interface so that if a router fails and comes back up, the preemption occurs and restores load balancing.

Router A is configured as the active router for group 1, and Router B is configured as the active router for group 2. The HSRP interface for Router A has an IP address of 10.0.0.1 with a group 1 standby priority of 110 (the default is 100). The HSRP interface for Router B has an IP address of 10.0.0.2 with a group 2 standby priority of 110.

Group 1 uses a virtual IP address of 10.0.0.3 and group 2 uses a virtual IP address of 10.0.0.4.

Router A Configuration

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/1
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.0.0.1 255.255.255.0
Switch(config-if)# standby 1 ip 10.0.0.3
Switch(config-if)# standby 1 priority 110
Switch(config-if)# standby 1 preempt
Switch(config-if)# standby 2 ip 10.0.0.4
Switch(config-if)# standby 2 preempt
Switch(config-if)# end
```

Router B Configuration

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/1
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.0.0.2 255.255.255.0
Switch(config-if)# standby 1 ip 10.0.0.3
Switch(config-if)# standby 1 preempt
Switch(config-if)# standby 2 ip 10.0.0.4
Switch(config-if)# standby 2 priority 110
Switch(config-if)# standby 2 preempt
Switch(config-if)# end
```

Configuring HSRP Authentication and Timers

You can optionally configure an HSRP authentication string or change the hello-time interval and holdtime.

When configuring these attributes, follow these guidelines:

- The authentication string is sent unencrypted in all HSRP messages. You must configure the same authentication string on all routers and access servers on a cable to ensure interoperation. Authentication mismatch prevents a device from learning the designated Hot Standby IP address and timer values from other routers configured with HSRP.
- Routers or access servers on which standby timer values are not configured can learn timer values from the active or standby router. The timers configured on an active router always override any other timer settings.
- All routers in a Hot Standby group should use the same timer values. Normally, the *holdtime* is greater than or equal to 3 times the *hellotime*.

Beginning in privileged EXEC mode, use one or more of these steps to configure HSRP authentication and timers on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Enter interface configuration mode, and enter the HSRP interface on which you want to set authentication.
Step 3	**standby** [*group-number*] **authentication** *string*	(Optional) **authentication** *string*—Enter a string to be carried in all HSRP messages. The authentication string can be up to eight characters in length; the default string is **cisco.** (Optional) *group-number*—The group number to which the command applies.
Step 4	**standby** [*group-number*] **timers** *hellotime holdtime*	(Optional) Configure the time between hello packets and the time before other routers declare the active router to be down. • *group-number*—The group number to which the command applies. • *hellotime*—The hello interval in seconds. The range is from 1 to 255; the default is 3 seconds. • *holdtime*—The time in seconds before the active or standby router is declared to be down. The range is from 1 to 255; the default is 10 seconds.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify the configuration of the standby groups.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no standby** [*group-number*] **authentication** *string* interface configuration command to delete an authentication string. Use the **no standby** [*group-number*] **timers** *hellotime holdtime* interface configuration command to restore timers to their default values.

This example shows how to configure *word* as the authentication string required to allow Hot Standby routers in group 1 to interoperate:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/1
Switch(config-if)# no switchport
Switch(config-if)# standby 1 authentication word
Switch(config-if)# end
```

This example shows how to set the timers on standby group 1 with the time between hello packets at 5 seconds and the time after which a router is considered down to be 15 seconds:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/1
Switch(config-if)# no switchport
Switch(config-if)# standby 1 ip
Switch(config-if)# standby 1 timers 5 15
Switch(config-if)# end
```

Enabling HSRP Support for ICMP Redirect Messages

The Internet Control Message Protocol (ICMP) is a network layer Internet protocol that provides message packets to report errors and other information relevant to IP processing. ICMP provides diagnostic functions, such as sending and directing error packets to the host.

When the switch is running HSRP, make sure hosts do not discover the interface (or real) MAC addresses of routers in the HSRP group. If a host is redirected by ICMP to the real MAC address of a router and that router later fails, packets from the host will be lost.

ICMP redirect messages are automatically enabled on interfaces configured with HSRP. This feature filters outgoing ICMP redirect messages through HSRP, in which the next hop IP address might be changed to an HSRP virtual IP address.

Configuring HSRP Groups and Clustering

When a device is participating in an HSRP standby routing and clustering is enabled, you can use the same standby group for command switch redundancy and HSRP redundancy. Use the **cluster standby-group** *HSRP-group-name* [**routing-redundancy**] global configuration command to enable the same HSRP standby group to be used for command switch and routing redundancy. If you create a cluster with the same HSRP standby group name without entering the **routing-redundancy** keyword, HSRP standby routing is disabled for the group.

This example shows how to bind standby group my_hsrp to the cluster and enable the same HSRP group to be used for command switch redundancy and router redundancy. The command can only be executed on the cluster command switch. If the standby group name or number does not exist, or if the switch is a cluster member switch, an error message appears.

```
Switch# configure terminal
Switch(config)# cluster standby-group my_hsrp routing-redundancy
Switch(config)# end
```

Troubleshooting HSRP

If one of the situations in Table 49-64 occurs, this message appears:

```
%FHRP group not consistent with already configured groups on the switch stack -
virtual MAC reservation failed
```

Table 49-64 Troubleshooting HSRP

Situation	Action
You configure more than 32 HSRP group instances.	Remove HSRP groups so that up to 32 group instances are configured.
You configure HSRP for IPv4 and HSRP for IPv6 at the same time	Configure either HSRP for IPv4 or HSRP for IPv6 on the switch.
You configure group numbers that are not in valid ranges of 256.	Configure group numbers in a valid range.

Displaying HSRP Configurations

From privileged EXEC mode, use this command to display HSRP settings:

show standby [*interface-id* [*group*]] [**brief**] [**detail**]

You can display HSRP information for the whole switch, for a specific interface, for an HSRP group, or for an HSRP group on an interface. You can also specify whether to display a concise overview of HSRP information or detailed HSRP information. The default display is **detail**. If there are a large number of HSRP groups, using the **show standby** command without qualifiers can result in an unwieldy display.

This is a an example of output from the **show standby** privileged EXEC command, displaying HSRP information for two standby groups (group 1 and group 100):

```
Switch# show standby
VLAN1 - Group 1
   Local state is Standby, priority 105, may preempt
   Hellotime 3 holdtime 10
   Next hello sent in 00:00:02.182
   Hot standby IP address is 172.20.128.3 configured
   Active router is 172.20.128.1 expires in 00:00:09
   Standby router is local
   Standby virtual mac address is 0000.0c07.ac01
   Name is bbb
VLAN1 - Group 100
   Local state is Active, priority 105, may preempt
   Hellotime 3 holdtime 10
   Next hello sent in 00:00:02.262
   Hot standby IP address is 172.20.138.51 configured
   Active router is local
   Standby router is unknown expired
   Standby virtual mac address is 0000.0c07.ac64
   Name is test
```

Configuring VRRP

The Virtual Router Redundancy Protocol (VRRP) is an election protocol that dynamically assigns responsibility for one or more virtual routers to the VRRP routers on a LAN, allowing several routers on a multiaccess link to utilize the same virtual IP address. A VRRP router is configured to run the VRRP protocol in conjunction with one or more other routers attached to a LAN. In a VRRP configuration, one router is elected as the virtual router master, with the other routers acting as backups in case the virtual router master fails.

VRRP Limitations

- The switch supports either HSRP or VRRP, but not both. The switch cannot join a stack that has both HSRP and VRRP configured.
- The VRRP implementation on the switch does not support the MIB specified in RFC 2787.
- The VRRP implementation on the switch supports only text**-based authentication.**
- You cannot enable VRRP for IPv4 and IPv6 groups simultaneously.

Configuring IPv6 ACLs

This chapter provides details about configuring IPv6 access control lists (ACLs) on the Cisco Industrial Ethernet Switches, hereafter referred to as *switch*.

When the switch is running the IP services image:

- You can filter IPv6 traffic by creating IPv6 ACLs and applying them to interfaces
- You can create and apply input router ACLs to filter Layer 3 management traffic

This chapter contains the following sections:

Information About IPv6 ACLs

A switch running the IP services image supports two types of IPv6 ACLs:

- IPv6 *router ACLs* on outbound or inbound traffic on Layer 3 interfaces only, which can be routed ports, switch virtual interfaces (SVIs), or Layer 3 EtherChannels.

  IPv6 router ACLs apply only to routed IPv6 packets.

- *IPv6 port ACLs* on inbound traffic on Layer 2 interfaces only. The switch applies IPv6 port ACLs to all IPv6 packets entering the interface.

You can apply both IPv4 and IPv6 ACLs to an interface.

As with IPv4 ACLs, IPv6 port ACLs take precedence over router ACLs:

- When an input router ACL and input port ACL exist in an SVI, packets received on ports to which a port ACL is applied are filtered by the port ACL. Routed IP packets received on other ports are filtered by the router ACL. Other packets are not filtered.
- When an output router ACL and input port ACL exist in an SVI, packets received on the ports to which a port ACL is applied are filtered by the port ACL. Outgoing routed IPv6 packets are filtered by the router ACL. Other packets are not filtered.

Note: When you apply *any* port ACL (IPv4, IPv6, or MAC) to an interface, that port ACL filters packets, and ignores any router ACLs attached to the SVI of the port VLAN.

Supported ACL Features

IPv6 ACLs on the switch have these characteristics:

- Fragmented frames (the **fragments** keyword as in IPv4) are supported.
- The same statistics supported in IPv4 are supported for IPv6 ACLs.
- If the switch runs out of hardware space, packets associated with the ACL are forwarded to the CPU, and the software applies the ACLs.
- Routed or bridged packets with hop-by-hop options have IPv6 ACLs applied in software.
- Logging is supported for router ACLs, but not for port ACLs.
- The switch supports IPv6 address-matching for a full range of prefix-lengths.

Note: For items not supported for IPv6 ACLS, see Guidelines and Limitations, page 824.

Prerequisites

Be sure to review Guidelines and Limitations, page 824 and the Before You Begin section within each configuration section before configuring a feature.

Guidelines and Limitations

ACLs for IPv6 Traffic Not Supported

- The switch does not support VLAN ACLs (VLAN maps) for IPv6 traffic.
- The switch does not apply MAC-based ACLs on IPv6 frames.
- You cannot apply IPv6 port ACLs to Layer 2 EtherChannels.
- The switch does not support output port ACLs.

Cisco IOS IPv6 ACLs Functions Not Supported

- The switch does not support matching on these keywords: **flowlabel**, **routing header**, and **undetermined-transport**.
- The switch does not support reflexive ACLs (the **reflect** keyword).

Access Control Entry (ACE) and ACLs

- When you apply an ACL to an interface and you attempt to add an access control entry (ACE) with an unsupported keyword, the switch does not allow the ACE to be added to the attached ACL.

Named ACLs

- IPv6 supports only named ACLs.

IPv6 ACLs Interactions With Other Switches or Features

- When you configure an IPv6 router ACL to deny a packet, the software does not route the packet. Instead, the software forwards a copy of the packet to the Internet Control Message Protocol (ICMP) queue to generate an ICMP unreachable message for the frame.
- If a bridged frame is to be dropped due to a port ACL, the frame is not bridged.

- You can create both IPv4 and IPv6 ACLs on a switch, and you can apply both IPv4 and IPv6 ACLs to the same interface.
  - Each ACL must have a unique name; and, an error message appears if you try to use a name that already exists on the switch.
  - You use different commands to create IPv4 and IPv6 ACLs and to attach IPv4 or IPv6 ACLs to the same Layer 2 or Layer 3 interface.

  If you use the wrong command to attach an ACL (for example, an IPv4 command to attach an IPv6 ACL), you receive an error message.
- You cannot use MAC ACLs to filter IPv6 frames. MAC ACLs can only filter non-IP frames.
- If the hardware memory is full, for any additional configured ACLs, the switch forwards the packets to the CPU, and the software applies the ACLs.

Default Settings

Parameters	Default
IPv6 ACLs	There are no default IPv6 ACLs configured or applied on the switch.

Configuring IPv6 ACLs

This section includes the following topics:

- Creating IPv6 ACLs, page 825
- Applying an IPv6 ACL to an Interface, page 829

BEFORE YOU BEGIN

Review the Guidelines and Limitations, page 824 for this feature.

Select one of the dual IPv4 and IPv6 SDM templates.

Creating IPv6 ACLs

Note: When you configure an unsupported IPv6 ACL, an error message appears, and the configuration does not take affect.

Use the **no** {**deny** | **permit**} IPv6 access-list configuration commands with keywords to remove the deny or permit conditions from the specified access list for the commands below.

Configuring IPv6 ACLs

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ipv6 access-list** *access-list-name*	Define an IPv6 access list using a name, and enter IPv6 access-list configuration mode.
3. a	{**deny** | **permit**} *protocol* {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix/ prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [**dscp** *value*] [**fragments**] [**log**] [**log-input**] [**routing**] [**sequence** *value*] [**time-range** *name*]	Deny or permit the packet, when specified conditions are matched. These are the conditions: ■ *protocol*–Name or number of an Internet protocol: **ahp**, **esp**, **icmp**, **ipv6**, **pcp**, **stcp**, **tcp**, or **udp**, or an integer in the range 0 to 255 representing an IPv6 protocol number. For additional specific parameters for ICMP, TCP, and UDP, see Steps 3b through 3d. ■ *source-ipv6-prefix/prefix-length* or *destination-ipv6-prefix/ prefix-length*–Source or destination IPv6 network or class of networks for which to set deny or permit conditions, specified in hexadecimal and using 16-bit values between colons. ■ Enter **any** as an abbreviation for the IPv6 prefix ::/0. ■ **host** *source-ipv6-address or destination-ipv6-address*– Define source or destination IPv6 host address for which to set deny or permit conditions, specified in hexadecimal and using 16-bit values between colons. ■ (Optional) *operator*–Operand that compares the source or destination ports of the specified protocol such as **lt** (less than), **gt** (greater than), **eq** (equal), **neq** (not equal), and **range**. If the operator follows the *source-ipv6-prefix/prefix-length* argument, it must match the source port. If the operator follows the *destination-ipv6- prefix/prefix-length* argument, it must match the destination port.

	Command	Purpose
		■ (Optional) *port-number*– Value of 0 to 65535 or TCP or UDP port name. Use TCP port names only when filtering TCP. Use UDP port names only when filtering UDP. ■ (Optional) **dscp** *value*–Match a differentiated services code point value against the traffic class value in the Traffic Class field of each IPv6 packet header. The acceptable range is from 0 to 63. ■ (Optional) **fragments–**Check noninitial fragments. Keyword is only visible when the protocol is **ipv6**. ■ (Optional, router ACLs only) **log–**Send a logging message to the console about the packet that matches the entry. Enter **log-input** to include the input interface in the log entry. ■ (Optional) **routing**–Specify routing of IPv6 packets. ■ (Optional) **sequence** *value*–Specify the sequence number for the access list statement. Value range is from 1 to 4294967295. ■ (Optional) **time-range** *name*–Specify the time range that applies to the deny or permit statement.
Step 3b	{**deny** | **permit**} **tcp** {*source-ipv6-prefix***/***prefix-length* | **any** | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix***/***prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [**ack**] [**dscp** *value*] [**established**] [**fin**] [**log**] [**log-input**] [**neq** {*port* | *protocol*}] [**psh**] [**range** {*port* | *protocol*}] [**rst**] [**routing**] [**sequence** *value*] [**syn**] [**time-range** *name*] [**urg**]	(Optional) Define a TCP access list and the access conditions. Enter **tcp** for Transmission Control Protocol. The parameters are the same as those described in Step 3a, with these additional optional parameters: ■ **ack**–Acknowledgment bit set. ■ **established**–An established connection. A match occurs if the TCP datagram has the ACK or RST bits set. ■ **fin**–Finished bit set; no more data from sender. ■ **neq** {*port* | *protocol*}–Match only packets that are not on a given port number. ■ **psh**–Push function bit set. ■ **range** {*port* | *protocol*}–Match only packets in the port number range. ■ **rst**–Reset bit set. ■ **syn**–Synchronize bit set. ■ **urg**–Urgent pointer bit set.

	Command	Purpose
Step 3c	{**deny** | **permit**} **udp** {*source-ipv6-prefix***/***prefix-length* | **any** | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix***/***prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [**dscp** *value*] [**log**] [**log-input**] [**neq** {*port* | *protocol*}] [**range** {*port* | *protocol*}] [**routing**] [**sequence** *value*] [**time-range** *name*]	(Optional) Define a UDP access list and the access conditions. Enter **udp** for the User Datagram Protocol. The UDP parameters are the same as those described for TCP, except that the [*operator* [*port*]] port number or name must be a UDP port number or name, and the **established** parameter is not valid for UDP.
Step 3d	{**deny** | **permit**} **icmp** {*source-ipv6-prefix***/***prefix-length* | **any** | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix***/***prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [*icmp-type* [*icmp-code*] | *icmp-message*] [**dscp** *value*] [**log**] [**log-input**] [**routing**] [**sequence** *value*] [**time-range** *name*]	(Optional) Define an ICMP access list and the access conditions. Enter **icmp** for Internet Control Message Protocol. The ICMP parameters are the same as those described for most IP protocols in Step 3a, with the addition of the ICMP message type and code parameters. These optional keywords have these meanings: ■ *icmp-type*—Enter to filter by ICMP message type, a number from 0 to 255. ■ *icmp-code*—Enter to filter ICMP packets that are filtered by the ICMP message code type, a number from 0 to 255. ■ *icmp-message*—Enter to filter ICMP packets by the ICMP message type name or the ICMP message type and code name. To see a list of ICMP message type names and code names, use the ? key.
4.	**end**	Return to privileged EXEC mode.
5.	**show ipv6 access-list**	Verify the access list configuration.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

The following example:

- Creates an IPv6 ACL named CISCO.
- Defines one deny entry that denies all packets that have a destination TCP port number greater than 5000 and a second deny entry that denies packets that have a source UDP port number less than 5000. The second deny entry also logs all matches to the console.
- Defines a permit entry to permit all ICMP packets and another permit entry that allows all other traffic. The second permit entry is necessary because an implicit deny-all condition is at the end of each IPv6 access list.

```
Switch(config)# ipv6 access-list CISCO
Switch(config-ipv6-acl)# deny tcp any any gt 5000
Switch config-ipv6-acl)# deny ::/0 lt 5000 ::/0 log
Switch(config-ipv6-acl)# permit icmp any any
Switch(config-ipv6-acl)# permit any any
```

Applying an IPv6 ACL to an Interface

BEFORE YOU BEGIN

Review the Guidelines and Limitations, page 824 for this feature.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Identify a Layer 2 interface (for port ACLs) or Layer 3 interface (for router ACLs) on which to apply an access list, and enter interface configuration mode.
3.	**no switchport**	If applying a router ACL, change the interface from Layer 2 mode (the default) to Layer 3 mode.
4.	**ipv6 address** *ipv6-address*	Configure an IPv6 address on a Layer 3 interface (for router ACLs). **Note:** This command is not required on Layer 2 interfaces or if the interface is already configured with an explicit IPv6 address. Use the **no ipv6 traffic-filter** *access-list-name* interface configuration command to remove an access list from an interface.
5.	**ipv6 traffic-filter** *access-list-name* {**in** | **out**}	Apply the access list to incoming or outgoing traffic on the interface. **Note:** The **out** keyword is not supported for Layer 2 interfaces (port ACLs).
6.	**end**	Return to privileged EXEC mode.
7.	**show running-config**	Verify the access list configuration.
8.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

This example shows how to apply the access list CISCO to outbound traffic on a Layer 3 interface:

```
Switch(config)# interface gigabitethernet 0/3
Switch(config-if)# no switchport
Switch(config-if)# ipv6 address 2001::/64 eui-64
Switch(config-if)# ipv6 traffic-filter CISCO out
```

Verifying IPv6 ACLs

You can display information about all configured access lists, all IPv6 access lists, or a specific access list by using one or more of the following privileged EXEC commands.

Command	Purpose
show access-lists	Display all access lists configured on the switch.
show ipv6 access-list [*access-list-name*]	Display all configured IPv6 access list or the access list specified by name.

Configuration Example

The following example:

- Creates an IPv6 ACL named CISCO.
- Defines one deny entry that denies all packets that have a destination TCP port number greater than 5000 and a second deny entry that denies packets that have a source UDP port number less than 5000. The second deny entry also logs all matches to the console.
- Defines a permit entry to permit all ICMP packets and another permit entry that allows all other traffic. The second permit entry is necessary because an implicit deny-all condition is at the end of each IPv6 access list.
- Applies the access list CISCO to outbound traffic on a Layer 3 interface.

```
Switch(config)# ipv6 access-list CISCO
Switch(config-ipv6-acl)# deny tcp any any gt 5000
Switch config-ipv6-acl)# deny ::/0 lt 5000 ::/0 log
Switch(config-ipv6-acl)# permit icmp any any
Switch(config-ipv6-acl)# permit any any
Switch(config-ipv6-acl)# exit
Switch(config)# interface gigabitethernet 0/3
Switch(config-if)# no switchport
Switch(config-if)# ipv6 address 2001::/64 eui-64
Switch(config-if)# ipv6 traffic-filter CISCO out
```

Configuring Embedded Event Manager

Embedded Event Manager (EEM) is a distributed and customized approach to event detection and recovery within a Cisco IOS device. EEM offers the ability to monitor events and take informational, corrective, or any other EEM action when the monitored events occur or when a threshold is reached. An EEM policy defines an event and the actions to be taken when that event occurs.

This chapter describes how to configure EEM and how to use it to monitor and manage the Cisco Industrial Ethernet Switches, hereafter referred to as *switch*.

Note: For complete syntax and usage information for the commands used in this chapter, see the documents listed in the Related Documents, page 838.

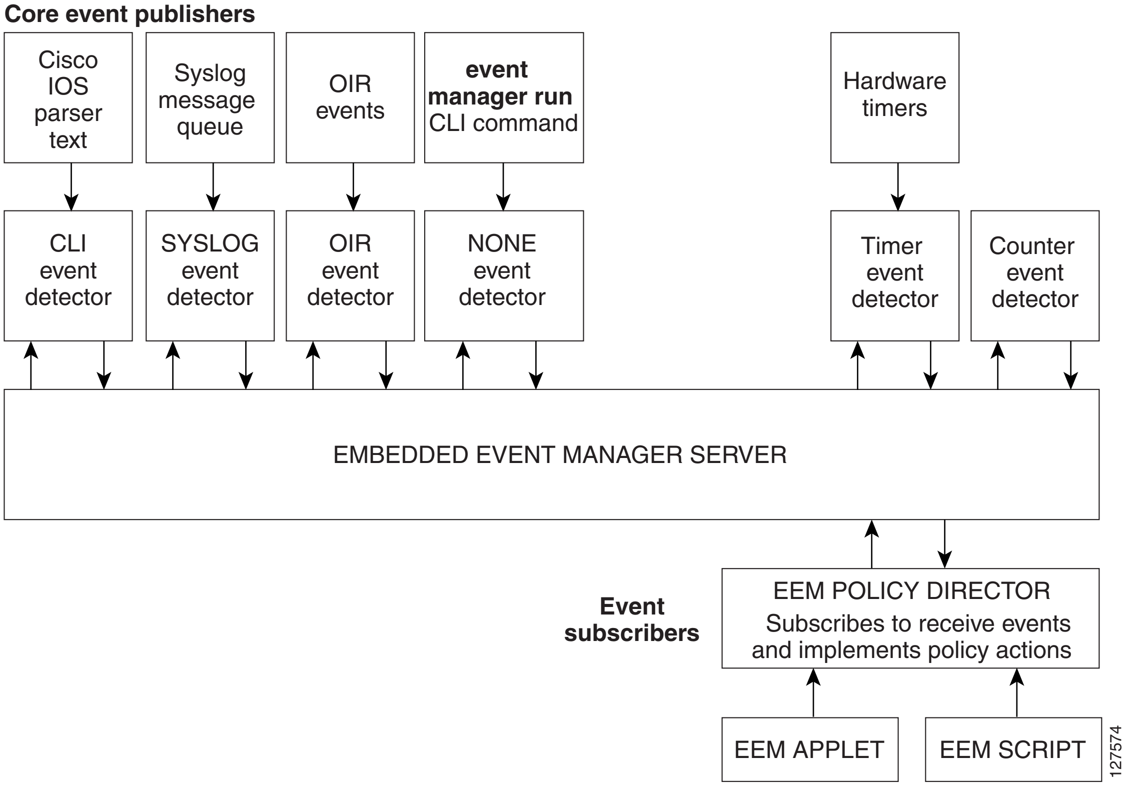
This chapter includes these sections:

Information About Embedded Event Manager

EEM monitors key system events and then acts on them through a set policy. This policy is a programmed script that you can use to customize a script to invoke an action based on a given set of events occurring. The script generates actions such as generating custom syslog or Simple Network Management Protocol (SNMP) traps, invoking CLI commands, forcing a failover, and so forth. The event management capabilities of EEM are useful because not all event management can be managed from the switch and because some problems compromise communication between the switch and the external network management device. Network availability is improved if automatic recovery actions are performed without rebooting the switch.

Figure 98 on page 832 shows the relationship between the EEM server, the core event publishers (event detectors), and the event subscribers (policies). The event publishers screen events and when there is a match on an event specification that is provided by the event subscriber. Event detectors notify the EEM server when an event occurs. The EEM policies then implement recovery based on the current state of the system and the actions specified in the policy for the given event.

Figure 98 Embedded Event Manager Core Event Detectors



See *EEM Configuration for Cisco Integrated Services Router Platforms Guide* for examples of EEM deployment.

This section includes the following topics:

- Event Detectors, page 832
- Embedded Event Manager Actions, page 833
- Embedded Event Manager Policies, page 834
- Embedded Event Manager Environment Variables, page 834
- EEM 3.2, page 834

Event Detectors

EEM software programs known as event detectors determine when an EEM event occurs. Event detectors are separate systems that provide an interface between the agent being monitored, for example SNMP, and the EEM polices where an action can be implemented.

EEM allows these event detectors:

- Application-specific event detector—Allows any EEM policy to publish an event.
- IOS CLI event detector—Generates policies based on the commands entered through the CLI.
- Generic Online Diagnostics (GOLD) event detector—Publishes an event when a GOLD failure event is detected on a specified card and subcard.
- Counter event detector—Publishes an event when a named counter crosses a specified threshold.

- Interface counter event detector—Publishes an event when a generic Cisco IOS interface counter for a specified interface crosses a defined threshold. A threshold can be specified as an absolute value or an incremental value. For example, if the incremental value is set to 50, an event would be published when the interface counter increases by 50.

  This detector also publishes an event about an interface based on the rate of change for the entry and exit values.

- None event detector—Publishes an event when the **event manager run** CLI command executes an EEM policy. EEM schedules and runs policies on the basis on an event specification within the policy itself. An EEM policy must be manually identified and registered before the **event manager run** command executes.

- Online insertion and removal event detector—Publishes an event when a hardware insertion or removal (OIR) event occurs.

- Remote procedure call (RPC) event detector—Invokes EEM policies from outside the switch over an encrypted connecting using Secure Shell (SSH) and uses Simple Object Access Protocol (SOAP) data encoding for exchanging XML-based messages. It also runs EEM policies and then gets the output in a SOAP XML-formatted reply.

- SNMP event detector—Allows a standard SNMP MIB object to be monitored and an event to be generated when

  - The object matches specified values or crosses specified thresholds.

  - The SNMP delta value, the difference between the monitored Object Identifier (OID) value at the beginning the period and the actual OID value when the event is published, matches a specified value.

- SNMP notification event detector—Intercepts SNMP trap and inform messages received by the switch. The event is generated when an incoming message matches a specified value or crosses a defined threshold.

- Syslog event detector—Allows for screening syslog messages for a regular expression pattern match. The selected messages can be further qualified, requiring that a specific number of occurrences be logged within a specified time. A match on a specified event criteria triggers a configured policy action.

- Timer event detector—Publishes events for the following different types of timers:

  - An absolute-time-of-day timer publishes an event when a specified absolute date and time occurs.

  - A countdown timer publishes an event when a timer counts down to zero.

  - A watchdog timer publishes an event when a timer counts down to zero. The timer automatically resets itself to its initial value and starts to count down again.

  - A CRON timer publishes an event by using a UNIX standard CRON specification to define when the event is to be published. A CRON timer never publishes events more than once per minute.

- Watchdog event detector (IOSWDSysMon)— Publishes an event when one of these events occurs:

  - CPU utilization for a Cisco IOS process crosses a threshold.

  - Memory utilization for a Cisco IOS process crosses a threshold.

  Two events can be monitored at the same time, and the event publishing criteria requires that one or both events cross their specified thresholds.

Embedded Event Manager Actions

These actions occur in response to an event:

- Modifying a named counter.

- Publishing an application-specific event.

- Generating an SNMP trap.
- Generating prioritized syslog messages.
- Reloading the Cisco IOS software.

Embedded Event Manager Policies

EEM can monitor events and provide information, or take corrective action when the monitored events occur or a threshold is reached. An EEM policy is an entity that defines an event and the actions to be taken when that event occurs.

There are two types of EEM policies: an applet or a script. An applet is a simple policy that is defined within the CLI configuration. It is a concise method for defining event screening criteria and the actions to be taken when that event occurs. Scripts are defined on the networking device by using an ASCII editor. The script, which can be a bytecode (.tbc) and text (.tcl) script, is then copied to the networking device and registered with EEM. You can also register multiple events in a .tcl file.

Cisco enhancements to TCL in the form of keyword extensions facilitate the development of EEM policies. These keywords identify the detected event, the subsequent action, utility information, counter values, and system information.

For complete information on configuring EEM policies and scripts, see *Embedded Event Manager Configuration Guide, Cisco IOS Release 15M&T*.

Embedded Event Manager Environment Variables

EEM uses environment variables in EEM policies. These variables are defined in an EEM policy tool command language (TCL) script by running a CLI command and the **event manager environment** command.

User-defined variables

Defined by the user for a user-defined policy.

- Cisco-defined variables

  Defined by Cisco for a specific sample policy.

- Cisco built-in variables (available in EEM applets)

  Defined by Cisco and can be read-only or read-write. The read-only variables are set by the system before an applet starts to execute. The single read-write variable, _exit_status, allows you to set the exit status for policies triggered from synchronous events.

Cisco-defined environment variables and Cisco system-defined environment variables might apply to one specific event detector or to all event detectors. Environment variables that are user-defined or defined by Cisco in a sample policy are set by using the **event manager environment** global configuration command. You must defined the variables in the EEM policy before you register the policy.

For information about the environmental variables that EEM supports, see *Embedded Event Manager Configuration Guide, Cisco IOS Release 15M&T*.

EEM 3.2

EEM 3.2 introduces these event detectors:

- Neighbor Discovery—Provides the ability to publish a policy to respond to automatic neighbor detection when:
  - a Cisco Discovery Protocol (CDP) cache entry is added, deleted, or updated.
  - a Link Layer Discovery Protocol (LLDP) cache entry is added, deleted or updated.

  - an interface link status changes.
  - an interface line status changes.

- Identity—Generates an event when AAA authorization and authentication is successful, when failure occurs, or after normal user traffic on the port is allowed to flow.
- Mac-Address-Table—Generates an event when a MAC address is learned in the MAC address table.

  Note: The Mac-Address-Table event detector is supported only on switch platforms and can be used only on Layer 2 interfaces where MAC addresses are learned. Layer 3 interfaces do not learn addresses, and routers do not usually support the MAC address-table infrastructure needed to notify EEM of a learned MAC address.

EEM 3.2 also introduces CLI commands to support the applets to work with the new event detectors.

Prerequisites

- Review the Information About Embedded Event Manager, page 831.
- If the **action snmp-trap** command is used, the **snmp-server enable traps event-manager** command must be enabled to permit SNMP traps to be sent from the Cisco IOS device to the SNMP server. Other relevant **snmp-server** commands must also be configured; for details see the **action snmp-trap** command page.

Guidelines and Limitations

The EEM feature is supported with both Lanbase and IP Services license starting with the 15.2(4)EC release for the IE 4010 and with the15.2(5)E release for IE 4000 and IE 5000. Prior to the 15.2(5)E release, IP Services license was required on the IE 4000 and IE 5000 platforms.

For complete information about configuring embedded event manager, see *Embedded Event Manager Configuration Guide, Cisco IOS Release 15M&T*.

Default Settings

No EEM policies are registered.

Configuring Embedded Event Manager

- Registering and Defining an Embedded Event Manager Applet, page 835
- Registering and Defining an Embedded Event Manager TCL Script, page 836

Registering and Defining an Embedded Event Manager Applet

BEFORE YOU BEGIN

Review the Information About Embedded Event Manager, page 831.

DETAILED STEPS

Note: Only one event applet command is allowed in an EEM applet. Multiple action applet commands are permitted. If you do not specify the **no event** and **no action** commands, the applet is removed when you exit configuration mode.

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**event manager applet** *applet-name*	Register the applet with EEM and enter applet configuration mode.
3.	**event snmp oid** *oid-value* **get-type** {**exact** | **next**} **entry-op** {**gt** | **ge** | **eq** | **ne** | **lt** | **le**} **entry-val** *entry-val* [**exit-comb** {**or** |**and**}] [**exit-op** {**gt** | **ge** | **eq** | **ne** | **lt** | **le**}] [**exit-val exit-val**] [**exit-time** *exit-time-val*] **poll-interval** *poll-int-val*	Specify the event criteria that causes the EEM applet to run. (Optional) Exit criteria. If exit criteria are not specified, event monitoring is re-enabled immediately.
4.	**action label syslog** [**priority** *priority-level*] **msg** *msg-text*	Specify the action when an EEM applet is triggered. Repeat this action to add other CLI commands to the applet. ■ (Optional) The priority keyword specifies the priority level of the syslog messages. If selected, you need to define the priority-level argument. ■ For *msg-text*, the argument can be character text, an environment variable, or a combination of the two.
5.	**end**	Exit applet configuration mode and return to privileged EXEC mode**.**

EXAMPLE

The following example shows how to configure an EEM applet that runs when there is an exact match on the value of a specified SNMP object ID that represents the amount of current process memory.

```
Switch(config-applet)# event snmp oid 1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type exact entry-op lt entry-val
5120000 poll-interval 10
```

These examples show actions that are taken in response to an EEM event:

```
Switch(config-applet)# action 1.0 syslog priority critical msg "Memory exhausted; current available
memory is $_snmp_oid_val bytes"

Switch (config-applet)# action 2.0 force-switchover
```

Registering and Defining an Embedded Event Manager TCL Script

BEFORE YOU BEGIN

Review the Information About Embedded Event Manager, page 831.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
1.	**show event manager environment** [**all** | *variable-name*]	(Optional) The **show event manager environment** command displays the name and value of the EEM environment variables. ■ (Optional) The **all** keyword displays the EEM environment variables. ■ (Optional) The *variable-name* argument displays information about the specified environment variable.
2.	**configure terminal**	Enter global configuration mode.
3.	**event manager environment** *variable-name string*	Configure the value of the specified EEM environment variable. Repeat this step for all the required environment variables.
4.	**event manager policy** *policy-file-name* [**type system**] [**trap**]	Register the EEM policy to run when the specified event defined within the policy occurs.
5.	**exit**	Exit global configuration mode and return to privileged EXEC mode.

EXAMPLE

This example shows the sample output for the show event manager environment command:

```
Switch# show event manager environment all
No.  Name                          Value
1    _cron_entry                   0-59/2 0-23/1 * * 0-6
2    _show_cmd                     show ver
3    _syslog_pattern               .*UPDOWN.*Ethernet1/0.*
4    _config_cmd1                  interface Ethernet1/0
5    _config_cmd2                  no shut
```

This example shows a CRON timer environment variable, which is assigned by the software, to be set to every second minute, every hour of every day:

```
Switch (config)# event manager environment_cron_entry 0-59/2 0-23/1 * * 0-6
```

This example shows the sample EEM policy named *tm_cli_cmd.tcl* registered as a system policy. The system policies are part of the Cisco IOS image. User-defined TCL scripts must first be copied to flash memory.

```
Switch (config)# event manager policy tm_cli_cmd.tcl type system
```

Verifying Configuration

To display information about EEM, including EEM registered policies and EEM history data, see *Cisco IOS Embedded Event Manager Command Reference*.

Configuration Example

This example shows the output for EEM when one of the fields specified by an SNMP object ID crosses a defined threshold:

```
Switch(config-applet)# event snmp oid 1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type exact entry-op lt entry-val
5120000 poll-interval 10
```

These examples show actions that are taken in response to an EEM event:

```
Switch(config-applet)# action 1.0 syslog priority critical msg "Memory exhausted; current available
memory is $_snmp_oid_val bytes"
```

```
Switch (config-applet)# action 2.0 force-switchover
```

This example shows the sample output for the show event manager environment command:

```
Switch# show event manager environment all
No.  Name                           Value
1    _cron_entry                    0-59/2 0-23/1 * * 0-6
2    _show_cmd                      show ver
3    _syslog_pattern                .*UPDOWN.*Ethernet1/0.*
4    _config_cmd1                   interface Ethernet1/0
5    _config_cmd2                   no shut
```

This example shows a CRON timer environment variable, which is assigned by the software, to be set to every second minute, every hour of every day:

```
Switch (config)# event manager environment_cron_entry 0-59/2 0-23/1 * * 0-6
```

This example shows the sample EEM policy named *tm_cli_cmd.tcl* registered as a system policy. The system policies are part of the Cisco IOS image. User-defined TCL scripts must first be copied to flash memory.

```
Switch (config)# event manager policy tm_cli_cmd.tcl type system
```

Related Documents

- Cisco IOS Master Command List, All Releases
- Cisco IOS Embedded Event Manager Command Reference
- Cisco IOS 15.2M&T Command References, Network Management
- Embedded Event Manager Configuration Guide, Cisco IOS Release 15M&T

Configuring IP Unicast Routing

This chapter describes how to configure IP Version 4 (IPv4) unicast routing on the Cisco Industrial Ethernet Switches, hereafter referred to as switch.

Note: Dynamic routing protocols are only supported on switches running IP Services feature set. Static routing is supported on Lan Base feature set.

For more detailed IPv4 unicast configuration information and complete syntax and usage information for the commands used in this chapter, see documents listed in the Related Documents, page 966.

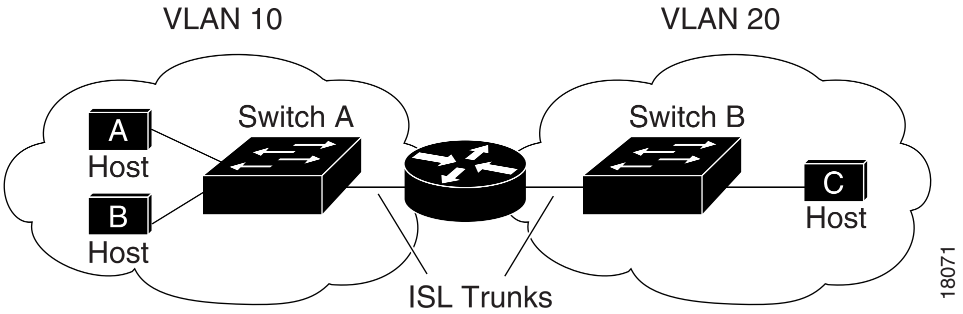
This chapter includes the following sections:

Information About IP Routing

In an IP network, each subnetwork is mapped to an individual VLAN. However, network devices in different VLANs cannot communicate with one another without a Layer 3 device (router) to route traffic between the VLAN, referred to as inter-VLAN routing. You configure one or more routers to route traffic to the appropriate destination VLAN.

Figure 99 on page 840 shows a basic routing topology. Switch A is in VLAN 10, and Switch B is in VLAN 20. The router has an interface in each VLAN.

Figure 99 Routing Topology Example



When Host A in VLAN 10 needs to communicate with Host B in VLAN 10, it sends a packet 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, finds the correct outgoing interface, and forwards the packet on the VLAN 20 interface to Switch B. Switch B receives the packet and forwards it to Host C.

Types of Routing

Routers and Layer 3 switches can route packets in the following ways:

- By using default routing–sending traffic with a destination unknown to the router to a default outlet or destination.
- By using preprogrammed static routes for the traffic

  Static unicast routing forwards packets from predetermined ports through a single path into and out of a network. Static routing does not automatically respond to changes in the network and therefore, might result in unreachable destinations.
- By dynamically calculating routes by using a routing protocol

  Dynamic routing protocols are used by routers to dynamically calculate the best route for forwarding traffic. Routing protocols supported by the switch are Routing Information Protocol (RIP), Border Gateway Protocol (BGP), Open Shortest Path First (OSPF) protocol, Enhanced IGRP (EIGRP), System-to-Intermediate System (IS-IS), and Bidirectional Forwarding Detection (BFD).

Prerequisites

- In order to use dynamic routing protocols, an IP Services License is needed.
- To support VLAN interfaces, create and configure VLANs on the switch, and assign VLAN membership to Layer 2 interfaces.
- By default, IPv4 routing is disabled on the switch, and you must enable it before routing can take place. See Enabling IPv4 Unicast Routing, page 858.
- We recommend that you configure the BFD interval parameters on an interface before configuring the routing protocol commands, especially when using EIGRP. For information about BFD, see Configuring BFD, page 925.

Guidelines and Limitations

- In the following procedures, the specified interface must be one of these Layer 3 interfaces:
  - A routed port: a physical port configured as a Layer 3 port by using the **no switchport** interface configuration command.

  - A switch virtual interface (SVI): a VLAN interface created by using the **interface vlan** *vlan_id* global configuration command and by default a Layer 3 interface.
  - An EtherChannel port channel in Layer 3 mode: a port-channel logical interface created by using the **interface port-channel** *port-channel-number* global configuration command and binding the Ethernet interface into the channel group.

- The switch does not support tunnel interfaces for unicast routed traffic.
- All Layer 3 interfaces on which routing will occur must have IP addresses assigned to them. See Assigning IP Addresses to Network Interfaces, page 842.
- A Layer 3 switch can have an IP address assigned to each routed port and SVI. The number of routed ports and SVIs that you can configure is not limited by software. However, the interrelationship between this number and the number and volume of features being implemented might have an impact on CPU utilization because of hardware limitations.

  To support IPv4 routing, use the **sdm prefer default** global configuration command to set the Switch Database Management (sdm) feature to balance resources. For more information on the SDM templates, see the **sdm prefer** command in the command reference listed in the Related Documents, page 966.

Steps for Configuring Routing

Configuring IPv4 routing consists of several main procedures:

- Configure Layer 3 interfaces.
- Enable IPv4 routing on the switch.
- Assign IPv4 addresses to the Layer 3 interfaces.
- Enable selected routing protocols on the switch.
- Configure routing protocol parameters (optional).

Configuring IP Addressing

IP routing requires that Layer 3 network interfaces are assigned IP addresses to enable the interfaces and to allow communication with the hosts on interfaces that use IP. These sections describe how to configure various IP addressing features. Assigning IP addresses to the interface is required; the other procedures are optional.

Default Addressing Configuration

Feature	Default Setting
IP address	None defined.
ARP	No permanent entries in the Address Resolution Protocol (ARP) cache. Encapsulation: Standard Ethernet-style ARP. Timeout: 14400 seconds (4 hours).
IP broadcast address	255.255.255.255 (all ones).
IP classless routing	Enabled.
IP default gateway	Disabled.
IP directed broadcast	Disabled (all IP directed broadcasts are dropped).
IP domain	Domain list: No domain names defined. Domain lookup: Enabled. Domain name: Enabled.
IP forward-protocol	If a helper address is defined or User Datagram Protocol (UDP) flooding is configured, UDP forwarding is enabled on default ports. Any-local-broadcast: Disabled. Turbo-flood: Disabled.
IP helper address	Disabled.
IP host	Disabled.
IRDP	Disabled. Defaults when enabled: ■ Broadcast IRDP advertisements. ■ Maximum interval between advertisements: 600 seconds. ■ Minimum interval between advertisements: 0.75 times max interval. ■ Preference: 0.
IP proxy ARP	Enabled.
IP routing	Disabled.
IP subnet-zero	Disabled.

Assigning IP Addresses to Network Interfaces

An IP address identifies a location to which IP packets can be sent. An interface can have one primary IP address. A mask identifies the bits that denote the network number in an IP address. When you use the mask to subnet a network, the mask is referred to as a subnet mask.

BEFORE YOU BEGIN

To receive an assigned network number, contact your Internet service provider.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
3.	**no shutdown**	Enable the interface if necessary. User network interfaces (UNIs) and enhanced network interfaces (ENIs) are disabled by default; network node interfaces (NNIs) are enabled by default.
4.	**no switchport**	Remove the interface from Layer 2 configuration mode (if it is a physical interface).
5.	**ip address** *ip-address subnet-mask*	Configure the IP address and IP subnet mask.
6.	**end**	Return to privileged EXEC mode.
7.	**show interfaces** [*interface-id*] **show ip interface** [*interface-id*] **show running-config interface** [*interface-id*]	Verify your entries.
8.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface gi0/2
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.1.5.1 255.255.255.0
Switch(config-if)# end
```

Enabling Subnet Zero

Enabling subnet zero provides the ability to configure and route to subnet 0 subnets.

You can use the all ones subnet (131.108.255.0) and even though it is discouraged, you can enable the use of subnet zero if you need the entire subnet space for your IP address.

BEFORE YOU BEGIN

Subnetting with a subnet address of zero is strongly discouraged because of the problems that can arise if a network and a subnet have the same addresses. For example, if network 131.108.0.0 is subnetted as 255.255.255.0, subnet zero would be written as 131.108.0.0, which is the same as the network address.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip subnet-zero**	Enable the use of subnet zero for interface addresses and routing updates.
3.	**end**	Return to privileged EXEC mode.
4.	**show running-config**	Verify your entry.
5.	**copy running-config startup-config**	(Optional) Save your entry in the configuration file.

Use the **no ip subnet-zero** global configuration command to restore the default and disable the use of subnet zero.

EXAMPLE

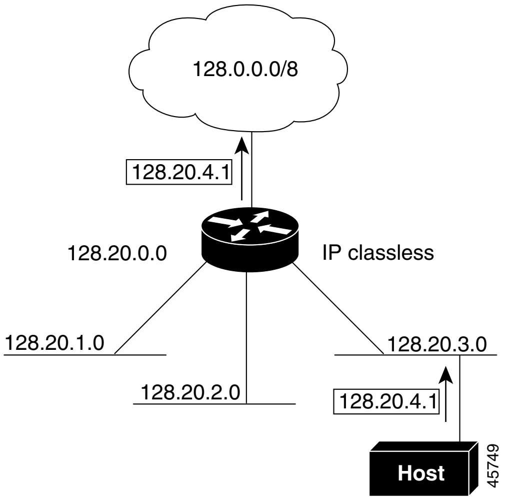
```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# ip subnet-zero
Switch(config)# end
```

Classless Routing

By default, classless routing behavior is enabled on the switch when it is configured to route. With classless routing, if a router receives packets for a subnet of a network with no default route, the router forwards the packet to the best supernet route. A *supernet* consists of contiguous blocks of Class C address spaces used to simulate a single, larger address space and is designed to relieve the pressure on the rapidly depleting Class B address space.

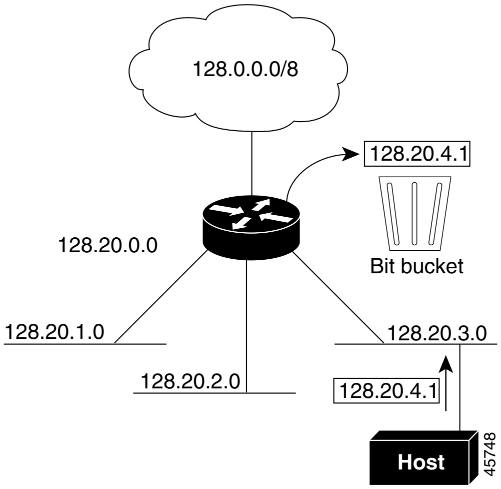
In Figure 100 on page 844, classless routing is enabled. When the host sends a packet to 120.20.4.1, instead of discarding the packet, the router forwards it to the best supernet route. If you disable classless routing and a router receives packets destined for a subnet of a network with no network default route, the router discards the packet.

Figure 100 IP Classless Routing



In Figure 101 on page 845, the router in network 128.20.0.0 is connected to subnets 128.20.1.0, 128.20.2.0, and 128.20.3.0. If the host sends a packet to 120.20.4.1, because there is no network default route, the router discards the packet.

Figure 101 No IP Classless Routing



To prevent the switch from forwarding packets destined for unrecognized subnets to the best supernet route possible, you can disable classless routing behavior.

BEFORE YOU BEGIN

Review the Information About IP Routing, page 839.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**no ip classless**	Disable classless routing behavior.
3.	**end**	Return to privileged EXEC mode.
4.	**show running-config**	Verify your entry.
5.	**copy running-config startup-config**	(Optional) Save your entry in the configuration file.

To restore the default and have the switch forward packets destined for a subnet of a network with no network default route to the best supernet route possible, use the **ip classless** global configuration command.

EXAMPLE

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# no ip classless
Switch(config)# end
```

Configuring Address Resolution Methods

You can control interface-specific handling of IP by using address resolution. A device using IP can have both a local address or MAC address, which uniquely defines the device on its local segment or LAN, and a network address, which identifies the network to which the device belongs. To communicate with a device on Ethernet, the software must learn the MAC address of the device. The process of learning the MAC address from an IP address is called *address resolution*. The process of learning the IP address from the MAC address is called *reverse address resolution*.

The switch can use these forms of address resolution:

- Address Resolution Protocol (ARP) is used to associate IP address with MAC addresses. Taking an IP address as input, ARP learns the associated MAC address and then stores the IP address/MAC address association in an ARP cache for rapid retrieval. Then the IP datagram is encapsulated in a link-layer frame and sent over the network. Encapsulation of IP datagrams and ARP requests or replies on IEEE 802 networks other than Ethernet is specified by the Subnetwork Access Protocol (SNAP).
- Proxy ARP helps hosts with no routing tables learn the MAC addresses of hosts on other networks or subnets. If the switch (router) receives an ARP request for a host that is not on the same interface as the ARP request sender, and if the router has all of its routes to the host through other interfaces, it generates a proxy ARP packet giving its own local data link address. The host that sent the ARP request then sends its packets to the router, which forwards them to the intended host.

The switch also uses the Reverse Address Resolution Protocol (RARP), which functions the same as ARP does, except that the RARP packets request an IP address instead of a local MAC address. Using RARP requires a RARP server on the same network segment as the router interface. Use the **ip rarp-server** *address* interface configuration command to identify the server.

For more information on RARP, see *IP Addressing: ARP Configuration Guide, Cisco IOS Release 15M&T*.

You can perform these tasks to configure address resolution:

- Defining a Static ARP Cache, page 846
- Setting ARP Encapsulation, page 847
- Enabling Proxy ARP, page 848

Defining a Static ARP Cache

ARP and other address resolution protocols provide dynamic mapping between IP addresses and MAC addresses. Because most hosts support dynamic address resolution, you usually do not need to specify static ARP cache entries. If you must define a static ARP cache entry, you can do so globally, which installs a permanent entry in the ARP cache that the switch uses to translate IP addresses into MAC addresses. Optionally, you can also specify that the switch respond to ARP requests as if it were the owner of the specified IP address. If you do not want the ARP entry to be permanent, you can specify a timeout period for the ARP entry.

BEFORE YOU BEGIN

Review the Configuring Address Resolution Methods, page 845.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**arp** *ip-address hardware-address type*	Globally associate an IP address with a MAC (hardware) address in the ARP cache, and specify encapsulation type as one of these: ■ **arpa**–ARP encapsulation for Ethernet interfaces ■ **snap**–Subnetwork Address Protocol encapsulation for Token Ring and FDDI interfaces ■ **sap**–HP's ARP type
3.	**arp** *ip-address hardware-address type* [*alias*]	(Optional) Specify that the switch respond to ARP requests as if it were the owner of the specified IP address.
4.	**interface** *interface-id*	Enter interface configuration mode, and specify the interface to configure.
5.	**no shutdown**	Enable the interface if necessary. By default, UNIs and ENIs are disabled and NNIs are enabled.
6.	**arp timeout** *seconds*	(Optional) Set the length of time an ARP cache entry will stay in the cache. The default is 14400 seconds (4 hours). The range is 0 to 2147483 seconds.
7.	**end**	Return to privileged EXEC mode.
8.	**show interfaces** [*interface-id*]	Verify the type of ARP and the timeout value used on all interfaces or a specific interface.
9.	**show arp** or **show ip arp**	View the contents of the ARP cache.
10.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove an entry from the ARP cache, use the **no arp** *ip-address hardware-address type* global configuration command. To remove all nonstatic entries from the ARP cache, use the **clear arp-cache** privileged EXEC command**.**

EXAMPLE

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# arp 10.0.0.0 aabb.cc03.8200 arpa
Switch(config)# end
```

Setting ARP Encapsulation

By default, Ethernet ARP encapsulation (represented by the **arpa** keyword) is enabled on an IP interface. You can change the encapsulation methods to SNAP if required by your network.

BEFORE YOU BEGIN

The encapsulation type specified in this procedure should match the encapsulation type specified in the Defining a Static ARP Cache, page 846.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
3.	**no shutdown**	Enable the interface if necessary. By default, UNIs and ENIs are disabled and NNIs are enabled.
4.	**arp** {**arpa** / **snap**}	Specify the ARP encapsulation method: ■ **arpa**–Address Resolution Protocol ■ **snap**–Subnetwork Address Protocol
5.	**end**	Return to privileged EXEC mode.
6.	**show interfaces** [*interface-id*]	Verify ARP encapsulation configuration on all interfaces or the specified interface.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable an encapsulation type, use the **no arp arpa** or **no arp snap** interface configuration command.

EXAMPLE

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface gi0/2
Switch(config-if)# arp arpa
Switch(config-if)# end
```

Enabling Proxy ARP

By default, the switch uses proxy ARP to help hosts learn MAC addresses of hosts on other networks or subnets. Follow these steps to enable proxy ARP if it has been disabled.

BEFORE YOU BEGIN

Review the Configuring Address Resolution Methods, page 845.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
3.	**no shutdown**	Enable the interface if necessary. By default, UNIs and ENIs are disabled and NNIs are enabled.
4.	**ip proxy-arp**	Enable proxy ARP on the interface.
5.	**end**	Return to privileged EXEC mode.
6.	**show ip interface** [*interface-id*]	Verify the configuration on the interface or all interfaces.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable proxy ARP on the interface, use the **no ip proxy-arp** interface configuration command.

EXAMPLE

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface gi0/2
Switch(config-if)# ip proxy-arp
Switch(config-if)# end
```

Routing Assistance When IP Routing is Disabled

These mechanisms allow the switch to learn about routes to other networks when it does not have IP routing enabled:

- Proxy ARP, page 849
- Default Gateway, page 849
- ICMP Router Discovery Protocol (IRDP), page 850

Proxy ARP

Proxy ARP, the most common method for learning about other routes, enables an Ethernet host with no routing information to communicate with hosts on other networks or subnets. The host assumes that all hosts are on the same local Ethernet and that they can use ARP to learn their MAC addresses. If a switch receives an ARP request for a host that is not on the same network as the sender, the switch evaluates whether it has the best route to that host. If it does, it sends an ARP reply packet with its own Ethernet MAC address, and the host that sent the request sends the packet to the switch, which forwards it to the intended host. Proxy ARP treats all networks as if they are local and performs ARP requests for every IP address.

Proxy ARP is enabled by default. To enable it after it has been disabled, see Enabling Proxy ARP, page 848. Proxy ARP works as long as other routers support it.

Default Gateway

Another method for locating routes is to define a default router or default gateway. All nonlocal packets are sent to this router, which either routes them appropriately or sends an IP Control Message Protocol (ICMP) redirect message back, defining which local router the host should use. The switch caches the redirect messages and forwards each packet as efficiently as possible. A limitation of this method is that there is no means of detecting when the default router has gone down or is unavailable.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip default-gateway** *ip-address*	Set up a default gateway (router).
3.	**end**	Return to privileged EXEC mode.
4.	**show ip redirects**	Display the address of the default gateway router to verify the setting.
5.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip default-gateway** global configuration command to disable this function.

EXAMPLE

```
Switch# configure terminal
```

```
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# ip default-gateway 192.31.7.18
Switch(config)# end
```

ICMP Router Discovery Protocol (IRDP)

Router discovery allows the switch to dynamically learn about routes to other networks using IRDP. IRDP allows hosts to locate routers. When operating as a client, the switch generates router discovery packets. When operating as a host, the switch receives router discovery packets. The switch can also listen to Routing Information Protocol (RIP) routing updates and use this information to infer locations of routers. The switch does not actually store the routing tables sent by routing devices; it merely keeps track of which systems are sending the data. The advantage of using IRDP is that it allows each router to specify both a priority and the time after which a device is assumed to be down if no further packets are received.

Each device discovered becomes a candidate for the default router, and a new highest-priority router is selected when a higher priority router is discovered, when the current default router is declared down, or when a TCP connection is about to time out because of excessive retransmissions.

The only required task for IRDP routing on an interface is to enable IRDP processing on that interface. When enabled, the default parameters apply. You can optionally change any of these parameters.

BEFORE YOU BEGIN

- The **ip irdp multicast** command allows for compatibility with Sun Microsystems Solaris, which requires IRDP packets to be sent out as multicasts. Many implementations cannot receive these multicasts; ensure end-host ability before using this command.
- If you change the **maxadvertinterval** value, the **holdtime** and **minadvertinterval** values also change, so it is important to first change the **maxadvertinterval** value before manually changing either the **holdtime** or **minadvertinterval** values.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
3.	**no shutdown**	Enable the interface if necessary. By default, UNIs and ENIs are disabled and NNIs are enabled.
4.	**ip irdp**	Enable IRDP processing on the interface.
5.	**ip irdp multicast**	(Optional) Send IRDP advertisements to the multicast address (224.0.0.1) instead of IP broadcasts.
6.	**ip irdp holdtime** *seconds*	(Optional) Set the IRDP period for which advertisements are valid. The default is three times the **maxadvertinterval** value. It must be greater than **maxadvertinterval** and cannot be greater than 9000 seconds. If you change the **maxadvertinterval** value, this value also changes.
7.	**ip irdp maxadvertinterval** *seconds*	(Optional) Set the IRDP maximum interval between advertisements. The default is 600 seconds.
8.	**ip irdp minadvertinterval** *seconds*	(Optional) Set the IRDP minimum interval between advertisements. The default is 0.75 times the **maxadvertinterval**. If you change the **maxadvertinterval**, this value changes to the new default (0.75 of **maxadvertinterval**).
9.	**ip irdp preference** *number*	(Optional) Set a device IRDP preference level. The allowed range is -2^{31} to 2^{31}. The default is 0. A higher value increases the router preference level.
10.	**ip irdp address** *address* [*number*]	(Optional) Specify an IRDP address and preference to proxy-advertise.
11.	**end**	Return to privileged EXEC mode.
12.	**show ip irdp**	Verify settings by displaying IRDP values.
13.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip irdp** interface configuration command to disable IRDP routing.

EXAMPLE

```
Switch(config)# interface ethernet 0 !Enable irdp on interface Ethernet 0.
Switch(config-if)# ip irdp
Switch(config-if)# ip irdp multicast !Send IRDP advertisements to the multicast address.
Switch(config-if)# ip irdp preference 900 !Increase router preference from 0 to 900.
Switch(config-if)# ip irdp maxadvertinterval 400 !Set maximum time between advertisements to 400 secs.
Switch(config-if)# ip irdp minadvertinterval 100 !Set minimum time between advertisements to 100 secs.
Switch(config-if)# ip irdp holdtime 6000 !Advertisements are good for 6000 seconds.
Switch(config-if)# ip irdp address 10.108.14.5 !Proxy-advertise 10.108.14.5 with default router
preference.
Switch(config-if)# ip irdp address 10.108.14.6 50 !Proxy-advertise 10.108.14.6 with preference of 50.
```

Configuring Broadcast Packet Handling

After configuring an IP interface address, you can enable routing and configure one or more routing protocols, or you can configure the way the switch responds to network broadcasts. A broadcast is a data packet destined for all hosts on a physical network. The switch supports two kinds of broadcasting:

- A directed broadcast packet is sent to a specific network or series of networks. A directed broadcast address includes the network or subnet fields.
- A flooded broadcast packet is sent to every network.

Note: You can also limit broadcast, unicast, and multicast traffic on Layer 2 interfaces by using the **storm-control** interface configuration command to set traffic suppression levels.

Routers provide some protection from broadcast storms by limiting their extent to the local cable. Bridges (including intelligent bridges), because they are Layer 2 devices, forward broadcasts to all network segments, thus propagating broadcast storms. The best solution to the broadcast storm problem is to use a single broadcast address scheme on a network. In most modern IP implementations, you can set the address to be used as the broadcast address. The switch supports several addressing schemes for forwarding broadcast messages.

- Enabling Directed Broadcast-to-Physical Broadcast Translation, page 852
- Forwarding UDP Broadcast Packets and Protocols, page 854
- Establishing an IP Broadcast Address, page 855
- Flooding IP Broadcasts, page 855

Enabling Directed Broadcast-to-Physical Broadcast Translation

By default, IP-directed broadcasts are not forwarded; they are dropped to make routers less susceptible to denial-of-service attacks. You can enable forwarding of IP-directed broadcasts on an interface where the broadcast becomes a physical (MAC-layer) broadcast. Only those protocols configured by using the **ip forward-protocol** global configuration command are forwarded.

BEFORE YOU BEGIN

You can specify an access list to control which broadcasts are forwarded. Only those IP packets permitted by the access list are eligible to be translated from directed broadcasts to physical broadcasts.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Enter interface configuration mode, and specify the interface to configure.
3.	**no shutdown**	Enable the interface if necessary. By default, UNIs and ENIs are disabled and NNIs are enabled.
4.	**ip directed-broadcast** [*access-list-number*]	Enable directed broadcast-to-physical broadcast translation on the interface. You can include an access list to control which broadcasts are forwarded. When an access list is specified, only IP packets permitted by the access list are eligible to be translated.
5.	**exit**	Return to global configuration mode.
6.	**ip forward-protocol** {**udp** [*port*] **| nd | sdns**}	Specify which protocols and ports the router forwards when forwarding broadcast packets. ■ **udp**–Forward UPD datagrams. *port*: (Optional) Destination port that controls which UDP services are forwarded. ■ **nd**–Forward ND datagrams. ■ **sdns**–Forward SDNS datagrams.
7.	**end**	Return to privileged EXEC mode.
8.	**show ip interface** [*interface-id*] or **show running-config**	Verify the configuration on the interface or all interfaces.
9.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip directed-broadcast** interface configuration command to disable translation of directed broadcast to physical broadcasts. Use the **no ip forward-protocol** global configuration command to remove a protocol or port.

EXAMPLE

The following example enables forwarding of IP directed broadcasts on Ethernet interface 0. The **ip forward-protocol** command using the **udp** keyword without specifying any port numbers allows forwarding of UDP packets on the default ports.

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface ethernet 0
Switch(config-if)# ip directed-broadcast
Switch(config-if)# exit
Switch(config)# ip forward-protocol udp
Switch(config)# end
```

Forwarding UDP Broadcast Packets and Protocols

User Datagram Protocol (UDP) is an IP host-to-host layer protocol that provides a low-overhead, connectionless session between two end systems and does not provide for acknowledgment of received datagrams. Network hosts occasionally use UDP broadcasts to find address, configuration, and name information. If such a host is on a network segment that does not include a server, UDP broadcasts are normally not forwarded. You can configure an interface on a router to forward certain classes of broadcasts to a helper address. You can use more than one helper address per interface.

You can specify a UDP destination port to control which UDP services are forwarded. You can specify multiple UDP protocols. You can also specify the Network Disk (ND) protocol, which is used by older diskless Sun workstations and the network security protocol SDNS.

By default, both UDP and ND forwarding are enabled if a helper address has been defined for an interface.

If you do not specify any UDP ports when you configure the forwarding of UDP broadcasts, you are configuring the router to act as a BOOTP forwarding agent. BOOTP packets carry DHCP information.

BEFORE YOU BEGIN

See the description for the **ip forward-protocol** interface configuration command in the *Cisco IOS IP Application Services Command Reference* for the list of ports that are forwarded by default if you do not specify any UDP ports.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
3.	**no shutdown**	Enable the interface if necessary. By default, UNIs and ENIs are disabled and NNIs are enabled.
4.	**ip helper-address** *address*	Enable forwarding and specify the destination address for forwarding UDP broadcast packets, including BOOTP.
5.	**exit**	Return to global configuration mode.
6.	**ip forward-protocol** {**udp** [*port*] **| nd | sdns**}	Specify which protocols the router forwards when forwarding broadcast packets.
7.	**end**	Return to privileged EXEC mode.
8.	**show ip interface** [*interface-id*] or **show running-config**	Verify the configuration on the interface or all interfaces.
9.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip helper-address** interface configuration command to disable the forwarding of broadcast packets to specific addresses. Use the **no ip forward-protocol** global configuration command to remove a protocol or port.

EXAMPLE

The following example defines a helper address and uses the **ip forward-protocol** command. Using the **udp** keyword without specifying any port numbers will allow forwarding of UDP packets on the default ports.

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface ethernet 0
Switch(config-if)# ip helper-address 10.24.42.2
Switch(config-if)# exit
Switch(config)# ip forward-protocol udp
Switch(config)# end
```

Establishing an IP Broadcast Address

The most popular IP broadcast address (and the default) is an address consisting of all ones (255.255.255.255). However, the switch can be configured to generate any form of IP broadcast address.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Enter interface configuration mode, and specify the interface to configure.
3.	**no shutdown**	Enable the interface if necessary. By default, UNIs and ENIs are disabled and NNIs are enabled.
4.	**ip broadcast-address** *ip-address*	Enter a broadcast address different from the default, for example 128.1.255.255.
5.	**end**	Return to privileged EXEC mode.
6.	**show ip interface** [*interface-id*]	Verify the broadcast address on the interface or all interfaces.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To restore the default IP broadcast address, use the **no ip broadcast-address** interface configuration command.

EXAMPLE

The following example specifies an IP broadcast address of 0.0.0.0:

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface ethernet 0
Switch(config-if)# ip broadcast-address 0.0.0.0
Switch(config-if)# end
```

Flooding IP Broadcasts

You can allow IP broadcasts to be flooded throughout your internetwork in a controlled fashion by using the database created by the bridging STP. Using this feature also prevents loops. To support this capability, bridging must be configured on each interface that is to participate in the flooding. If bridging is not configured on an interface, the interface can receive broadcasts but it never forwards the broadcasts it receives, and the router never uses that interface to send broadcasts received on a different interface.

Packets that are forwarded to a single network address using the IP helper-address mechanism can be flooded. Only one copy of the packet is sent on each network segment.

To be considered for flooding, packets must meet these criteria. (Note that these are the same conditions used to consider packet forwarding using IP helper addresses.)

- The packet must be a MAC-level broadcast.
- The packet must be an IP-level broadcast.

- The packet must be a TFTP, DNS, Time, NetBIOS, ND, or BOOTP packet, or a UDP specified by the **ip forward-protocol udp** global configuration command.
- The time-to-live (TTL) value of the packet must be at least two.

A flooded UDP datagram is given the destination address specified with the **ip broadcast-address** interface configuration command on the output interface. The destination address can be set to any address so it might change as the datagram propagates through the network. The source address is never changed. The TTL value is decremented.

When a flooded UDP datagram is sent out an interface (and the destination address possibly changed), the datagram is handed to the normal IP output routines and is, therefore, subject to access lists, if they are present on the output interface.

BEFORE YOU BEGIN

Ensure that bridging is configured on each interface that is to participate in the flooding.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip forward-protocol spanning-tree**	Use the bridging spanning-tree database to flood UDP datagrams.
3.	**end**	Return to privileged EXEC mode.
4.	**show running-config**	Verify your entry.
5.	**copy running-config startup-config**	(Optional) Save your entry in the configuration file.

Use the **no ip forward-protocol spanning-tree** global configuration command to disable the flooding of IP broadcasts.

EXAMPLE

The following example permits IP broadcasts to be flooded through the internetwork in a controlled fashion:

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# ip forward-protocol spanning-tree
Switch(config)# end
```

Speeding up STP-Based UDP Flooding

In the switch, the majority of packets are forwarded in hardware; most packets do not go through the switch CPU. For those packets that do go to the CPU, you can speed up spanning tree-based UDP flooding by a factor of about four to five times by using turbo-flooding. This feature is supported over Ethernet interfaces configured for ARP encapsulation.

BEFORE YOU BEGIN

Enable the flooding of IP broadcasts as described in the Flooding IP Broadcasts, page 855.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode
2.	**ip forward-protocol turbo-flood**	Use the spanning-tree database to speed up flooding of UDP datagrams.
3.	**end**	Return to privileged EXEC mode.
4.	**show running-config**	Verify your entry.
5.	**copy running-config startup-config**	(Optional) Save your entry in the configuration file.

To disable this feature, use the **no ip forward-protocol turbo-flood** global configuration command.

EXAMPLE

The following example shows how to speed up the flooding of UDP packets using the spanning-tree algorithm:

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# ip forward-protocol turbo-flood
Switch(config)# end
```

Monitoring and Maintaining IP Addressing

When the contents of a particular cache, table, or database have become or are suspected to be invalid, you can remove all its contents by using the **clear** privileged EXEC commands.

Command	Purpose
clear arp-cache	Clear the IP ARP cache and the fast-switching cache.
clear host {*name* | *}	Remove one or all entries from the hostname and the address cache.
clear ip route {*network* [*mask*] |*}	Remove one or more routes from the IP routing table.

You can display specific statistics, such as the contents of IP routing tables, caches, and databases; the reachability of nodes; and the routing path that packets are taking through the network.

Command	Purpose
show arp	Display the entries in the ARP table.
show hosts	Display the default domain name, style of lookup service, name server hosts, and the cached list of hostnames and addresses.
show ip aliases	Display IP addresses mapped to TCP ports (aliases).
show ip arp	Display the IP ARP cache.
show ip interface [*interface-id*]	Display the IP status of interfaces.
show ip irdp	Display IRDP values.
show ip masks *address*	Display the masks used for network addresses and the number of subnets using each mask.
show ip redirects	Display the address of a default gateway.
show ip route [*address* [*mask*]] | [*protocol*]	Display the current state of the routing table.
show ip route summary	Display the current state of the routing table in summary form.

Enabling IPv4 Unicast Routing

By default, the switch is in Layer 2 switching mode and IP routing is disabled. To use the Layer 3 capabilities of the switch, you must enable IP routing.

BEFORE YOU BEGIN

Review the Guidelines and Limitations, page 840.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip routing**	Enable IP routing.
3.	**router** *ip_routing_protocol*	Specify an IP routing protocol. This step might include other commands, such as specifying the networks to route with the **network** (RIP) router configuration command. For information on specific protocols, see sections later in this chapter.
4.	**end**	Return to privileged EXEC mode.
5.	**show running-config**	Verify your entries.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip routing** global configuration command to disable routing.

EXAMPLE

This example shows how to enable IP routing using RIP as the routing protocol:

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# ip routing
Switch(config)# router rip
Switch(config-router)# network 10.0.0.0
Switch(config-router)# end
```

Configuring RIP

The Routing Information Protocol (RIP) is an interior gateway protocol (IGP) used in small, homogeneous networks. It is a distance-vector routing protocol that uses broadcast User Datagram Protocol (UDP) data packets to exchange routing information. You can find detailed information about RIP in *IP Routing Fundamentals,* published by Cisco Press.

Using RIP, the switch sends routing information updates (advertisements) every 30 seconds. If a router does not receive an update from another router for 180 seconds or more, it marks the routes served by that router as unusable. If there is still no update after 240 seconds, the router removes all routing table entries for the non-updating router.

RIP uses hop counts to rate the value of different routes. The hop count is the number of routers that can be traversed in a route. A directly connected network has a hop count of zero; a network with a hop count of 16 is unreachable. This small range (0 to 15) makes RIP unsuitable for large networks.

If the router has a default network path, RIP advertises a route that links the router to the pseudonetwork 0.0.0.0. The 0.0.0.0 network does not exist, but is treated by RIP as a network to implement default routing. The switch advertises the default network if a default was learned by RIP or if the router has a gateway of last resort and RIP is configured with a default metric. RIP sends updates to the interfaces in specified networks. If an interface's network is not specified, it is not advertised in any RIP update.

This section includes the following topics:

- Default RIP Configuration, page 860
- Configuring Basic RIP Parameters, page 860
- Configuring RIP Authentication, page 862
- Configuring Split Horizon, page 863

Default RIP Configuration

Feature	Default Setting
Auto summary	Enabled.
Default-information originate	Disabled.
Default metric	Built-in; automatic metric translations.
IP RIP authentication key-chain	No authentication. Authentication mode: clear text.
IP RIP receive version	According to the **version** router configuration command.
IP RIP send version	According to the **version** router configuration command.
IP RIP triggered	According to the **version** router configuration command.
IP split horizon	Varies with media.
Neighbor	None defined.
Network	None specified.
Offset list	Disabled.
Output delay	0 milliseconds.
Timers basic	■ Update: 30 seconds. ■ Invalid: 180 seconds. ■ Hold-down: 180 seconds. ■ Flush: 240 seconds.
Validate-update-source	Enabled.
Version	Receives RIP Version 1 and 2 packets; sends Version 1 packets.

Configuring Basic RIP Parameters

To configure RIP, you enable RIP routing for a network and optionally configure other parameters. RIP configuration commands are ignored until you configure the network number.

BEFORE YOU BEGIN

Complete the RIP network strategy and planning for your network. For example, you must decide whether to receive and send only RIP Version 1 or RIP Version 2 packets and whether to use RIP authentication. (RIP Version 1 does not support authentication.)

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip routing**	Enable IP routing. (Required only if IP routing is disabled.)
3.	**router rip**	Enable a RIP routing process, and enter router configuration mode.
4.	**network** *network number*	Associate a network with a RIP routing process. You can specify multiple **network** commands. RIP routing updates are sent and received through interfaces only on these networks. **Note:** You must configure a network number for RIP commands to take effect.
5.	**neighbor** *ip-address*	(Optional) Define a neighboring router with which to exchange routing information. This step allows routing updates from RIP (normally a broadcast protocol) to reach nonbroadcast networks.
6.	**offset list** [*access-list number | name*] {**in** | **out**} *offset* [*type number*]	(Optional) Apply an offset list to routing metrics to increase incoming and outgoing metrics to routes learned through RIP. You can limit the offset list with an access list or an interface.
7.	**timers basic** *update invalid holddown flush*	(Optional) Adjust routing protocol timers. Valid ranges for all timers are 0 to 4294967295 seconds. ■ *update*–The time between sending routing updates. The default is 30 seconds. ■ *invalid*–The timer after which a route is declared invalid. The default is 180 seconds. ■ *holddown*–The time before a route is removed from the routing table. The default is 180 seconds. ■ *flush*–The amount of time for which routing updates are postponed. The default is 240 seconds.
8.	**version** {**1 | 2**}	(Optional) Configure the switch to receive and send only RIP Version 1 or RIP Version 2 packets. By default, the switch receives Version 1 and 2 but sends only Version 1. You can also use the interface commands **ip rip** {**send** | **receive**} **version 1** | **2** | **1 2**} to control what versions are used for sending and receiving on interfaces.
9.	**no auto summary**	(Optional) Disable automatic summarization. By default, the switch summarizes subprefixes when crossing classful network boundaries. Disable summarization (RIP Version 2 only) to advertise subnet and host routing information to classful network boundaries.
10.	**no validate-update-source**	(Optional) Disable validation of the source IP address of incoming RIP routing updates. By default, the switch validates the source IP address of incoming RIP routing updates and discards the update if the source address is not valid. Under normal circumstances, disabling this feature is not recommended. However, if you have a router that is off-network and you want to receive its updates, you can use this command.
11.	**output-delay** *delay*	(Optional) Add interpacket delay for RIP updates sent. By default, packets in a multiple-packet RIP update have no delay added between packets. If you are sending packets to a lower-speed device, you can add an interpacket delay in the range of 8 to 50 milliseconds.

	Command	Purpose
12.	**end**	Return to privileged EXEC mode.
13.	**show ip protocols**	Verify your entries.
14.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To turn off the RIP routing process, use the **no router rip** global configuration command.

To display the parameters and current state of the active routing protocol process, use the **show ip protocols** privileged EXEC command. Use the **show ip rip database** privileged EXEC command to display summary address entries in the RIP database.

EXAMPLE

In the following example, RIP updates are sent to all interfaces on network 10.108.0.0 except Ethernet interface 1. However, in this case, a neighbor router configuration command is included. This command permits the sending of routing updates to specific neighbors. One copy of the routing update is generated per neighbor.

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# ip routing
Switch(config)# router rip
Switch(config-router)# network 10.108.0.0
Router(config-router)# passive-interface Ethernet 1
Router(config-router)# neighbor 10.108.20.4
Router(config-router)# end
```

Configuring RIP Authentication

RIP Version 1 does not support authentication. If you are sending and receiving RIP Version 2 packets, you can enable RIP authentication on an interface. The key chain specifies the set of keys that can be used on the interface. If a key chain is not configured, no authentication is performed, not even the default. Therefore, you must also perform the tasks in the Managing Authentication Keys, page 964.

The switch supports two modes of authentication on interfaces for which RIP authentication is enabled: plain text and MD5. The default is plain text.

BEFORE YOU BEGIN

Configure RIP as described in the Configuring Basic RIP Parameters, page 860.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Enter interface configuration mode, and specify the interface to configure.
3.	**no shutdown**	Enable the interface if necessary. By default, UNIs and ENIs are disabled and NNIs are enabled.
4.	**ip rip authentication key-chain** *name-of-chain*	Enable RIP authentication.
5.	**ip rip authentication mode** [**text** | **md5**}	Configure the interface to use plain text authentication (the default) or MD5 digest authentication.
6.	**end**	Return to privileged EXEC mode.
7.	**show running-config interface** [*interface-id*]	Verify your entries.
8.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To restore clear text authentication, use the **no ip rip authentication mode** interface configuration command. To prevent authentication, use the **no ip rip authentication key-chain** interface configuration command.

EXAMPLE

The following example configures the interface to accept and send any key belonging to the key chain named trees and configures the interface to use MD5 authentication:

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface ethernet 0
Switch(config-if)# ip rip authentication key-chain trees
Switch(config-if)# ip rip authentication mode md5
Switch(config-if)# end
```

Configuring Split Horizon

Routers connected to broadcast-type IP networks and using distance-vector routing protocols normally use the split-horizon mechanism to reduce the possibility of routing loops. Split horizon blocks information about routes from being advertised by a router on any interface from which that information originated. This feature can optimize communication among multiple routers when links are broken.

BEFORE YOU BEGIN

In general, Cisco does not recommend disabling split horizon unless you are certain that your application requires disabling it to properly advertise routes.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Enter interface configuration mode, and specify the interface to configure.
3.	**no shutdown**	Enable the interface if necessary. By default, UNIs and ENIs are disabled and NNIs are enabled.
4.	**ip address** *ip-address subnet-mask*	Configure the IP address and IP subnet.
5.	**no ip split-horizon**	Disable split horizon on the interface.
6.	**end**	Return to privileged EXEC mode.
7.	**show ip interface** *interface-id*	Verify your entries.
8.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To enable the split horizon mechanism, use the **ip split-horizon** interface configuration command.

EXAMPLE

The following simple example disables split horizon on a serial link:

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface serial 0
Switch(config-if)# no ip split-horizon
Switch(config-if)# end
```

Configuring Summary Addresses

To configure an interface running RIP to advertise a summarized local IP address pool on a network access server for dial-up clients, use the **ip summary-address rip** interface configuration command.

Note: If split horizon is enabled, neither autosummary nor interface IP summary addresses are advertised.

BEFORE YOU BEGIN

If the interface is in Layer 2 mode (the default), you must enter a **no switchport** interface configuration command before entering the **ip address** interface configuration command.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
3.	**no shutdown**	Enable the interface if necessary. By default, UNIs and ENIs are disabled and NNIs are enabled.
4.	**ip address** *ip-address subnet-mask*	Configure the IP address and IP subnet.
5.	**ip summary-address rip** *ip address ip-network mask*	Configure the IP address to be summarized and the IP network mask.
6.	**no ip split horizon**	Disable split horizon on the interface.
7.	**end**	Return to privileged EXEC mode.
8.	**show ip interface** *interface-id*	Verify your entries.
9.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable IP summarization, use the **no ip summary-address rip** router configuration command.

EXAMPLE

In this example, the major net is 10.0.0.0. The summary address 10.2.0.0 overrides the autosummary address of 10.0.0.0 so that 10.2.0.0 is advertised out interface Gigabit Ethernet port 2, and 10.0.0.0 is not advertised.

```
Switch(config)# router rip
Switch(config-router)# interface gi0/2
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.1.5.1 255.255.255.0
Switch(config-if)# ip summary-address rip 10.2.0.0 255.255.0.0
Switch(config-if)# no ip split-horizon
Switch(config-if)# exit
Switch(config)# router rip
Switch(config-router)# network 10.0.0.0
Switch(config-router)# neighbor 2.2.2.2 peer-group mygroup
Switch(config-router)# end
```

Configuring OSPF

Open Shortest Path First (OSPF) is an Interior Gateway Protocol (IGP) designed expressly for IP networks, supporting IP subnetting and tagging of externally derived routing information. OSPF also allows packet authentication and uses IP multicast when sending and receiving packets.

This section briefly describes how to configure OSPF. For a complete description of the OSPF commands, see the OSPF documents listed in the Related Documents, page 966.

Note: OSPF classifies different media into broadcast, nonbroadcast multiaccess (NBMA), or point-to-point networks. Broadcast and nonbroadcast networks can also be configured as point-to-multipoint networks. The switch supports all these network types.

The Cisco implementation conforms to the OSPF Version 2 specifications with these key features:

- Definition of stub areas is supported.

- Routes learned through any IP routing protocol can be redistributed into another IP routing protocol. At the intradomain level, this means that OSPF can import routes learned through EIGRP and RIP. OSPF routes can also be exported into RIP.
- Plain text and MD5 authentication among neighboring routers within an area is supported.
- Configurable routing interface parameters include interface output cost, retransmission interval, interface transmit delay, router priority, router dead and hello intervals, and authentication key.
- Virtual links are supported.
- Not-so-stubby-areas (NSSAs) per RFC 1587 are supported.

OSPF typically requires coordination among many internal routers, *area border routers* (ABRs) connected to multiple areas, and *autonomous system boundary routers* (ASBRs). The minimum configuration would use all default parameter values, no authentication, and interfaces assigned to areas. If you customize your environment, you must ensure coordinated configuration of all routers.

This section includes the following topics:

Default OSPF Configuration

Feature	Default Setting
Interface parameters	Cost: No default cost predefined. Retransmit interval: 5 seconds. Transmit delay: 1 second. Priority: 1. Hello interval: 10 seconds. Dead interval: 4 times the hello interval. No authentication. No password specified. MD5 authentication disabled.
Area	Authentication type: 0 (no authentication). Default cost: 1. Range: Disabled. Stub: No stub area defined. NSSA: No NSSA area defined.
Auto cost	100 Mbps.
Default-information originate	Disabled. When enabled, the default metric setting is 10, and the external route type default is Type 2.
Default metric	Built-in, automatic metric translation, as appropriate for each routing protocol.
Distance OSPF	dist1 (all routes within an area): 110. dist2 (all routes from one area to another): 110. and dist3 (routes from other routing domains): 110.
OSPF database filter	Disabled. All outgoing link-state advertisements (LSAs) are flooded to the interface.
IP OSPF name lookup	Disabled.
Log adjacency changes	Enabled.
Neighbor	None specified.
Neighbor database filter	Disabled. All outgoing LSAs are flooded to the neighbor.
Network area	Disabled.
NSF[1] awareness	Enabled[2]. Allows Layer 3 switches to continue forwarding packets from a neighboring NSF-capable router during hardware or software changes.
Router ID	No OSPF routing process defined.
Summary address	Disabled.

Feature	Default Setting
Timers LSA group pacing	240 seconds.
Timers shortest path first (spf)	spf delay: 5 seconds. spf-holdtime: 10 seconds.
Virtual link	No area ID or router ID defined. Hello interval: 10 seconds. Retransmit interval: 5 seconds. Transmit delay: 1 second. Dead interval: 40 seconds. Authentication key: no key predefined. Message-digest key (MD5): no key predefined.

1. NSF = Nonstop forwarding

2. OSPF NSF awareness is enabled for IPv4 on switches running the IP services image.

Nonstop Forwarding Awareness

The OSPF NSF Awareness feature is supported for IPv4 in the IP services image. When the neighboring router is NSF-capable, the Layer 3 switch continues to forward packets from the neighboring router during the interval between the primary Route Processor (RP) in a router crashing and the backup RP taking over, or while the primary RP is manually reloaded for a non-disruptive software upgrade.

This feature cannot be disabled. For more information about this feature, see the "Configuring Nonstop Forwarding" chapter in the *High Availability Configuration Guide, Cisco IOS Release 15S*.

Configuring Basic OSPF Parameters

Enabling OSPF requires that you create an OSPF routing process, specify the range of IP addresses to be associated with the routing process, and assign area IDs to be associated with that range.

BEFORE YOU BEGIN

Complete the OSPF network strategy and planning for your network. For example, you must decide whether multiple areas are required.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**router ospf** *process-id*	Enable OSPF routing, and enter router configuration mode. The process ID is an internally used identification parameter that is locally assigned and can be any positive integer. Each OSPF routing process has a unique value.
3.	**network** *address wildcard-mask* **area** *area-id*	Define an interface on which OSPF runs and the area ID for that interface. You can use the wildcard-mask to use a single command to define one or more multiple interfaces to be associated with a specific OSPF area. The area ID can be a decimal value or an IP address.
4.	**end**	Return to privileged EXEC mode.
5.	**show ip protocols**	Verify your entries.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To terminate an OSPF routing process, use the **no router ospf** *process-id* global configuration command.

EXAMPLE

This example shows how to configure an OSPF routing process and assign it a process number of 109:

```
Switch(config)# router ospf 109
Switch(config-router)# network 131.108.0.0 255.255.255.0 area 24
```

Configuring OSPF Interfaces

You can use the **ip ospf** interface configuration commands to modify interface-specific OSPF parameters. You are not required to modify any of these parameters, but some interface parameters (hello interval, dead interval, and authentication key) must be consistent across all routers in an attached network.

Note: The **ip ospf** interface configuration commands are all optional.

BEFORE YOU BEGIN

If you modify these parameters, be sure all routers in the network have compatible values.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
3.	**no shutdown**	Enable the interface if necessary. By default, UNIs and ENIs are disabled and NNIs are enabled.
4.	**ip ospf** *cost*	(Optional) Explicitly specify the cost of sending a packet on the interface.
5.	**ip ospf retransmit-interval** *seconds*	(Optional) Specify the number of seconds between link state advertisement transmissions. The range is 1 to 65535 seconds. The default is 5 seconds.
6.	**ip ospf transmit-delay** *seconds*	(Optional) Set the estimated number of seconds to wait before sending a link state update packet. The range is 1 to 65535 seconds. The default is 1 second.
7.	**ip ospf priority** *number*	(Optional) Set priority to help find the OSPF designated router for a network. The range is from 0 to 255. The default is 1.
8.	**ip ospf hello-interval** *seconds*	(Optional) Set the number of seconds between hello packets sent on an OSPF interface. The value must be the same for all nodes on a network. The range is 1 to 65535 seconds. The default is 10 seconds.
9.	**ip ospf dead-interval** *seconds*	(Optional) Set the number of seconds after the last device hello packet was seen before its neighbors declare the OSPF router to be down. The value must be the same for all nodes on a network. The range is 1 to 65535 seconds. The default is 4 times the hello interval.
10.	**ip ospf authentication-key** *key*	(Optional) Assign a password to be used by neighboring OSPF routers. The password can be any string of keyboard-entered characters up to 8 bytes in length. All neighboring routers on the same network must have the same password to exchange OSPF information.
11.	**ip ospf message digest-key** *keyid* **md5** *key*	(Optional) Enable MDS authentication. ■ *keyid*–An identifier from 1 to 255. ■ *key*–An alphanumeric password of up to 16 bytes.
12.	**ip ospf database-filter all out**	(Optional) Block flooding of OSPF LSA packets to the interface. By default, OSPF floods new LSAs over all interfaces in the same area, except the interface on which the LSA arrives.
13.	**end**	Return to privileged EXEC mode.

	Command	Purpose
14.	**show ip ospf interface** [*interface-name*]	Display OSPF-related interface information.
15.	**show ip ospf neighbor detail**	Display NSF awareness status of neighbor switch. The output matches one of these examples: - *Options is 0x52* *LLS Options is 0x1 (LR)* When both of these lines appear, the neighbor switch is NSF aware. - *Options is 0x42*—This means the neighbor switch is not NSF aware.
16.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** form of these commands to remove the configured parameter value or return to the default value.

EXAMPLE

The following example specifies a cost of 65 and sets the interval between link-state advertisement (LSA) retransmissions to 1 second:

```
Switch# configure terminal
Switch(config)# interface GigabitEthernet 0/0
Switch(config-if)# ip ospf cost 65
Switch(config-if)# ip ospf retransmit-interval 1
Switch(config-if)# end
```

Configuring OSPF Network Types

OSPF classifies different media into the three types of networks by default:

- Broadcast networks (Ethernet, Token Ring, and FDDI)
- Nonbroadcast multiaccess (NBMA) networks (Switched Multimegabit Data Service [SMDS], Frame Relay, and X.25)
- Point-to-point networks (High-Level Data Link Control [HDLC], PPP)

You can also configure network interfaces as either a broadcast or an NBMA network and as point-to point or point-to-multipoint, regardless of the default media type.

Configuring OSPF for Nonbroadcast Networks

Because many routers might be attached to an OSPF network, a designated router is selected for the network. If broadcast capability is not configured in the network, the designated router selection requires special configuration parameters. You need to configure these parameters only for devices that are eligible to become the designated router or backup designated router (in other words, routers with a nonzero router priority value).

BEFORE YOU BEGIN

Complete the OSPF network strategy and planning for your network.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**router ospf** *process-id*	Configure an OSPF routing process and enter router configuration mode.
3.	**neighbor** *ip-address* [**priority** *number*] [**poll-interval** *seconds*]	Specify an OSPF neighbor with neighbor parameters as required. ■ *ip-address*–Enter the interface IP address of the OSPF neighbor. ■ (Optional) **priority** *number*–Specify the router priority value of the nonbroadcast neighbor associated with the IP address. The range is 0 to 255; the default is 0. ■ (Optional) **poll-interval** *seconds*–Specify a number that represents the poll interval time (in seconds). This value should be much larger than the hello interval. The range is 0-4294967295; the default is 120 seconds (2 minutes).
4.	**end**	Return to privileged EXEC mode.
5.	**show ip ospf** [*process-id*]	Display OSPF-related information.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

On point-to-multipoint, nonbroadcast networks, you then use the **neighbor** router configuration command to identify neighbors. Assigning a cost to a neighbor is optional.

EXAMPLE

The following example declares a router at address 192.168.3.4 on a nonbroadcast network, with a priority of 1 and a poll interval of 180 seconds:

```
Switch# configure terminal
Switch(config)# router ospf
Switch(config-router)# neighbor 192.168.3.4 priority 1 poll-interval 180
Switch(config-router)# end
```

Configuring Network Types for OSPF Interfaces

You can configure network interfaces as either broadcast or NBMA and as point-to point or point-to-multipoint, regardless of the default media type.

An OSPF point-to-multipoint interface is defined as a numbered point-to-point interface with one or more neighbors. On point-to-multipoint broadcast networks, specifying neighbors is optional. When you configure an interface as point-to-multipoint when the media does not support broadcast, you should use the **neighbor** command to identify neighbors.

BEFORE YOU BEGIN

Complete the OSPF network strategy and planning for your network.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
3.	**no shutdown**	Enable the interface if necessary. By default, UNIs and ENIs are disabled and NNIs are enabled.
4.	**ip ospf network** {**broadcast** | **non-broadcast** | {**point-to-multipoint** [**non-broadcast**] | **point-to-point**}}	Configure the OSFP network type for the specified interface. Select one of these network types: ■ **broadcast**–Specify an OSPF broadcast multi-access network. ■ **non-broadcast**–Specify an OSPF NBMA network. ■ **point-to-multipoint**–Specify an OSPF point-to-multipoint network. If you do not enter another keyword, the interface is point-to-multipoint for broadcast media. ■ **point-to-multipoint non-broadcast**–Specify an OSPF nonbroadcast point-to-multipoint network. ■ **point-to-point**–Specify an OSPF point-to-point network.
5.	**exit**	Return to global configuration mode.
6.	**router ospf** *process-id*	(Optional for point-to-multipoint; required for point-to-multipoint nonbroadcast) Configure an OSPF routing process and enter router configuration mode.
7.	**neighbor** *ip-address* **cost** *number*	(Optional for point-to-multipoint; required for point-to-multipoint nonbroadcast). Specify a configured OSPF neighbor and assign a cost to the neighbor. ■ *ip-address*–Enter the interface IP address of the OSPF neighbor. ■ **cost** *number*–Specify a cost for the neighbor as an integer from 1 to 65535. **Note:** On point-to-multipoint broadcast networks, specifying a neighbor is optional, but if you do specify a neighbor, you must specify a cost for that neighbor. On point-to-multipoint nonbroadcast neighbors, you must specify a neighbor, but assigning a cost to the neighbor is optional. If not specified, neighbors assume the cost of the interface, based on the **ip ospf cost** interface configuration command.
8.	**end**	Return to privileged EXEC mode.
9.	**show ip ospf interface** [*interface-id*]	Display OSPF-related interface information.
10.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** form of the **ip ospf network** command to return to the default network type for the media.

EXAMPLE

The following example sets your OSPF network as a broadcast network:

```
interface serial 0
 ip address 192.168.77.17 255.255.255.0
 ip ospf network broadcast
 encapsulation frame-relay
```

The following example illustrates a point-to-multipoint network with broadcast:

```
interface serial 0
 ip address 10.0.1.1 255.255.255.0
 encapsulation frame-relay
 ip ospf cost 100
 ip ospf network point-to-multipoint
 frame-relay map ip 10.0.1.3 202 broadcast
 frame-relay map ip 10.0.1.4 203 broadcast
 frame-relay map ip 10.0.1.5 204 broadcast
 frame-relay local-dlci 200
!
router ospf 1
 network 10.0.1.0 0.0.0.255 area 0
 neighbor 10.0.1.5 cost 5
 neighbor 10.0.1.4 cost 10
```

Configuring OSPF Area Parameters

You can optionally configure several OSPF area parameters. These parameters include authentication for password-based protection against unauthorized access to an area, stub areas, and not-so-stubby-areas (NSSAs). *Stub areas* are areas into which information on external routes is not sent. Instead, the area border router (ABR) generates a default external route into the stub area for destinations outside the autonomous system (AS). An NSSA does not flood all LSAs from the core into the area, but can import AS external routes within the area by redistribution.

Route summarization is the consolidation of advertised addresses into a single summary route to be advertised by other areas. If network numbers are contiguous, you can use the **area range** router configuration command to configure the ABR to advertise a summary route that covers all networks in the range.

Note: The OSPF **area** router configuration commands are all optional.

BEFORE YOU BEGIN

Evaluate the following considerations before you implement this feature:

- You can set a Type 7 default route that can be used to reach external destinations. When configured, the router generates a Type 7 default into the NSSA or the NSSA ABR.
- Every router within the same area must agree that the area is NSSA; otherwise, the routers will not be able to communicate.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**router ospf** *process-id*	Enable OSPF routing, and enter router configuration mode.
3.	**area** *area-id* **authentication**	(Optional) Allow password-based protection against unauthorized access to the identified area. The identifier can be either a decimal value or an IP address.
4.	**area** *area-id* **authentication message-digest**	(Optional) Enable MD5 authentication on the area.
5.	**area** *area-id* **stub** [**no-summary**]	(Optional) Define an area as a stub area. The **no-summary** keyword prevents an ABR from sending summary link advertisements into the stub area.
6.	**area** *area-id* **nssa** [**no-redistribution**] [**default-information-originate**] [**no-summary**]	(Optional) Defines an area as a not-so-stubby-area. Every router within the same area must agree that the area is NSSA. Select one of these keywords: ■ **no-redistribution**—Select when the router is an NSSA ABR and you want the **redistribute** command to import routes into normal areas, but not into the NSSA. ■ **default-information-originate**—Select on an ABR to allow importing type 7 LSAs into the NSSA. ■ **no-redistribution**—Select to not send summary LSAs into the NSSA.
7.	**area** *area-id* **range** *address mask*	(Optional) Specify an address range for which a single route is advertised. Use this command only with area border routers.
8.	**end**	Return to privileged EXEC mode.
9.	**show ip ospf** [*process-id*] **show ip ospf** [*process-id* [*area-id*]] **database**	Display information about the OSPF routing process in general or for a specific process ID to verify configuration. Display lists of information related to the OSPF database for a specific router.
10.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** form of these commands to remove the configured parameter value or to return to the default value.

EXAMPLE

The following example mandates authentication for areas 0 and 10.0.0.0 of OSPF routing process 201. Authentication keys are also provided.

```
interface ethernet 0
 ip address 192.168.251.201 255.255.255.0
 ip ospf authentication-key adcdefgh
!
interface ethernet 1
 ip address 10.56.0.201 255.255.0.0
 ip ospf authentication-key ijklmnop
!
router ospf 201
 network 10.0.0.0 0.255.255.255 area 10.0.0.0
 network 192.168.0.0 0.0.255.255 area 0
 area 10.0.0.0 authentication
 area 0 authentication
```

Configuring Other OSPF Parameters

You can optionally configure other OSPF parameters in router configuration mode.

- Route summarization: When redistributing routes from other protocols as described in the Using Route Maps to Redistribute Routing Information, page 953, each route is advertised individually in an external LSA. To help decrease the size of the OSPF link state database, you can use the **summary-address** router configuration command to advertise a single router for all the redistributed routes included in a specified network address and mask.

- Virtual links: In OSPF, all areas must be connected to a backbone area. You can establish a virtual link in case of a backbone-continuity break by configuring two Area Border Routers as endpoints of a virtual link. Configuration information includes the identity of the other virtual endpoint (the other ABR) and the nonbackbone link that the two routers have in common (the transit area). Virtual links cannot be configured through a stub area.

- Default route: When you specifically configure redistribution of routes into an OSPF routing domain, the route automatically becomes an autonomous system boundary router (ASBR). You can force the ASBR to generate a default route into the OSPF routing domain.

- Domain Name Server (DNS) names for use in all OSPF **show** privileged EXEC command displays makes it easier to identify a router than displaying it by router ID or neighbor ID.

- Default Metrics: OSPF calculates the OSPF metric for an interface according to the bandwidth of the interface. The metric is calculated as *ref-bw* divided by bandwidth, where *ref* is 10 by default, and bandwidth (*bw*) is specified by the **bandwidth** interface configuration command. For multiple links with high bandwidth, you can specify a larger number to differentiate the cost on those links.

- Administrative distance is a rating of the trustworthiness of a routing information source, an integer between 0 and 255, with a higher value meaning a lower trust rating. An administrative distance of 255 means the routing information source cannot be trusted at all and should be ignored. OSPF uses three different administrative distances: routes within an area (interarea), routes to another area (interarea), and routes from another routing domain learned through redistribution (external). You can change any of the distance values.

- Passive interfaces: Because interfaces between two devices on an Ethernet represent only one network segment, to prevent OSPF from sending hello packets for the sending interface, you must configure the sending device to be a passive interface. Both devices can identify each other through the hello packet for the receiving interface.

- Route calculation timers: You can configure the delay time between when OSPF receives a topology change and when it starts the shortest path first (SPF) calculation and the hold time between two SPF calculations.

- Log neighbor changes: You can configure the router to send a syslog message when an OSPF neighbor state changes, providing a high-level view of changes in the router.

BEFORE YOU BEGIN

Complete the OSPF network strategy and planning for your network.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**router ospf** *process-id*	Enable OSPF routing, and enter router configuration mode.
3.	**summary-address** *address mask*	(Optional) Specify an address and IP subnet mask for redistributed routes so that only one summary route is advertised.
4.	**area** *area-id* **virtual-link** router-id [**hello-interval** *seconds*] [**retransmit-interval** *seconds*] [**trans**] [[**authentication-key** *key*] | **message-digest-key** *keyid* **md5** *key*]]	(Optional) Establish a virtual link and set its parameters. See Configuring OSPF Interfaces, page 869 for parameter definitions and the Default OSPF Configuration, page 867 for virtual link defaults.
5.	**default-information originate** [**always**] [**metric** *metric-value*] [**metric-type** *type-value*] [**route-map** *map-name*]	(Optional) Force the ASBR to generate a default route into the OSPF routing domain. Parameters are all optional.
6.	**ip ospf name-lookup**	(Optional) Configure DNS name lookup. The default is disabled.
7.	**ip auto-cost reference-bandwidth** *ref-bw*	(Optional) Specify an address range for which a single route will be advertised. Use this command only with area border routers.
8.	**distance ospf** {[**inter-area** *dist1*] [**inter-area** *dist2*] [**external** *dist3*]}	(Optional) Change the OSPF distance values. The default distance for each type of route is 110. The range is 1 to 255.
9.	**passive-interface** *type number*	(Optional) Suppress the sending of hello packets through the specified interface.
10.	**timers throttle spf** *spf-delay spf-holdtime spf-wait*	(Optional) Configure route calculation timers. ■ *spf-delay*—Delay between receiving a change to SPF calculation. The range is from 1 to 600000 miliseconds. ■ *spf-holdtime*—Delay between first and second SPF calculation. The range is form 1 to 600000 in milliseconds. ■ *spf-wait*—Maximum wait time in milliseconds for SPF calculations. The range is from 1 to 600000 in milliseconds.
11.	**ospf log-adj-changes**	(Optional) Send syslog message when a neighbor state changes.
12.	**end**	Return to privileged EXEC mode.
13.	**show ip ospf** [*process-id* [*area-id*]] **database**	Display lists of information related to the OSPF database for a specific router. For some of the keyword options, see Monitoring OSPF, page 879.
14.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

In the following example, the summary address 10.1.0.0 includes address 10.1.1.0, 10.1.2.0, 10.1.3.0, and so on. Only the address 10.1.0.0 is advertised in an external link-state advertisement.

```
Switch(config)# router ospf 201
Switch(config-router)# summary-address 10.1.0.0 255.255.0.0
Switch(config-router)# end
```

Changing LSA Group Pacing

The OSPF LSA group pacing feature allows the router to group OSPF LSAs and pace the refreshing, check-summing, and aging functions for more efficient router use. This feature is enabled by default with a 4-minute default pacing interval, and you will not usually need to modify this parameter. The optimum group pacing interval is inversely proportional to the number of LSAs the router is refreshing, check-summing, and aging. For example, if you have approximately 10,000 LSAs in the database, decreasing the pacing interval would benefit you. If you have a very small database (40 to 100 LSAs), increasing the pacing interval to 10 to 20 minutes might benefit you slightly.

BEFORE YOU BEGIN

Do not change the packet pacing timers unless all other options to meet OSPF packet flooding requirements have been exhausted. Specifically, network operators should prefer summarization, stub area usage, queue tuning, and buffer tuning before changing the default flooding timers. Furthermore, there are no guidelines for changing timer values; each OSPF deployment is unique and should be considered on a case-by-case basis. The network operator assumes the risks associated with changing the default timer values.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**router ospf** *process-id*	Enable OSPF routing, and enter router configuration mode.
3.	**timers pacing lsa-group** *seconds*	Change the group pacing of LSAs.
4.	**end**	Return to privileged EXEC mode.
5.	**show running-config**	Verify your entries.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default value, use the **no timers pacing lsa-group** router configuration command.

EXAMPLE

The following example configures OSPF group packet-pacing updates between LSA groups to occur in 60-second intervals for OSPF routing process 1:

```
Switch(config)# router ospf 1
Switch(config-router)# timers pacing lsa-group 60
```

Configuring a Loopback Interface

OSPF uses the highest IP address configured on the interfaces as its router ID. If this interface is down or removed, the OSPF process must recalculate a new router ID and resend all its routing information out its interfaces. If a loopback interface is configured with an IP address, OSPF uses this IP address as its router ID, even if other interfaces have higher IP addresses. Because loopback interfaces never fail, this provides greater stability. OSPF automatically prefers a loopback interface over other interfaces, and it chooses the highest IP address among all loopback interfaces.

BEFORE YOU BEGIN

The IP address for the loopback interface must be unique and not in use by another interface.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface loopback 0**	Create a loopback interface, and enter interface configuration mode.
3.	**ip address** *address mask*	Assign an IP address to this interface.
4.	**end**	Return to privileged EXEC mode.
5.	**show ip interface**	Verify your entries.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no interface loopback 0** global configuration command to disable the loopback interface.

EXAMPLE

```
Switch(config)# interface loopback 0
Switch(config-if)# ip address 10.108.1.1 255.255.255.0
```

Monitoring OSPF

You can display specific statistics such as the contents of IP routing tables, caches, and databases.

Following are some of the privileged EXEC commands for displaying OSPF statistics. For more **show ip ospf database** privileged EXEC command options and for explanations of fields in the resulting display, see *Cisco IOS IP Routing: OSPF Command Reference*.

Command	Purpose
show ip ospf [*process-id*]	Display general information about OSPF routing processes.
show ip ospf [*process-id*] **database** [**router**] [*link-state-id*] **show ip ospf** [*process-id*] **database** [**router**] [**self-originate**] **show ip ospf** [*process-id*] **database** [**router**] [**adv-router** [*ip-address*]] **show ip ospf** [*process-id*] **database** [**network**] [*link-state-id*] **show ip ospf** [*process-id*] **database** [**summary**] [*link-state-id*] **show ip ospf** [*process-id*] **database** [**asbr-summary**] [*link-state-id*] **show ip ospf** [*process-id*] **database** [**external**] [*link-state-id*] **show ip ospf** [*process-id area-id*] **database** [**database-summary**]	Display lists of information related to the OSPF database.
show ip ospf border-routes	Display the internal OSPF routing ABR and ASBR table entries.
show ip ospf interface [*interface-name*]	Display OSPF-related interface information.
show ip ospf neighbor [*interface-name*] [*neighbor-id*] **detail**	Display OSPF interface neighbor information.
show ip ospf virtual-links	Display OSPF-related virtual links information.

Configuring EIGRP

Enhanced IGRP (EIGRP) is a Cisco proprietary enhanced version of the Interior Gateway Routing Protocol (IGRP). EIGRP uses the same distance vector algorithm and distance information as IGRP; however, the convergence properties and the operating efficiency of EIGRP are significantly improved.

The convergence technology employs an algorithm referred to as the Diffusing Update Algorithm (DUAL), which guarantees loop-free operation at every instant throughout a route computation and allows all devices involved in a topology change to synchronize at the same time. Routers that are not affected by topology changes are not involved in recomputations.

IP EIGRP provides increased network width. With RIP, the largest possible width of your network is 15 hops. Because the EIGRP metric is large enough to support thousands of hops, the only barrier to expanding the network is the transport-layer hop counter. EIGRP increments the transport control field only when an IP packet has traversed 15 routers and the next hop to the destination was learned through EIGRP.

EIGRP has these four basic components:

- *Neighbor discovery and recovery* is the process that routers use to dynamically learn of other routers on their directly attached networks. Routers must also discover when their neighbors become unreachable or inoperative. Neighbor discovery and recovery is achieved by periodically sending small hello packets. As long as hello packets are received, the neighbor is alive and functioning. When this status is determined, the neighboring routers exchange routing information.
- The *reliable transport protocol* is responsible for guaranteed, ordered delivery of EIGRP packets to all neighbors. It supports intermixed transmission of multicast and unicast packets. Some EIGRP packets must be sent reliably, and others need not be. For efficiency, reliability is provided only when necessary. For example, on a multiaccess network that has multicast capabilities, it is not necessary to send hellos reliably to all neighbors individually. Therefore, EIGRP sends a single multicast hello with an indication in the packet informing the receivers that the packet need not be acknowledged. Other types of packets (such as updates) require acknowledgment, which is shown in the packet. To ensure low convergence time, the reliable transport sends multicast packets quickly when there are unacknowledged packets pending.
- The *DUAL finite state machine* handles the decision process for all route computations. It tracks all routes advertised by all neighbors and uses the distance information (known as a metric) to select efficient, loop-free paths. DUAL selects routes to be inserted into a routing table based on feasible successors. A successor is a neighboring router used for packet forwarding that has a least-cost path to a destination that is guaranteed not to be part of a routing loop.

  When there are no feasible successors, but there are neighbors advertising the destination, a recomputation must occur to determine a new successor. The amount of time it takes to recompute the route affects the convergence time. When a topology change occurs, DUAL tests for feasible successors to avoid unnecessary recomputation.
- The *protocol-dependent modules* are responsible for network layer protocol-specific tasks. An example is the IP EIGRP module, which is responsible for sending and receiving EIGRP packets that are encapsulated in IP. It is also responsible for parsing EIGRP packets and informing DUAL of the new information received. Routing decisions are stored in the IP routing table. EIGRP also redistributes routes learned by other IP routing protocols.

This section includes the following topics:

- Default EIGRP Configuration, page 881
- Configuring Basic EIGRP Parameters, page 882
- Configuring EIGRP Interfaces, page 883
- Configuring EIGRP Route Authentication, page 884
- Configuring EIGRP Stub Routing, page 886
- Monitoring and Maintaining EIGRP, page 887

Default EIGRP Configuration

Feature	Default Setting
Auto summary	Enabled. Subprefixes are summarized to the classful network boundary when crossing classful network boundaries.
Default-information	Exterior routes are accepted and default information is passed between EIGRP processes when doing redistribution.
Default metric	Only connected routes and interface static routes can be redistributed without a default metric. The metric includes: ■ Bandwidth: 0 or greater kbps. ■ Delay (tens of microseconds): 0 or any positive number that is a multiple of 39.1 nanoseconds. ■ Reliability: any number between 0 and 255 (255 means 100 percent reliability). ■ Loading: effective bandwidth as a number between 0 and 255 (255 is 100 percent loading). ■ MTU: maximum transmission unit size of the route in bytes. 0 or any positive integer.
Distance	Internal distance: 90. External distance: 170.
EIGRP log-neighbor changes	Disabled. No adjacency changes logged.
IP authentication key-chain	No authentication provided.
IP authentication mode	No authentication provided.
IP bandwidth-percent	50 percent.
IP hello interval	For low-speed nonbroadcast multiaccess (NBMA) networks: 60 seconds; all other networks: 5 seconds.
IP hold-time	For low-speed NBMA networks: 180 seconds; all other networks: 15 seconds.
IP split-horizon	Enabled.
IP summary address	No summary aggregate addresses are predefined.
Metric weights	tos: 0; k1 and k3: 1; k2, k4, and k5: 0.
Network	None specified.
NSF[1] Awareness	Enabled[2]. Allows Layer 3 switches to continue forwarding packets from a neighboring NSF-capable router during hardware or software changes.
Offset-list	Disabled.
Router EIGRP	Disabled.
Set metric	No metric set in the route map.
Traffic-share	Distributed proportionately to the ratios of the metrics.
Variance	1 (equal-cost load balancing).

[1] NSF = Nonstop Forwarding

[2] EIGRP NSF awareness is enabled for IPv4 on switches running the IP services image.

To create an EIGRP routing process, you must enable EIGRP and associate networks. EIGRP sends updates to the interfaces in the specified networks. If you do not specify an interface network, it is not advertised in any EIGRP update.

Nonstop Forwarding Awareness

The EIGRP NSF Awareness feature is supported for IPv4 in the IP services image. When the neighboring router is NSF-capable, the Layer 3 switch continues to forward packets from the neighboring router during the interval between the primary Route Processor (RP) in a router failing and the backup RP taking over, or while the primary RP is manually reloaded for a nondisruptive software upgrade.

This feature cannot be disabled. For more information on this feature, see the “Configuring Nonstop Forwarding” chapter in the *High Availability Configuration Guide, Cisco IOS Release 15S*.

Configuring Basic EIGRP Parameters

In this procedure, configuring the routing process is required; other steps are optional.

BEFORE YOU BEGIN

Complete the EIGRP network strategy and planning for your network.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**router eigrp** *autonomous-system*	Enable an EIGRP routing process, and enter router configuration mode. The AS number identifies the routes to other EIGRP routers and is used to tag routing information.
3.	**network** *network-number*	Associate networks with an EIGRP routing process. EIGRP sends updates to the interfaces in the specified networks.
4.	**eigrp log-neighbor-changes**	(Optional) Enable logging of EIGRP neighbor changes to monitor routing system stability.
5.	**metric weights** *tos k1 k2 k3 k4 k5*	(Optional) Adjust the EIGRP metric. Although the defaults have been carefully set to provide excellent operation in most networks, you can adjust them. **Caution: Setting metrics is complex and is not recommended without guidance from an experienced network designer.**
6.	**offset list** [*access-list number* | *name*] {**in** | **out**} *offset* [*type number*]	(Optional) Apply an offset list to routing metrics to increase incoming and outgoing metrics to routes learned through EIGRP. You can limit the offset list with an access list or an interface.
7.	**no auto-summary**	(Optional) Disable automatic summarization of subnet routes into network-level routes.
8.	**ip summary-address eigrp** *autonomous-system-number address mask*	(Optional) Configure a summary aggregate.
9.	**end**	Return to privileged EXEC mode.

10.	**show ip protocols**	Verify your entries.
		For NSF awareness, the output shows:
		`*** IP Routing is NSF aware ***`
		`EIGRP NSF enabled`

11.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** forms of these commands to disable the feature or return the setting to the default value.

EXAMPLE

The following example configures EIGRP autonomous system 1 and establishes neighbors through networks 172.16.0.0 and 192.168.0.0:

```
Switch(config)# router eigrp 1
Switch(config-router)# network 172.16.0.0
Switch(config-router)# network 192.168.0.0
```

Configuring EIGRP Interfaces

Other optional EIGRP parameters can be configured on an interface basis.

BEFORE YOU BEGIN

Enable EIGRP as described in the Configuring Basic EIGRP Parameters, page 882.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
3.	**no shutdown**	Enable the interface if necessary. By default, UNIs and ENIs are disabled and NNIs are enabled.
4.	**ip bandwidth-percent eigrp** *percent*	(Optional) Configure the percentage of bandwidth that can be used by EIGRP on an interface. The default is 50 percent.
5.	**ip summary-address eigrp** *autonomous-system-number address mask*	(Optional) Configure a summary aggregate address for a specified interface (not usually necessary if auto-summary is enabled).
6.	**ip hello-interval eigrp** *autonomous-system-number seconds*	(Optional) Change the hello time interval for an EIGRP routing process. The range is 1 to 65535 seconds. The default is 60 seconds for low-speed NBMA networks and 5 seconds for all other networks.
7.	**ip hold-time eigrp** *autonomous-system-number seconds*	(Optional) Change the hold time interval for an EIGRP routing process. The range is 1 to 65535 seconds. The default is 180 seconds for low-speed NBMA networks and 15 seconds for all other networks. **Caution: Do not adjust the hold time without consulting Cisco technical support.**
8.	**no ip split-horizon eigrp** *autonomous-system-number*	(Optional) Disable split horizon to allow route information to be advertised by a router out any interface from which that information originated.
9.	**end**	Return to privileged EXEC mode.
10.	**show ip eigrp interface**	Display which interfaces EIGRP is active on and information about EIGRP relating to those interfaces.
11.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** forms of these commands to disable the feature or return the setting to the default value.

EXAMPLE

The following example allows EIGRP to use up to 75 percent (42 kbps) of a 56-kbps serial link in autonomous system 209:

```
Switch(config)# interface serial 0
Switch(config-if)# bandwidth 56
Switch(config-if)# ip bandwidth-percent eigrp 209 75
```

Configuring EIGRP Route Authentication

EIGRP route authentication provides MD5 authentication of routing updates from the EIGRP routing protocol to prevent the introduction of unauthorized or false routing messages from unapproved sources.

BEFORE YOU BEGIN

Enable EIGRP as described in the Configuring Basic EIGRP Parameters, page 882.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
3.	**no shutdown**	Enable the interface if necessary. By default, UNIs and ENIs are disabled and NNIs are enabled.
4.	**ip authentication mode eigrp** *autonomous-system* **md5**	Enable MD5 authentication in IP EIGRP packets.
5.	**ip authentication key-chain eigrp** *autonomous-system key-chain*	Enable authentication of IP EIGRP packets.
6.	**exit**	Return to global configuration mode.
7.	**key chain** *name-of-chain*	Identify a key chain and enter key-chain configuration mode. Match the name configured in Step 4.
8.	**key** *number*	In key-chain configuration mode, identify the key number.
9.	**key-string** *text*	In key-chain key configuration mode, identify the key string.
10.	**accept-lifetime** *start-time* {**infinite** | *end-time* | **duration** *seconds*}	(Optional) Specify the time period during which the key can be received. The *start-time* and *end-time* syntax can be either *hh*:*mm*:*ss Month date year* or *hh*:*mm*:*ss date Month year*. The default is forever with the default *start-time* and the earliest acceptable date as January 1, 1993. The default *end-time* and **duration** is **infinite**.
11.	**send-lifetime** *start-time* {**infinite** | *end-time* | **duration** *seconds*}	(Optional) Specify the time period during which the key can be sent. The *start-time* and *end-time* syntax can be either *hh*:*mm*:*ss Month date year* or *hh*:*mm*:*ss date Month year*. The default is forever with the default *start-time* and the earliest acceptable date as January 1, 1993. The default *end-time* and **duration** is **infinite**.
12.	**end**	Return to privileged EXEC mode.
13.	**show key chain**	Display authentication key information.
14.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** forms of these commands to disable the feature or to return the setting to the default value.

EXAMPLE

The following example configures EIGRP to apply authentication to address-family autonomous system 1 and identifies a key chain named SITE1:

```
Switch(config)# router eigrp virtual-name
Switch(config-router)# address-family ipv4 autonomous-system 1
Switch(config-router-af)# af-interface ethernet0/0
Switch(config-router-af-interface)# authentication key-chain SITE1
Switch(config-router-af-interface)# authentication mode md5
```

Configuring EIGRP Stub Routing

The EIGRP stub routing feature reduces resource utilization by moving routed traffic closer to the end user. In a network using EIGRP stub routing, the only allowable route for IP traffic to the user is through a switch that is configured with EIGRP stub routing. The switch sends the routed traffic to interfaces that are configured as user interfaces or are connected to other devices.

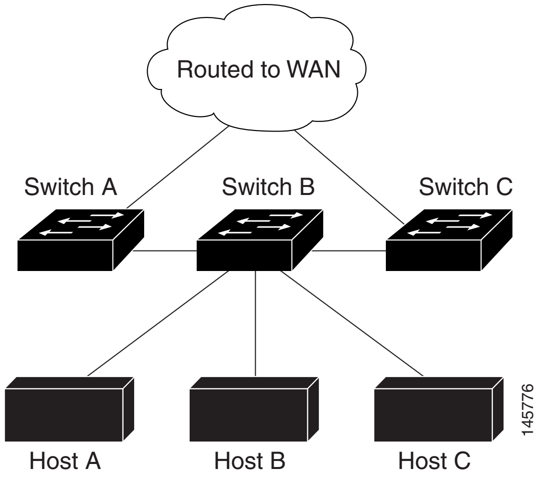
When using EIGRP stub routing, you need to configure the distribution and remote routers to use EIGRP and to configure only the switch as a stub. Only specified routes are propagated from the switch. The switch responds to all queries for summaries, connected routes, and routing updates.

Note: EIGRP stub routing only advertises connected or summary routes from the routing tables to other switches in the network. The switch uses EIGRP stub routing at the access layer to eliminate the need for other types of routing advertisements. If you try to configure multi-VRF-CE and EIGRP stub routing at the same time, the configuration is not allowed.

Any neighbor that receives a packet informing it of the stub status does not query the stub router for any routes, and a router that has a stub peer does not query that peer. The stub router depends on the distribution router to send the proper updates to all peers.

In Figure 102 on page 886, switch B is configured as an EIGRP stub router. Switches A and C are connected to the rest of the WAN. Switch B advertises connected, static, redistribution, and summary routes to switch A and C. Switch B does not advertise any routes learned from switch A (and the reverse).

Figure 102 EIGRP Stub Router Configuration



For more information about EIGRP stub routing, see *IP Routing: EIGRP Configuration Guide, Cisco IOS Release 15M&T*.

BEFORE YOU BEGIN

Complete the EIGRP network strategy and planning for your network.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**router eigrp 1**	Configure a remote or distribution router to run an EIGRP process and enter router configuration mode.
3.	**network** *network-number*	Associate networks with an EIGRP routing process.
4.	**eigrp stub** [**receive-only** | **connected** | **static** | **summary**]	Configure a remote router as an EIGRP stub router. The keywords have these meanings: ■ Enter **receive-only** to set the router as a receive-only neighbor. ■ Enter **connected** to advertise connected routes. ■ Enter **static** to advertise static routes. ■ Enter **summary** to advertise summary routes.
5.	**end**	Return to privileged EXEC mode.
6.	**show ip eigrp neighbor detail**	Verify that a remote router has been configured as a stub router with EIGRP. The last line of the output shows the stub status of the remote or spoke router.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Enter the **show ip eigrp neighbor detail** privileged EXEC command from the distribution router to verify the configuration.

EXAMPLE

In the following example, the **eigrp stub** command is used to configure the router as a stub that advertises connected and summary routes:

```
Switch(config)# router eigrp 1
Switch(config-router)# network 10.0.0.0
Switch(config-router)# eigrp stub
```

Monitoring and Maintaining EIGRP

You can delete neighbors from the neighbor table. You can also display various EIGRP routing statistics.

Command	Purpose
clear ip eigrp neighbors [*if-address* | *interface*]	Delete neighbors from the neighbor table.
show ip eigrp interface [*interface*] [*as number*]	Display information about interfaces configured for EIGRP.
show ip eigrp neighbors [*type-number*]	Display EIGRP discovered neighbors.
show ip eigrp topology [*autonomous-system-number*] | [[*ip-address*] *mask*]]	Display the EIGRP topology table for a given process.
show ip eigrp traffic [*autonomous-system-number*]	Display the number of packets sent and received for all or a specified EIGRP process.

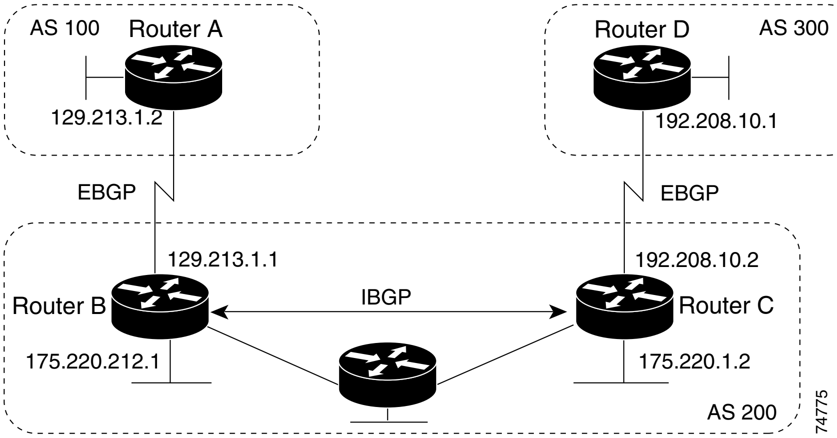
Configuring BGP

The Border Gateway Protocol (BGP) is an exterior gateway protocol used to set up an interdomain routing system for loop-free exchanges of routing information between autonomous systems. Autonomous systems are made up of routers that operate under the same administration and that run Interior Gateway Protocols (IGPs), such as RIP or OSPF, within their boundaries and that interconnect by using an Exterior Gateway Protocol (EGP). BGP Version 4 is the standard EGP for interdomain routing in the Internet.

For details about BGP configuration and commands, see the BGP documents listed in Related Documents, page 966.

Routers that belong to the same autonomous system (AS) and that exchange BGP updates run *internal BGP* (IBGP), and routers that belong to different autonomous systems and that exchange BGP updates run *external BGP* (EBGP). Most configuration commands are the same for configuring EBGP and IBGP. The difference is that the routing updates are exchanged either between autonomous systems (EBGP) or within an AS (IBGP). Figure 103 on page 888 shows a network that is running both EBGP and IBGP.

Figure 103 EBGP, IBGP, and Multiple Autonomous Systems



Before exchanging information with an external AS, BGP ensures that networks within the AS can be reached by defining internal BGP peering among routers within the AS and by redistributing BGP routing information to IGPs that run within the AS, such as IGRP and OSPF.

Routers that run a BGP routing process are often referred to as BGP *speakers*. BGP uses the Transmission Control Protocol (TCP) as its transport protocol (specifically port 179). Two BGP speakers that have a TCP connection to each other for exchanging routing information are known as *peers* or *neighbors*. In Figure 103 on page 888, Routers A and B are BGP peers, as are Routers B and C and Routers C and D. The routing information is a series of AS numbers that describe the full path to the destination network. BGP uses this information to construct a loop-free map of autonomous systems.

The network has these characteristics:

- Routers A and B are running EBGP, and Routers B and C are running IBGP. Note that the EBGP peers are directly connected and that the IBGP peers are not. As long as there is an IGP running that allows the two neighbors to reach one another, IBGP peers do not have to be directly connected.
- All BGP speakers within an AS must establish a peer relationship with each other. That is, the BGP speakers within an AS must be fully meshed logically. BGP4 provides two techniques that reduce the requirement for a logical full mesh: *confederations* and *route reflectors*.
- AS 200 is a *transit AS* for AS 100 and AS 300—that is, AS 200 is used to transfer packets between AS 100 and AS 300.

BGP peers initially exchange their full BGP routing tables and then send only incremental updates. BGP peers also exchange keepalive messages (to ensure that the connection is up) and notification messages (in response to errors or special conditions).

In BGP, each route consists of a network number, a list of autonomous systems that information has passed through (the *autonomous system path*), and a list of other *path attributes*. The primary function of a BGP system is to exchange network reachability information, including information about the list of AS paths, with other BGP systems. This information can be used to determine AS connectivity, to prune routing loops, and to enforce AS-level policy decisions.

A router or switch running Cisco IOS does not select or use an IBGP route unless it has a route available to the next-hop router and it has received synchronization from an IGP (unless IGP synchronization is disabled). When multiple routes are available, BGP bases its path selection on *attribute* values. See Configuring BGP Decision Attributes, page 896 for information about BGP attributes.

BGP Version 4 supports classless interdomain routing (CIDR) so you can reduce the size of your routing tables by creating aggregate routes, resulting in *supernets*. CIDR eliminates the concept of network classes within BGP and supports the advertising of IP prefixes.

This section includes the following topics:

Default BGP Configuration

Feature	Default Setting
Aggregate address	Disabled: None defined.
AS path access list	None defined.
Auto summary	Enabled.
Best path	■ The router considers *as-path* in choosing a route and does not compare similar routes from external BGP peers. ■ Compare router ID: Disabled.
BGP community list	■ Number: None defined. When you permit a value for the community number, the list defaults to an implicit deny for everything else that has not been permitted. ■ Format: Cisco default format (32-bit number).
BGP confederation identifier/peers	■ Identifier: None configured. ■ Peers: None identified.
BGP Fast external fallover	Enabled.
BGP local preference	100. The range is 0 to 4294967295 with the higher value preferred.
BGP network	None specified; no backdoor route advertised.
BGP route dampening	Disabled by default. When enabled: ■ Half-life is 15 minutes. ■ Re-use is 750 (10-second increments). ■ Suppress is 2000 (10-second increments). ■ Max-suppress-time is 4 times half-life; 60 minutes.
BGP router ID	The IP address of a loopback interface if one is configured or the highest IP address configured for a physical interface on the router.
Default information originate (protocol or network redistribution)	Disabled.
Default metric	Built-in, automatic metric translations.
Distance	■ External route administrative distance: 20 (acceptable values are from 1 to 255). ■ Internal route administrative distance: 200 (acceptable values are from 1 to 255). ■ Local route administrative distance: 200 (acceptable values are from 1 to 255).
Distribute list	■ In (filter networks received in updates): Disabled. ■ Out (suppress networks from being advertised in updates): Disabled.
Internal route redistribution	Disabled.
IP prefix list	None defined.

Feature	Default Setting
Multi exit discriminator (MED)	■ Always compare: Disabled. Does not compare MEDs for paths from neighbors in different autonomous systems. ■ Best path compare: Disabled. ■ MED missing as worst path: Disabled. ■ Deterministic MED comparison is disabled.
Neighbor	■ Advertisement interval: 30 seconds for external peers; 5 seconds for internal peers. ■ Change logging: Enabled. ■ Conditional advertisement: Disabled. ■ Default originate: No default route is sent to the neighbor. ■ Description: None. ■ Distribute list: None defined. ■ External BGP multihop: Only directly connected neighbors are allowed. ■ Filter list: None used. ■ Maximum number of prefixes received: No limit.
Neighbor	■ Next hop (router as next hop for BGP neighbor): Disabled. ■ Password: Disabled. ■ Peer group: None defined; no members assigned. ■ Prefix list: None specified. ■ Remote AS (add entry to neighbor BGP table): No peers defined. ■ Private AS number removal: Disabled. ■ Route maps: None applied to a peer. ■ Send community attributes: None sent to neighbors. ■ Shutdown or soft reconfiguration: Not enabled. ■ Timers: keepalive: 60 seconds; holdtime: 180 seconds. ■ Update source: Best local address. ■ Version: BGP Version 4. ■ Weight: Routes learned through BGP peer: 0; routes sourced by the local router: 32768.
NSF[1] Awareness	Disabled[2]. Allows Layer 3 switches to continue forwarding packets from a neighboring NSF-capable router during hardware or software changes.
Route reflector	None configured.

Feature	Default Setting
Synchronization (BGP and IGP)	Enabled.
Table map update	Disabled.
Timers	Keepalive: 60 seconds; holdtime: 180 seconds.

1. NSF = Nonstop Forwarding

2. BGP NSF Awareness can be enabled for IPv4 on switches with the IP services image by enabling Graceful Restart.

Nonstop Forwarding Awareness

The BGP NSF Awareness feature is supported for IPv4 in the IP services image. To enable this feature with BGP routing, you need to enable Graceful Restart. When the neighboring router is NSF-capable, and this feature is enabled, the Layer 3 switch continues to forward packets from the neighboring router during the interval between the primary Route Processor (RP) in a router failing and the backup RP taking over, or while the primary RP is manually reloaded for a nondisruptive software upgrade. For more information, see *IP Routing: BGP Configuration Guide, Cisco IOS Release 15M&T*.

Enabling BGP Routing

To enable BGP routing, you establish a BGP routing process and define the local network. Because BGP must completely recognize the relationships with its neighbors, you must also specify a BGP neighbor.

BGP supports two kinds of neighbors: internal and external. *Internal neighbors* are in the same AS; *external neighbors* are in different autonomous systems. External neighbors are usually adjacent to each other and share a subnet, but internal neighbors can be anywhere in the same AS.

The switch supports the use of private AS numbers, usually assigned by service providers and given to systems whose routes are not advertised to external neighbors. The private AS numbers are from 64512 to 65535. You can configure external neighbors to remove private AS numbers from the AS path by using the **neighbor remove-private-as** router configuration command. Then when an update is passed to an external neighbor, if the AS path includes private AS numbers, these numbers are dropped.

If your AS must pass traffic through it from another AS to a third AS, it is important to be consistent about the routes it advertises. If BGP advertises a route before all routers in the network learn about the route through the IGP, the AS might receive traffic that some routers can not yet route. To prevent this from happening, BGP must wait until the IGP has propagated information across the AS so that BGP is *synchronized* with the IGP. Synchronization is enabled by default. If your AS does not pass traffic from one AS to another AS, or if all routers in your autonomous systems are running BGP, you can disable synchronization, which allows your network to carry fewer routes in the IGP and allows BGP to converge more quickly.

BEFORE YOU BEGIN

You should know your network design and how you want traffic to flow through it before configuring BGP. Gather the network requirements you need, which should include the following:

- Whether you need to run IBGP for internal connectivity
- External connectivity to the service provider network
- Configuration parameters such as neighbor IP addresses and their AS number, and which networks you will advertise through BGP

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip routing**	Enable IP routing (required only if IP routing is disabled).
3.	**router bgp** *autonomous-system*	Enable a BGP routing process, assign it an AS number, and enter router configuration mode. The AS number can be from 1 to 65535, with 64512 to 65535 designated as private autonomous numbers.
4.	**network** *network-number* [**mask** *network-mask*] [**route-map** *route-map-name*]	Configure a network as local to this AS, and enter it in the BGP table.
5.	**neighbor** {*ip-address* | *peer-group-name*} **remote-as** *number*	Add an entry to the BGP neighbor table specifying that the neighbor identified by the IP address belongs to the specified AS. For EBGP, neighbors are usually directly connected, and the IP address is the address of the interface at the other end of the connection. For IBGP, the IP address can be the address of any of the router interfaces.
6.	**neighbor** {*ip-address* | *peer-group-name*} **remove-private-as**	(Optional) Remove private AS numbers from the AS-path in outbound routing updates.
7.	**no synchronization**	(Optional) Disable synchronization between BGP and an IGP.
8.	**no auto-summary**	(Optional) Disable automatic network summarization. By default, when a subnet is redistributed from an IGP into BGP, only the network route is inserted into the BGP table.
9.	**bgp fast-external-fallover**	(Optional) Automatically reset a BGP session when a link between external neighbors goes down. By default, the session is not immediately reset.
10.	**bgp graceful-restart**	(Optional) Enable NSF awareness on switch. By default, NSF awareness is disabled.

	Command	Purpose
11.	**end**	Return to privileged EXEC mode.

12.	**show ip bgp network** *network-number*
	or
	show ip bgp neighbor
		Verify the configuration.
		Verify that NSF awareness (Graceful Restart) is enabled on the neighbor.
		If NSF awareness is enabled on the switch and the neighbor, this message appears:
		`Graceful Restart Capability: advertised and received`
		If NSF awareness is enabled on the switch, but not on the neighbor, this message appears:
		`Graceful Restart Capability: advertised`

13.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no router bgp** *autonomous-system* global configuration command to remove a BGP AS. Use the **no network** *network-number* router configuration command to remove the network from the BGP table. Use the **no neighbor** {*ip-address* | *peer-group-name*} **remote-as** *number* router configuration command to remove a neighbor. Use the **no neighbor** {*ip-address* | *peer-group-name*} **remove-private-as** router configuration command to include private AS numbers in updates to a neighbor. Use the **synchronization** router configuration command to re-enable synchronization.

EXAMPLE

These examples show how to configure BGP on the routers in Figure 103 on page 888.

Router A:

```
Switch(config)# router bgp 100
Switch(config-router)# neighbor 129.213.1.1 remote-as 200
```

Router B:

```
Switch(config)# router bgp 200
Switch(config-router)# neighbor 129.213.1.2 remote-as 100
Switch(config-router)# neighbor 175.220.1.2 remote-as 200
```

Router C:

```
Switch(config)# router bgp 200
Switch(config-router)# neighbor 175.220.212.1 remote-as 200
Switch(config-router)# neighbor 192.208.10.1 remote-as 300
```

Router D:

```
Switch(config)# router bgp 300
Switch(config-router)# neighbor 192.208.10.2 remote-as 200
```

To verify that BGP peers are running, use the **show ip bgp neighbors** privileged EXEC command. This is the output of this command on Router A:

```
Switch# show ip bgp neighbors

BGP neighbor is 129.213.1.1, remote AS 200, external link
 BGP version 4, remote router ID 175.220.212.1
 BGP state = established, table version = 3, up for 0:10:59
 Last read 0:00:29, hold time is 180, keepalive interval is 60 seconds
 Minimum time between advertisement runs is 30 seconds
```

```
Received 2828 messages, 0 notifications, 0 in queue
Sent 2826 messages, 0 notifications, 0 in queue
Connections established 11; dropped 10
```

Anything other than *state = established* means that the peers are not running. The remote router ID is the highest IP address on that router (or the highest loopback interface). Each time the table is updated with new information, the table version number increments. A table version number that continually increments means that a route is flapping, causing continual routing updates.

For exterior protocols, a reference to an IP network from the **network** router configuration command controls only which networks are advertised. This is in contrast to Interior Gateway Protocols (IGPs), such as EIGRP, which also use the **network** command to specify where to send updates.

Managing Routing Policy Changes

Routing policies for a peer include all the configurations that might affect inbound or outbound routing table updates. When you have defined two routers as BGP neighbors, they form a BGP connection and exchange routing information. If you later change a BGP filter, weight, distance, version, or timer, or make a similar configuration change, you must reset the BGP sessions so that the configuration changes take effect.

There are two types of reset: hard reset and soft reset. The switch supports a soft reset without any prior configuration when both BGP peers support the soft route refresh capability, which is advertised in the OPEN message sent when the peers establish a TCP session. A soft reset allows the dynamic exchange of route refresh requests and routing information between BGP routers and the subsequent re-advertisement of the respective outbound routing table.

- When soft reset generates inbound updates from a neighbor, it is called *dynamic inbound soft reset*.
- When soft reset sends a set of updates to a neighbor, it is called *outbound soft reset*.

A soft inbound reset causes the new inbound policy to take effect. A soft outbound reset causes the new local outbound policy to take effect without resetting the BGP session. As a new set of updates is sent during outbound policy reset, a new inbound policy can also take effect.

Table 65 Advantages and Disadvantages of Hard and Soft Resets

Type of Reset	Advantages	Disadvantages
Hard reset	No memory overhead.	The prefixes in the BGP, IP, and FIB tables provided by the neighbor are lost. Not recommended.
Outbound soft reset	No configuration; no storing of routing table updates.	Does not reset inbound routing table updates.
Dynamic inbound soft reset	Does not clear the BGP session and cache. Does not require storing of routing table updates and has no memory overhead.	Both BGP routers must support the route refresh capability.

BEFORE YOU BEGIN

Enable BGP routing as described in the Enabling BGP Routing, page 892.

DETAILED STEPS

	Command	Purpose
1.	**show ip bgp neighbors**	Display whether a neighbor supports the route refresh capability. When supported, this message appears for the router: *Received route refresh capability from peer.*
2.	**clear ip bgp** {***** | *address* | *peer-group-name*}	Reset the routing table on the specified connection. ■ Enter an asterisk (*) to specify that all connections be reset. ■ Enter an IP *address* to specify the connection to be reset. ■ Enter a peer group name to reset the peer group.
3.	**clear ip bgp** {***** | *address* | *peer-group-name*} **soft out**	(Optional) Perform an outbound soft reset to reset the inbound routing table on the specified connection. Use this command if route refresh is supported. ■ Enter an asterisk (*) to specify that all connections be reset. ■ Enter an IP *address* to specify the connection to be reset. ■ Enter a peer group name to reset the peer group.
4.	**show ip bgp** **show ip bgp neighbors**	Verify the reset by checking information about the routing table and about BGP neighbors.

EXAMPLE

In the following example, an outbound soft reset is initiated for sessions with all routers in the autonomous system numbered 35700:

```
Switch# clear ip bgp 35700 soft out
```

Configuring BGP Decision Attributes

When a BGP speaker receives updates from multiple autonomous systems that describe different paths to the same destination, it must choose the single best path for reaching that destination. The decision is based on the value of attributes that the update contains and other BGP-configurable factors. The selected path is entered into the BGP routing table and propagated to its neighbors.

When a BGP peer learns two EBGP paths for a prefix from a neighboring AS, it chooses the best path and inserts that path in the IP routing table. If BGP multipath support is enabled and the EBGP paths are learned from the same neighboring autonomous systems, multiple paths are installed in the IP routing table. Then, during packet switching, per-packet or per-destination load balancing is performed among the multiple paths. The **maximum-paths** router configuration command controls the number of paths allowed.

These factors summarize the order in which BGP evaluates the attributes for choosing the best path:

1. If the path specifies a next hop that is inaccessible, drop the update. The BGP next-hop attribute, automatically determined by the software, is the IP address of the next hop that is going to be used to reach a destination. For EBGP, this is usually the IP address of the neighbor specified by the **neighbor remote-as** router configuration command. You can disable next-hop processing by using route maps or the **neighbor next-hop-self** router configuration command.

2. Prefer the path with the largest weight (a Cisco proprietary parameter). The weight attribute is local to the router and not propagated in routing updates. By default, the weight attribute is 32768 for paths that the router originates and zero for other paths. You can use access lists, route maps, or the **neighbor weight** router configuration command to set weights.

3. Prefer the route with the highest local preference. Local preference is part of the routing update and exchanged among routers in the same AS. The default value of the local preference attribute is 100. You can set local preference by using the **bgp default local-preference** router configuration command or by using a route map.
4. Prefer the route that was originated by BGP running on the local router.
5. Prefer the route with the shortest AS path.
6. Prefer the route with the lowest origin type. An interior route or IGP is lower than a route learned by EGP, and an EGP-learned route is lower than one of unknown origin or learned in another way.
7. Prefer the route with the lowest multi-exit discriminator (MED) metric attribute if the neighboring AS is the same for all routes considered. You can configure the MED by using route maps or by using the **default-metric** router configuration command. When an update is sent to an IBGP peer, the MED is included.
8. Prefer the external (EBGP) path over the internal (IBGP) path.
9. Prefer the route that can be reached through the closest IGP neighbor (the lowest IGP metric). This means that the router will prefer the shortest internal path within the AS to reach the destination (the shortest path to the BGP next-hop).
10. If these conditions are all true, insert the route for this path into the IP routing table:
    - Both the best route and this route are external.
    - Both the best route and this route are from the same neighboring autonomous system.
    - Maximum-paths is enabled.
11. If multipath is not enabled, prefer the route with the lowest IP address value for the BGP router ID. The router ID is usually the highest IP address on the router or the loopback (virtual) address, but might be implementation-specific.

BEFORE YOU BEGIN

Enable BGP routing as described in the Enabling BGP Routing, page 892.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**router bgp** *autonomous-system*	Enable a BGP routing process, assign it an AS number, and enter router configuration mode.
3.	**bgp best-path as-path ignore**	(Optional) Configure the router to ignore AS path length in selecting a route.
4.	**neighbor** {*ip-address* | *peer-group-name*} **next-hop-self**	(Optional) Disable next-hop processing on BGP updates to a neighbor by entering a specific IP address to be used instead of the next-hop address.
5.	**neighbor** {*ip-address* | *peer-group-name*} **weight** *weight*	(Optional) Assign a weight to a neighbor connection. Acceptable values are from 0 to 65535; the largest weight is the preferred route. Routes learned through another BGP peer have a default weight of 0; routes sourced by the local router have a default weight of 32768.
6.	**default-metric** *number*	(Optional) Set a MED metric to set preferred paths to external neighbors. All routes without a MED will also be set to this value. The range is 1 to 4294967295. The lowest value is the most desirable.
7.	**bgp bestpath med missing-as-worst**	(Optional) Configure the switch to consider a missing MED as having a value of infinity, making the path without a MED value the least desirable path.
8.	**bgp always-compare med**	(Optional) Configure the switch to compare MEDs for paths from neighbors in different autonomous systems. By default, MED comparison is only done among paths in the same AS.
9.	**bgp bestpath med confed**	(Optional) Configure the switch to consider the MED in choosing a path from among those advertised by different subautonomous systems within a confederation.
10.	**bgp deterministic med**	(Optional) Configure the switch to consider the MED variable when choosing among routes advertised by different peers in the same AS.
11.	**bgp default local-preference** *value*	(Optional) Change the default local preference value. The range is 0 to 4294967295; the default value is 100. The highest local preference value is preferred.
12.	**maximum-paths** *number*	(Optional) Configure the number of paths to be added to the IP routing table. The default is to only enter the best path in the routing table. The range is from 1 to 8. Having multiple paths allows load balancing among the paths.
13.	**end**	Return to privileged EXEC mode.
14.	**show ip bgp** **show ip bgp neighbors**	Verify the reset by checking information about the routing table and about BGP neighbors.
15.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no** form of each command to return to the default state.

EXAMPLE

The following example forces all updates destined for 10.108.1.1 to advertise this router as the next hop:

```
Switch(config)# router bgp 109
Switch(config-router)# neighbor 10.108.1.1 next-hop-self
```

In the following example, the local BGP routing process is configured to compare the MED from alternative paths, regardless of the autonomous system from which the paths are received:

```
Switch(config)# router bgp 500000
Switch(config-router)# bgp always-compare-med
```

Configuring BGP Filtering with Route Maps

Within BGP, you can use route maps to control and to modify routing information and to define the conditions by which routes are redistributed between routing domains. See Using Route Maps to Redistribute Routing Information, page 953 for more information about route maps. Each route map has a name that identifies the route map (*map tag*) and an optional sequence number.

BEFORE YOU BEGIN

Enable BGP routing as described in the Enabling BGP Routing, page 892.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**route-map** *map-tag* [[**permit** | **deny**] | *sequence-number*]]	Create a route map, and enter route-map configuration mode.
3.	**set ip next-hop** *ip-address* [*...ip-address*] [**peer-address**]	(Optional) Set a route map to disable next-hop processing. ■ In an inbound route map, set the next hop of matching routes to be the neighbor peering address, overriding third-party next hops. ■ In an outbound route map of a BGP peer, set the next hop to the peering address of the local router, disabling the next-hop calculation.
4.	**end**	Return to privileged EXEC mode.
5.	**show route-map** [*map-name*]	Display all route maps configured or only the one specified to verify configuration.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no route-map** *map-tag* command to delete the route map. Use the **no set ip next-hop** *ip-address* command to re-enable next-hop processing.

EXAMPLE

In the following example, the inbound route map named rmap sets the next hop:

```
Switch(config)# route-map rmap permit 10
Switch(config-route-map)# set ip next-hop 10.2.0.1
```

Configuring BGP Filtering by Neighbor

You can filter BGP advertisements by using AS-path filters, such as the **as-path access-list** global configuration command and the **neighbor filter-list** router configuration command. You can also use access lists with the **neighbor distribute-list** router configuration command. Distribute-list filters are applied to network numbers. See Controlling Advertising and Processing in Routing Updates, page 962 for information about the **distribute-list** command.

You can use route maps on a per-neighbor basis to filter updates and to modify various attributes. A route map can be applied to either inbound or outbound updates. Only the routes that pass the route map are sent or accepted in updates. On both inbound and outbound updates, matching is supported based on AS path, community, and network numbers. Autonomous-system path matching requires the **match as-path access-lis**t route-map command, community-based matching requires the **match community-list** route-map command, and network-based matching requires the **ip access-list** global configuration command.

BEFORE YOU BEGIN

Enable BGP routing as described in the Enabling BGP Routing, page 892.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**router bgp** *autonomous-system*	Enable a BGP routing process, assign it an AS number, and enter router configuration mode.
3.	**neighbor** {*ip-address* | *peer-group name*} **distribute-list** {*access-list-number* | *name*} {**in** | **out**}	(Optional) Filter BGP routing updates to or from neighbors as specified in an access list. **Note:** You can also use the **neighbor prefix-list** router configuration command to filter updates, but you cannot use both commands to configure the same BGP peer.
4.	**neighbor** {*ip-address* | *peer-group name*} **route-map** *map-tag* {**in** | **out**}	(Optional) Apply a route map to filter an incoming or outgoing route.
5.	**end**	Return to privileged EXEC mode.
6.	**show ip bgp neighbors**	Verify the configuration.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no neighbor distribute-list** command to remove the access list from the neighbor. Use the **no neighbor route-map** *map-tag* router configuration command to remove the route map from the neighbor.

EXAMPLE

The following router configuration mode example applies list 39 to incoming advertisements from neighbor172.16.4.1. List 39 permits the advertisement of network 10.109.0.0.

```
Switch(config)# router bgp 109
Switch(config-router)# network 10.108.0.0
Switch(config-router)# neighbor 172.16.4.1 distribute-list 39 in
```

Configuring BGP Filtering By Access Lists

Another method of filtering is to specify an access list filter on both incoming and outbound updates, based on the BGP autonomous system paths. Each filter is an access list based on regular expressions. (See Using Regular Expressions in BGP for more information on forming regular expressions.) To use this method, define an autonomous system path access list, and apply it to updates to and from particular neighbors.

BEFORE YOU BEGIN

Enable BGP routing as described in the Enabling BGP Routing, page 892.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip as-path access-list** *access-list-number* {**permit** | **deny**} *as-regular-expressions*	Define a BGP-related access list.
3.	**router bgp** *autonomous-system*	Enter BGP router configuration mode.
4.	**neighbor** {*ip-address* | *peer-group name*} **filter-list** {*access-list-number* | *name*} {**in** | **out** | **weight** *weight*}	Establish a BGP filter based on an access list.
5.	**end**	Return to privileged EXEC mode.
6.	**show ip bgp neighbors** [**paths** regular-expression]	Verify the configuration.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

In the following example, an autonomous system path access list (number 500) is defined to configure the router to not advertise any path through or from autonomous system 65535 to the 10.20.2.2 neighbor:

```
Switch(config)# ip as-path access-list 500 deny _65535_
Switch(config)# ip as-path access-list 500 deny ^65535$
Switch(config)# router bgp 50000
Switch(config-router)# neighbor 192.168.1.1 remote-as 65535
Switch(config-router)# neighbor 10.20.2.2 remote-as 40000
Switch(config-router)# neighbor 10.20.2.2 filter-list 500 out
Switch(config-router)# end
```

Configuring Prefix Lists for BGP Filtering

You can use prefix lists as an alternative to access lists in many BGP route filtering commands, including the **neighbor distribute-list** router configuration command. Filtering by a prefix list involves matching the prefixes of routes with those listed in the prefix list, as when matching access lists. When there is a match, the route is used. Whether a prefix is permitted or denied is based upon these rules:

- An empty prefix list permits all prefixes.
- An implicit deny is assumed if a given prefix does not match any entries in a prefix list.
- When multiple entries of a prefix list match a given prefix, the sequence number of a prefix list entry identifies the entry with the lowest sequence number.

By default, sequence numbers are generated automatically and incremented in units of five. If you disable the automatic generation of sequence numbers, you must specify the sequence number for each entry. You can specify sequence values in any increment. If you specify increments of one, you cannot insert additional entries into the list; if you choose very large increments, you might run out of values.

You do not need to specify a sequence number when removing a configuration entry. **Show** commands include the sequence numbers in their output.

Before using a prefix list in a command, you must set up the prefix list.

BEFORE YOU BEGIN

Enable BGP routing as described in the Enabling BGP Routing, page 892.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip prefix-list** *list-name* [**seq** *seq-value*] **deny** | **permit** *network/len* [**ge** *ge-value*] [**le** *le-value*]	Create a prefix list with an optional sequence number to **deny** or **permit** access for matching conditions. You must enter at least one **permit** or **deny** clause. ■ *network/len* is the network number and length (in bits) of the network mask. ■ (Optional) **ge** and **le** values specify the range of the prefix length to be matched. The specified *ge-value* and *le-value* must satisfy this condition: *len* < *ge-value* < *le-value* < 32
3.	**ip prefix-list** *list-name* **seq** *seq-value* **deny** | **permit** *network/len* [**ge** *ge-value*] [**le** *le-value*]	(Optional) Add an entry to a prefix list, and assign a sequence number to the entry.
4.	**end**	Return to privileged EXEC mode.
5.	**show ip prefix list** [**detail** | **summary**] *name* [*network/len*] [**seq** *seq-num*] [**longer**] [**first-match**]	Verify the configuration by displaying information about a prefix list or prefix list entries.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete a prefix list and all of its entries, use the **no ip prefix-list** *list-name* global configuration command. To delete an entry from a prefix list, use the **no ip prefix-list seq** *seq-value* global configuration command. To disable automatic generation of sequence numbers, use the **no ip prefix-list sequence number** command; to reenable automatic generation, use the **ip prefix-list sequence number** command. To clear the hit-count table of prefix list entries, use the **clear ip prefix-list** privileged EXEC command.

EXAMPLE

In the following example, a prefix list is configured to deny the default route 0.0.0.0/0:

```
Switch(config)# ip prefix-list RED deny 0.0.0.0/0
```

In the following example, a prefix list is configured to permit traffic from the 172.16.1.0/24 subnet:

```
Switch(config)# ip prefix-list BLUE permit 172.16.1.0/24
```

In the following example, a prefix list is configured to permit routes from the 10.0.0.0/8 network that have a mask length that is less than or equal to 24 bits:

```
Switch(config)# ip prefix-list YELLOW permit 10.0.0.0/8 le 24
```

In the following example, a prefix list is configured to deny routes from the 10.0.0.0/8 network that have a mask length that is greater than or equal to 25 bits:

```
Switch(config)# ip prefix-list PINK deny 10.0.0.0/8 ge 25
```

In the following example, a prefix list is configured to permit routes from any network that have a mask length from 8 to 24 bits:

```
Switch(config)# ip prefix-list GREEN permit 0.0.0.0/0 ge 8 le 24
```

In the following example, a prefix list is configured to deny any route with any mask length from the 10.0.0.0/8 network:

```
Switch(config)# ip prefix-list ORANGE deny 10.0.0.0/8 le 32
```

Configuring BGP Community Filtering

One way that BGP controls the distribution of routing information based on the value of the COMMUNITIES attribute. A *community* is a group of destinations that share some common attribute. Each destination can belong to multiple communities. AS administrators can define to which communities a destination belongs. By default, all destinations belong to the general Internet community. The community is identified by the COMMUNITIES attribute, an optional, transitive, global attribute in the numerical range from 1 to 4294967200. These are some predefined, well-known communities:

- **internet**—Advertise this route to the Internet community. All routers belong to it.
- **no-export**—Do not advertise this route to EBGP peers.
- **no-advertise**—Do not advertise this route to any peer (internal or external).
- **local-as**—Do not advertise this route to peers outside the local autonomous system.

Based on the community, you can control which routing information to accept, prefer, or distribute to other neighbors. A BGP speaker can set, append, or modify the community of a route when learning, advertising, or redistributing routes. When routes are aggregated, the resulting aggregate has a COMMUNITIES attribute that contains all communities from all the initial routes.

You can use community lists to create groups of communities to use in a match clause of a route map. As with an access list, a series of community lists can be created. Statements are checked until a match is found. As soon as one statement is satisfied, the test is concluded.

To set the COMMUNITIES attribute and match clauses based on communities, see the **match community-list** and **set community** route-map configuration commands in the Using Route Maps to Redistribute Routing Information, page 953.

By default, no COMMUNITIES attribute is sent to a neighbor. You can specify that the COMMUNITIES attribute be sent to the neighbor at an IP address by using the **neighbor send-community** router configuration command.

BEFORE YOU BEGIN

Enable BGP routing as described in the Enabling BGP Routing, page 892.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip community-list** *community-list-number* {**permit** | **deny**} *community-number*	Create a community list, and assign it a number. ■ The *community-list-number* is an integer from 1 to 99 that identifies one or more permit or deny groups of communities. ■ The *community-number* is the number configured by a **set community** route-map configuration command.
3.	**router bgp** *autonomous-system*	Enter BGP router configuration mode.
4.	**neighbor** {*ip-address* | *peer-group name*} **send-community**	Specify that the COMMUNITIES attribute be sent to the neighbor at this IP address.
5.	**set comm-list** *list-num* **delete**	(Optional) Remove communities from the community attribute of an inbound or outbound update that match a standard or extended community list specified by a route map.
6.	**exit**	Return to global configuration mode.
7.	**ip bgp-community new-format**	(Optional) Display and parse BGP communities in the format AA:NN. A BGP community is displayed in a two-part format 2 bytes long. The Cisco default community format is in the format NNAA. In the most recent RFC for BGP, a community takes the form AA:NN, where the first part is the AS number and the second part is a 2-byte number.
8.	**end**	Return to privileged EXEC mode.
9.	**show ip bgp community**	Verify the configuration.
10.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

In the following example, a standard community list is configured that permits routes from network 10 in autonomous system 50000:

```
Router(config)# ip community-list 1 permit 50000:10
```

In the following router configuration mode example, the router belongs to autonomous system 109 and is configured to send the communities attribute to its neighbor at IP address 172.16.70.23:

```
Switch(config)# router bgp 109
Switch(config-router)# neighbor 172.16.70.23 send-community
```

In the following example, a router that uses the 32-bit number community format is upgraded to use the AA:NN format:

```
Switch(config)# ip bgp-community new-format
```

The following sample output shows how BGP community numbers are displayed when the **ip bgp-community new-format** command is enabled:

```
Switch# show ip bgp 10.0.0.0
BGP routing table entry for 10.0.0.0/8, version 4
Paths: (2 available, best #2, table Default-IP-Routing-Table)
  Advertised to non peer-group peers:
  10.0.33.35
```

```
35
  10.0.33.35 from 10.0.33.35 (192.168.3.3)
    Origin incomplete, metric 10, localpref 100, valid, external
    Community: 1:1
Local
  0.0.0.0 from 0.0.0.0 (10.0.33.34)
    Origin incomplete, metric 0, localpref 100, weight 32768, valid, sourced, best
```

Configuring BGP Neighbors and Peer Groups

Often many BGP neighbors are configured with the same update policies (that is, the same outbound route maps, distribute lists, filter lists, update source, and so on). Neighbors with the same update policies can be grouped into peer groups to simplify configuration and to make updating more efficient. When you have configured many peers, we recommend this approach.

To configure a BGP peer group, you create the peer group, assign options to the peer group, and add neighbors as peer group members. You configure the peer group by using the **neighbor** router configuration commands. By default, peer group members inherit all the configuration options of the peer group, including the remote-as (if configured), version, update-source, out-route-map, out-filter-list, out-dist-list, minimum-advertisement-interval, and next-hop-self. All peer group members also inherit changes made to the peer group. Members can also be configured to override the options that do not affect outbound updates.

To assign configuration options to an individual neighbor, specify any of these router configuration commands by using the neighbor IP address. To assign the options to a peer group, specify any of the commands by using the peer group name. You can disable a BGP peer or peer group without removing all the configuration information by using the **neighbor shutdown** router configuration command.

BEFORE YOU BEGIN

Enable BGP routing as described in the Enabling BGP Routing, page 892.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**router bgp** *autonomous-system*	Enter BGP router configuration mode.
3.	**neighbor** *peer-group-name* **peer-group**	Create a BGP peer group.
4.	**neighbor** *ip-address* **peer-group** *peer-group-name*	Make a BGP neighbor a member of the peer group.
5.	**neighbor** {*ip-address* | *peer-group-name*} **remote-as** *number*	Specify a BGP neighbor. If a peer group is not configured with a **remote-as** *number*, use this command to create peer groups containing EBGP neighbors. The range is 1 to 65535.
6.	**neighbor** {*ip-address* | *peer-group-name*} **description** *text*	(Optional) Associate a description with a neighbor.
7.	**neighbor** {*ip-address* | *peer-group-name*} **default-originate** [**route-map** *map-name*]	(Optional) Allow a BGP speaker (the local router) to send the default route 0.0.0.0 to a neighbor for use as a default route.
8.	**neighbor** {*ip-address* | *peer-group-name*} **send-community**	(Optional) Specify that the COMMUNITIES attribute be sent to the neighbor at this IP address.
9.	**neighbor** {*ip-address* | *peer-group-name*} **update-source** *interface*	(Optional) Allow internal BGP sessions to use any operational interface for TCP connections.
10.	**neighbor** {*ip-address* | *peer-group-name*} **ebgp-multihop**	(Optional) Allow BGP sessions, even when the neighbor is not on a directly connected segment. The multihop session is not established if the only route to the multihop peer's address is the default route (0.0.0.0).
11.	**neighbor** {*ip-address* | *peer-group-name*} **local-as** *number*	(Optional) Specify an AS number to use as the local AS. The range is 1 to 65535.
12.	**neighbor** {*ip-address* | *peer-group-name*} **advertisement-interval** *seconds*	(Optional) Set the minimum interval between sending BGP routing updates.
13.	**neighbor** {*ip-address* | *peer-group-name*} **maximum-prefix** *maximum* [*threshold*]	(Optional) Control how many prefixes can be received from a neighbor. The range is 1 to 4294967295. The *threshold* (optional) is the percentage of maximum at which a warning message is generated. The default is 75 percent.
14.	**neighbor** {*ip-address* | *peer-group-name*} **next-hop-self**	(Optional) Disable next-hop processing on the BGP updates to a neighbor.
15.	**neighbor** {*ip-address* | *peer-group-name*} **password** *string*	(Optional) Set MD5 authentication on a TCP connection to a BGP peer. The same password must be configured on both BGP peers, or the connection between them is not made.
16.	**neighbor** {*ip-address* | *peer-group-name*} **route-map** *map-name* {**in** | **out**}	(Optional) Apply a route map to incoming or outgoing routes.
17.	**neighbor** {*ip-address* | *peer-group-name*} **send-community**	(Optional) Specify that the COMMUNITIES attribute be sent to the neighbor at this IP address.

	Command	Purpose
18.	**neighbor** {*ip-address* | *peer-group-name*} **timers** *keepalive holdtime*	(Optional) Set timers for the neighbor or peer group. ■ The *keepalive* interval is the time within which keepalive messages are sent to peers. The range is 1 to 4294967295 seconds; the default is 60. ■ The *holdtime* is the interval after which a peer is declared inactive after not receiving a keepalive message from it. The range is 1 to 4294967295 seconds; the default is 180.
19.	**neighbor** {*ip-address* | *peer-group-name*} **weight** *weight*	(Optional) Specify a weight for all routes from a neighbor.
20.	**neighbor** {*ip-address* | *peer-group-name*} **distribute-list** {*access-list-number* | *name*} {**in** | **out**}	(Optional) Filter BGP routing updates to or from neighbors, as specified in an access list.
21.	**neighbor** {*ip-address* | *peer-group-name*} **filter-list** *access-list-number* {**in** | **out** | **weight** *weight*}	(Optional) Establish a BGP filter.
22.	**neighbor** {*ip-address* | *peer-group-name*} **version** *value*	(Optional) Specify the BGP version to use when communicating with a neighbor.
23.	**neighbor** {*ip-address* | *peer-group-name*} **soft-reconfiguration inbound**	(Optional) Configure the software to start storing received updates.
24.	**end**	Return to privileged EXEC mode.
25.	**show ip bgp neighbors**	Verify the configuration.
26.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable an existing BGP neighbor or neighbor peer group, use the **neighbor shutdown** router configuration command. To enable a previously existing neighbor or neighbor peer group that had been disabled, use the **no neighbor shutdown** router configuration command.

EXAMPLE

The following example configures a peer group and sets the minimum time between sending BGP routing updates to 10 seconds for the peer group:

```
Switch(config)# router bgp 45000
Switch(config-router)# neighbor mygroup peer-group
Switch(config-router)# neighbor 192.168.1.2 remote-as 40000
Switch(config-router)# neighbor 192.168.3.2 remote-as 50000
Switch(config-router)# neighbor 192.168.1.2 peer-group mygroup
Switch(config-router)# neighbor 192.168.3.2 peer-group mygroup
Switch(config-router)# neighbor mygroup advertisement-interval 10
```

Configuring Aggregate Addresses

Classless interdomain routing (CIDR) enables you to create aggregate routes (or *supernets*) to minimize the size of routing tables. You can configure aggregate routes in BGP either by redistributing an aggregate route into BGP or by creating an aggregate entry in the BGP routing table. An aggregate address is added to the BGP table when there is at least one more specific entry in the BGP table.

BEFORE YOU BEGIN

Enable BGP routing as described in the Enabling BGP Routing, page 892.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**router bgp** *autonomous-system*	Enter BGP router configuration mode.
3.	**aggregate-address** *address mask*	Create an aggregate entry in the BGP routing table. The aggregate route is advertised as coming from the AS, and the atomic aggregate attribute is set to indicate that information might be missing.
4.	**aggregate-address** *address mask* **as-set**	(Optional) Generate AS set path information. This command creates an aggregate entry following the same rules as the previous command, but the advertised path will be an AS_SET consisting of all elements contained in all paths. Do not use this keyword when aggregating many paths because this route must be continually withdrawn and updated.
5.	**aggregate-address** *address-mask* **summary-only**	(Optional) Advertise summary addresses only.
6.	**aggregate-address** *address mask* **suppress-map** *map-name*	(Optional) Suppress selected, more specific routes.
7.	**aggregate-address** *address mask* **advertise-map** *map-name*	(Optional) Generate an aggregate based on conditions specified by the route map.
8.	**aggregate-address** *address mask* **attribute-map** *map-name*	(Optional) Generate an aggregate with attributes specified in the route map.
9.	**end**	Return to privileged EXEC mode.
10.	**show ip bgp neighbors** [**advertised-routes**]	Verify the configuration.
11.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete an aggregate entry, use the **no aggregate-address** *address mask* router configuration command. To return options to the default values, use the command with keywords.

EXAMPLE

In the following example, an aggregate BGP address is created in router configuration mode. The path advertised for this route will be an AS_SET consisting of all elements contained in all paths that are being summarized.

```
Switch(config)# router bgp 50000
Switch(config-router)# aggregate-address 10.0.0.0 255.0.0.0 as-set
```

In the following example, a route map called MAP-ONE is created to match on an AS-path access list. The path advertised for this route will be an AS_SET consisting of elements contained in paths that are matched in the route map.

```
Switch(config)# ip as-path access-list 1 deny ^1234_
Switch(config)# ip as-path access-list 1 permit .*
Switch(config)# !
Switch(config)# route-map MAP-ONE
Switch(config-route-map)# match ip as-path 1
Switch(config-route-map)# exit
Switch(config)# router bgp 50000
Switch(config-router)# address-family ipv4
Switch(config-router-af)# aggregate-address 10.0.0.0 255.0.0.0 as-set advertise-map MAP-ONE
Switch(config-router-af)# end
```

Configuring Routing Domain Confederations

One way to reduce the IBGP mesh is to divide an autonomous system into multiple subautonomous systems and to group them into a single confederation that appears as a single autonomous system. Each autonomous system is fully meshed within itself and has a few connections to other autonomous systems in the same confederation. Even though the peers in different autonomous systems have EBGP sessions, they exchange routing information as if they were IBGP peers. Specifically, the next hop, MED, and local preference information is preserved. You can then use a single IGP for all of the autonomous systems.

To configure a BGP confederation, you must specify a confederation identifier that acts as the autonomous system number for the group of autonomous systems.

BEFORE YOU BEGIN

Enable BGP routing as described in the Enabling BGP Routing, page 892.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**router bgp** *autonomous-system*	Enter BGP router configuration mode.
3.	**bgp confederation identifier** *autonomous-system*	Configure a BGP confederation identifier.
4.	**bgp confederation peers** *autonomous-system* [*autonomous-system* ...]	Specify the autonomous systems that belong to the confederation and that will be treated as special EBGP peers.
5.	**end**	Return to privileged EXEC mode.
6.	**show ip bgp neighbor** **show ip bgp network**	Verify the configuration.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

In the following example, the routing domain is divided into autonomous systems 50001, 50002, 50003, 50004, 50005, and 50006 and is identified by the confederation identifier 50007. Neighbor 10.2.3.4 is a peer inside of the routing domain confederation. Neighbor 10.4.5.6 is a peer outside of the routing domain confederation. To external peers and routing domains, the confederation appears as a single autonomous system with the number 50007.

```
router bgp 50000
 bgp confederation identifier 50007
 bgp confederation peers 50001 50002 50003 50004 50005 50006
 neighbor 10.2.3.4 remote-as 50001
 neighbor 10.4.5.6 remote-as 40000
 end
```

Configuring BGP Route Reflectors

BGP requires that all of the IBGP speakers be fully meshed. When a router receives a route from an external neighbor, it must advertise it to all internal neighbors. To prevent a routing information loop, all IBPG speakers must be connected. The internal neighbors do not send routes learned from internal neighbors to other internal neighbors.

With route reflectors, all IBGP speakers need not be fully meshed because another method is used to pass learned routes to neighbors. When you configure an internal BGP peer to be a *route reflector*, it is responsible for passing IBGP learned routes to a set of IBGP neighbors. The internal peers of the route reflector are divided into two groups: *client peers* and *nonclient peers (*all the other routers in the autonomous system). A route reflector reflects routes between these two groups. The route reflector and its client peers form a *cluster*. The nonclient peers must be fully meshed with each other, but the client peers need not be fully meshed. The clients in the cluster do not communicate with IBGP speakers outside their cluster.

When the route reflector receives an advertised route, it takes one of these actions, depending on the neighbor:

- A route from an external BGP speaker is advertised to all clients and nonclient peers.
- A route from a nonclient peer is advertised to all clients.
- A route from a client is advertised to all clients and nonclient peers. Hence, the clients need not be fully meshed.

Usually a cluster of clients have a single route reflector, and the cluster is identified by the route reflector router ID. To increase redundancy and to avoid a single point of failure, a cluster might have more than one route reflector. In this case, all route reflectors in the cluster must be configured with the same 4-byte cluster ID so that a route reflector can recognize updates from route reflectors in the same cluster. All the route reflectors serving a cluster should be fully meshed and should have identical sets of client and nonclient peers.

BEFORE YOU BEGIN

Enable BGP routing as described in the Enabling BGP Routing, page 892.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**router bgp** *autonomous-system*	Enter BGP router configuration mode.
3.	**neighbor** *ip-address* | *peer-group-name* **route-reflector-client**	Configure the local router as a BGP route reflector and the specified neighbor as a client.
4.	**bgp cluster-id** *cluster-id*	(Optional) Configure the cluster ID if the cluster has more than one route reflector.
5.	**no bgp client-to-client reflection**	(Optional) Disable client-to-client route reflection. By default, the routes from a route reflector client are reflected to other clients. However, if the clients are fully meshed, the route reflector does not need to reflect routes to clients.
6.	**end**	Return to privileged EXEC mode.
7.	**show ip bgp**	Verify the configuration. Display the originator ID and the cluster-list attributes.
8.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

In the following router configuration mode example, the local router is a route reflector. It passes learned IBGP routes to the neighbor at 172.16.70.24.

```
router bgp 5
 neighbor 172.16.70.24 route-reflector-client
```

Configuring Route Dampening

Route flap dampening minimizes the propagation of flapping routes across an internetwork. A route is considered to be flapping when it is repeatedly available, then unavailable, then available, then unavailable, and so on. When route dampening is enabled, a numeric *penalty* value is assigned to a route when it flaps. When a route's accumulated penalties reach a configurable limit, BGP suppresses advertisements of the route, even if the route is running. The *reuse limit* is a configurable value that is compared with the penalty. If the penalty is less than the reuse limit, a suppressed route that is up is advertised again.

Dampening is not applied to routes that are learned by IBGP. This policy prevents the IBGP peers from having a higher penalty for routes external to the AS.

BEFORE YOU BEGIN

Enable BGP routing as described in the Enabling BGP Routing, page 892.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**router bgp** *autonomous-system*	Enter BGP router configuration mode.
3.	**bgp dampening**	Enable BGP route dampening.
4.	**bgp dampening** *half-life reuse suppress max-suppress* [**route-map** *map*]	(Optional) Change the default values of route dampening factors.
5.	**end**	Return to privileged EXEC mode.
6.	**show ip bgp flap-statistics** [{**regexp** *regexp*} | {**filter-list** *list*} | {*address mask* [**longer-prefix**]}]	(Optional) Monitor the flaps of all paths that are flapping. The statistics are deleted when the route is not suppressed and is stable.
7.	**show ip bgp dampened-paths**	(Optional) Display the dampened routes, including the time remaining before they are suppressed.
8.	**clear ip bgp flap-statistics** [{**regexp** *regexp*} | {**filter-list** *list*} | {*address mask* [**longer-prefix**]}	(Optional) Clear BGP flap statistics to make it less likely that a route will be dampened.
9.	**clear ip bgp dampening**	(Optional) Clear route dampening information, and unsuppress the suppressed routes.
10.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable flap dampening, use the **no bgp dampening** router configuration command without keywords. To set dampening factors back to the default values, use the **no bgp dampening** router configuration command with values.

EXAMPLE

In the following example, BGP dampening is applied to prefixes filtered through the route-map named BLUE:

```
Switch(config)# ip prefix-list RED permit 10.0.0.0/8
Switch(config)# !
Switch(config)# route-map BLUE

Switch(config-route-map)# match ip address ip prefix-list RED
Switch(config-route-map)# exit
Switch(config)# router bgp 50000

Switch(config-router)# address-family ipv4
Switch(config-router-af)# bgp dampening route-map BLUE
Switch(config-router-af)# end
```

Monitoring and Maintaining BGP

You can remove all contents of a particular cache, table, or database. This might be necessary when the contents of the particular structure have become or are suspected to be invalid.

You can display specific statistics, such as the contents of BGP routing tables, caches, and databases. You can use the information to get resource utilization and solve network problems. You can also display information about node reachability and discover the routing path your device's packets are taking through the network.

Command	Purpose
clear ip bgp *address*	Reset a particular BGP connection.
clear ip bgp *	Reset all BGP connections.
clear ip bgp peer-group *tag*	Remove all members of a BGP peer group.
show ip bgp *prefix*	Display peer groups and peers not in peer groups to which the prefix has been advertised. Also display prefix attributes such as the next hop and the local prefix.
show ip bgp cidr-only	Display all BGP routes that contain subnet and supernet network masks.
show ip bgp community [*community-number*] [**exact**]	Display routes that belong to the specified communities.
show ip bgp community-list *community-list-number* [**exact-match**]	Display routes that are permitted by the community list.
show ip bgp filter-list *access-list-number*	Display routes that are matched by the specified AS path access list.
show ip bgp inconsistent-as	Display the routes with inconsistent originating autonomous systems.
show ip bgp regexp *regular-expression*	Display the routes that have an AS path that matches the specified regular expression entered on the command line.
show ip bgp	Display the contents of the BGP routing table.
show ip bgp neighbors [*address*]	Display detailed information on the BGP and TCP connections to individual neighbors.
show ip bgp neighbors [*address*] [**advertised-routes** | **dampened-routes** | **flap-statistics** | **paths** *regular-expression* | **received-routes** | **routes**]	Display routes learned from a particular BGP neighbor.
show ip bgp paths	Display all BGP paths in the database.
show ip bgp peer-group [*tag*] [**summary**]	Display information about BGP peer groups.
show ip bgp summary	Display the status of all BGP connections.

You can also enable the logging of messages generated when a BGP neighbor resets, comes up, or goes down by using the **bgp log-neighbor changes** router configuration command.

Configuring ISO CLNS Routing

The International Organization for Standardization (ISO) Connectionless Network Service (CLNS) protocol is a standard for the network layer of the Open System Interconnection (OSI) model. Addresses in the ISO network architecture are referred to as network service access point (NSAP) addresses and network entity titles (NETs). Each node in an OSI network has one or more NETs. In addition, each node has many NSAP addresses.

When you enable connectionless routing on the switch by using the **clns routing** global configuration command, the switch makes only forwarding decisions, with no routing-related functionality. For dynamic routing, you must also enable a routing protocol. The switch supports the Intermediate System-to-Intermediate System (IS-IS) dynamic routing protocols for ISO CLNS networks. This routing protocol supports the concept of *areas*. Within an area, all routers know how to reach all the system IDs. Between areas, routers know how to reach the proper area. IS-IS supports two levels of routing: *station routing* (within an area) and *area routing* (between areas).

The key difference between the ISO IGRP and IS-IS NSAP addressing schemes is in the definition of area addresses. Both use the system ID for Level 1 routing (routing within an area). However, they differ in the way addresses are specified for area routing. An ISO IGRP NSAP address includes three separate fields for routing: the *domain*, *area*, and *system ID*. An IS-IS address includes two fields: a single continuous *area* field (comprising the domain and area fields) and the *system ID*.

For more detailed information about ISO CLNS, see the ISO CLNS documents listed in the Related Documents, page 966.

Configuring IS-IS Dynamic Routing

IS-IS is an ISO dynamic routing protocol. Enabling IS-IS requires that you create an IS-IS routing process and assign it to a specific interface, rather than to a network. You can specify more than one IS-IS routing process per Layer 3 switch or router by using the multiarea IS-IS configuration syntax. You then configure the parameters for each instance of the IS-IS routing process.

Small IS-IS networks are built as a single area that includes all the routers in the network. As the network grows larger, it is usually reorganized into a backbone area made up of the connected set of all Level 2 routers from all areas, which is in turn connected to local areas. Within a local area, routers know how to reach all system IDs. Between areas, routers know how to reach the backbone, and the backbone routers know how to reach other areas.

Routers establish Level 1 adjacencies to perform routing within a local area (station routing). Routers establish Level 2 adjacencies to perform routing between Level 1 areas (area routing).

A single Cisco router can participate in routing in up to 29 areas and can perform Level 2 routing in the backbone. In general, each routing process corresponds to an area. By default, the first instance of the routing process configured performs both Level 1and Level 2 routing. You can configure additional router instances, which are automatically treated as Level 1 areas. You must configure the parameters for each instance of the IS-IS routing process individually.

For IS-IS multiarea routing, you can configure only one process to perform Level 2 routing, although you can define up to 29 Level 1 areas for each Cisco unit. If Level 2 routing is configured on any process, all additional processes are automatically configured as Level 1. You can configure this process to perform Level 1 routing at the same time. If Level 2 routing is not desired for a router instance, remove the Level 2 capability using the **is-type** global configuration command. Use the **is-type** command also to configure a different router instance as a Level 2 router.

This section briefly describes how to configure IS-IS routing. For more detailed information about IS-IS, see the IS-IS documents listed in the Related Documents, page 966.

This section includes the following topics:

- Default IS-IS Configuration, page 914
- Nonstop Forwarding Awareness, page 914
- Configuring IS-IS Global Parameters, page 917

- Configuring IS-IS Interface Parameters, page 921

Default IS-IS Configuration

Feature	Default Setting
Ignore link-state PDU (LSP) errors	Enabled.
IS-IS type	Conventional IS-IS: the router acts as both a Level 1 (station) and a Level 2 (area) router. Multiarea IS-IS: the first instance of the IS-IS routing process is a Level 1-2 router. Remaining instances are Level 1 routers.
Default-information originate	Disabled.
Log IS-IS adjacency state changes.	Disabled.
LSP generation throttling timers	Maximum interval between two consecutive occurrences: 5 seconds. Initial LSP generation delay: 50 ms. Hold time between the first and second LSP generation: 5000 ms.
LSP maximum lifetime (without a refresh)	1200 seconds (20 minutes) before the LSP packet is deleted.
LSP refresh interval	Send LSP refreshes every 900 seconds (15 minutes).
Maximum LSP packet size	1497 bytes.
NSF[1] Awareness	Enabled[2]. Allows Layer 3 switches to continue forwarding packets from a neighboring NSF-capable router during hardware or software changes.
Partial route computation (PRC) throttling timers	Maximum PRC wait interval: 5 seconds. Initial PRC calculation delay after a topology change: 2000 ms. Hold time between the first and second PRC calculation: 5000 ms.
Partition avoidance	Disabled.
Password	No area or domain password is defined, and authentication is disabled.
Set-overload-bit	Disabled. When enabled, if no arguments are entered, the overload bit is set immediately and remains set until you enter the **no set-overload-bit** command.
Shortest path first (SPF) throttling timers	Maximum interval between consecutive SFPs: 10 seconds. Initial SFP calculation after a topology change: 5500 ms. Holdtime between the first and second SFP calculation: 5500 ms.
Summary-address	Disabled.

[1] NSF = Nonstop Forwarding

[2] IS-IS NSF awareness is enabled for IPv4 on switches running the IP services image.

Nonstop Forwarding Awareness

The integrated IS-IS NSF Awareness feature is supported for IPv4 in the IP services image. The feature allows customer premises equipment (CPE) routers that are NSF-aware to help NSF-capable routers perform nonstop forwarding of packets. The local router is not necessarily performing NSF, but its awareness of NSF allows the integrity and accuracy of the routing database and link-state database on the neighboring NSF-capable router to be maintained during the switchover process.

This feature is automatically enabled and requires no configuration. For more information on this feature, see the "Configuring Nonstop Forwarding" chapter in the *High Availability Configuration Guide, Cisco IOS Release 15S*.

Enabling IS-IS Routing

To enable IS-IS, you specify a name and NET for each routing process. You then enable IS-IS routing on the interface and specify the area for each instance of the routing process.

BEFORE YOU BEGIN

You should know your network design and how you want traffic to flow through it before configuring IS-IS. Define areas, prepare an addressing plan for the devices (including defining the NETs), and determine the interfaces that will run integrated IS-IS. To facilitate verification, a matrix of adjacencies should be prepared before you configure your devices, showing what neighbors should be expected in the adjacencies table.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**clns routing**	Enable ISO connectionless routing on the switch.
3.	**router isis** [*area tag*]	Enable the IS-IS routing for the specified routing process and enter IS-IS routing configuration mode. (Optional) Use the *area tag* argument to identify the area to which the IS-IS router is assigned. You must enter a value if you are configuring multiple IS-IS areas. The first IS-IS instance configured is Level 1-2 by default. Later instances are automatically Level 1. You can change the level of routing by using the **is-type** global configuration command.
4.	**net** *network-entity-title*	Configure the NETs for the routing process. If you are configuring multiarea IS-IS, specify a NET for each routing process. You can specify a name for a NET and for an address.
5.	**is-type** {**level-1** | **level-1-2** | **level-2-only**}	(Optional) You can configure the router to act as a Level 1 (station) router, a Level 2 (area) router for multi-area routing, or both (the default): ■ **level-1**–act as a station router only ■ **level-1-2**–act as both a station router and an area router ■ **level 2**–act as an area router only
6.	**exit**	Return to global configuration mode.
7.	**interface** *interface-id*	Specify an interface to route IS-IS, and enter interface configuration mode. If the interface is not already configured as a Layer 3 interface, enter the **no switchport** command to put it into Layer 3 mode.
8.	**no shutdown**	Enable the interface if necessary. By default, UNIs and ENIs are disabled and NNIs are enabled.
9.	**ip router isis** [*area tag*]	Configure an IS-IS routing process for ISO CLNS on the interface and attach an area designator to the routing process.
10.	**clns router isis** [*area tag*]	Enable ISO CLNS on the interface.
11.	**ip address** *ip-address-mask*	Define the IP address for the interface. An IP address is required on all interfaces in an area enabled for IS-IS if any one interface is configured for IS-IS routing.
12.	**end**	Return to privileged EXEC mode.
13.	**show isis** [*area tag*] **database detail**	Verify your entries.
14.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable IS-IS routing, use the **no router isis** *area-tag* router configuration command.

EXAMPLE

This example shows how to configure three routers to run conventional IS-IS as an IP routing protocol. In conventional IS-IS, all routers act as Level 1 and Level 2 routers (by default).

Router A:

```
Switch(config)# clns routing
Switch(config)# router isis
Switch(config-router)# net 49.0001.0000.0000.000a.00
Switch(config-router)# exit
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip router isis
Switch(config-if)# clns router isis
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# ip router isis
Switch(config-if)# clns router isis
Switch(config-router)# exit
```

Router B:

```
Switch(config)# clns routing
Switch(config)# router isis
Switch(config-router)# net 49.0001.0000.0000.000b.00
Switch(config-router)# exit
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip router isis
Switch(config-if)# clns router isis
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# ip router isis
Switch(config-if)# clns router isis
Switch(config-router)# exit
```

Router C:

```
Switch(config)# clns routing
Switch(config)# router isis
Switch(config-router)# net 49.0001.0000.0000.000c.00
Switch(config-router)# exit
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip router isis
Switch(config-if)# clns router isis
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# ip router isis
Switch(config-if)# clns router isis
Switch(config-router)# exit
```

Configuring IS-IS Global Parameters

These are some optional IS-IS global parameters that you can configure:

- You can force a default route into an IS-IS routing domain by configuring a default route controlled by a route map. You can also specify other filtering options configurable under a route map.
- You can configure the router to ignore IS-IS LSPs that are received with internal checksum errors or to purge corrupted LSPs, which causes the initiator of the LSP to regenerate it.
- You can assign passwords to areas and domains.
- You can create aggregate addresses that are represented in the routing table by a summary address (route-summarization). Routes learned from other routing protocols can also be summarized. The metric used to advertise the summary is the smallest metric of all the specific routes.
- You can set an overload bit.

- You can configure the LSP refresh interval and the maximum time that an LSP can remain in the router database without a refresh
- You can set the throttling timers for LSP generation, shortest path first computation, and partial route computation.
- You can configure the switch to generate a log message when an IS-IS adjacency changes state (up or down).
- If a link in the network has a maximum transmission unit (MTU) size of less than 1500 bytes, you can lower the LSP MTU so that routing will still occur.
- The partition avoidance router configuration command prevents an area from becoming partitioned when full connectivity is lost among a Level1-2 border router, adjacent Level 1 routers, and end hosts.

BEFORE YOU BEGIN

Enable IS-IS routing as described in the Enabling IS-IS Routing, page 915.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**clns routing**	Enable ISO connectionless routing on the switch.
3.	**router isis**	Specify the IS-IS routing protocol and enter router configuration mode.
4.	**default-information originate** [**route-map** *map-name*]	(Optional) Force a default route into the IS-IS routing domain. If you enter **route-map** *map-name,* the routing process generates the default route if the route map is satisfied.
5.	**ignore-lsp-errors**	(Optional) Configure the router to ignore LSPs with internal checksum errors, instead of purging the LSPs. This command is enabled by default (corrupted LSPs are dropped). To purge the corrupted LSPs, enter the **no ignore-lsp-errors** router configuration command.
6.	**area-password** *password*	(Optional Configure the area authentication password, which is inserted in Level 1 (station router level) LSPs.
7.	**domain-password** *password*	(Optional) Configure the routing domain authentication password, which is inserted in Level 2 (area router level) LSPs.
8.	**summary-address** *address mask* [**level-1** | **level-1-2** | **level-2**]	(Optional) Create a summary of addresses for a given level.
9.	**set-overload-bit** [**on-startup** {*seconds* | **wait-for-bgp**}]	(Optional) Set an overload bit (a hippity bit) to allow other routers to ignore the router in their shortest path first (SPF) calculations if the router is having problems. ■ (Optional) **on-startup**—sets the overload bit only on startup. If **on-startup** is not specified, the overload bit is set immediately and remains set until you enter the **no set-overload-bit** command. If **on-startup** is specified, you must enter a number of seconds or **wait-for-bgp**. ■ *seconds*—When the **on-startup** keyword is configured, causes the overload bit to be set upon system startup and remain set for this number of seconds. The range is from 5 to 86400 seconds. ■ **wait-for-bgp**—When the **on-startup** keyword is configured, causes the overload bit to be set upon system startup and remain set until BGP has converged. If BGP does not signal IS-IS that it is converged, IS-IS will turn off the overload bit after 10 minutes.
10.	**lsp-refresh-interval** *seconds*	(Optional) Set an LSP refresh interval in seconds. The range is from 1 to 65535 seconds. The default is to send LSP refreshes every 900 seconds (15 minutes).
11.	**max-lsp-lifetime** *seconds*	(Optional) Set the maximum time that LSP packets remain in the router database without being refreshed. The range is from 1 to 65535 seconds. The default is 1200 seconds (20 minutes). After the specified time interval, the LSP packet is deleted.

	Command	Purpose
12.	**lsp-gen-interval** [**level-1** | **level-2**] *lsp-max-wait* [*lsp-initial-wait lsp-second-wait*]	(Optional) Set the IS-IS LSP generation throttling timers: ■ *lsp-max-wait*—the maximum interval (in seconds) between two consecutive occurrences of an LSP being generated. The range is 1 to 120, the default is 5. ■ *lsp-initial-wait*—the initial LSP generation delay (in milliseconds). The range is 1 to 10000; the default is 50. ■ *lsp-second-wait*—the hold time between the first and second LSP generation (in milliseconds). The range is 1 to 10000; the default is 5000.
13.	**spf-interval** [**level-1** | **level-2**] *spf-max-wait* [*spf-initial-wait spf-second-wait*]	(Optional) Sets IS-IS shortest path first (SPF) throttling timers. ■ *spf-max-wait*—the maximum interval between consecutive SFPs (in seconds). The range is 1 to 120, the default is 10. ■ *spf-initial-wait*—the initial SFP calculation after a topology change (in milliseconds). The range is 1 to 10000; the default is 5500. ■ *spf-second-wait*—the holdtime between the first and second SFP calculation (in milliseconds). The range is 1 to 10000; the default is 5500.
14.	**prc-interval** *prc-max-wait* [*prc-initial-wait prc-second-wait*]	(Optional) Sets IS-IS partial route computation (PRC) throttling timers. ■ *prc-max-wait*—the maximum interval (in seconds) between two consecutive PRC calculations. The range is 1 to 120; the default is 5. ■ *prc-initial-wait*—the initial PRC calculation delay (in milliseconds) after a topology change. The range is 1 to 10,000; the default is 2000. ■ *prc-second-wait*—the hold time between the first and second PRC calculation (in milliseconds). The range is 1 to 10,000; the default is 5000.
15.	**log-adjacency-changes** [**detail**]	(Optional) Set the router to log IS-IS adjacency state changes. Enter **detail** to include all changes generated by events that are not related to the Intermediate System-to-Intermediate System Hellos, including End System-to-Intermediate System PDUs and link state packets (LSPs).
16.	**lsp-mtu** *size*	(Optional) Specify the maximum LSP packet size in bytes. The range is 128 to 4352; the default is 1497 bytes. **Note:** If any link in the network has a reduced MTU size, you must change the LSP MTU size on all routers in the network.
17.	**partition avoidance**	(Optional) Causes an IS-IS Level 1-2 border router to stop advertising the Level 1 area prefix into the Level 2 backbone when full connectivity is lost among the border router, all adjacent level 1 routers, and end hosts.

	Command	Purpose
18.	**end**	Return to privileged EXEC mode.
19.	**show clns**	Verify your entries.
20.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable default route generation, use the **no default-information originate** router configuration command. Use the **no area-password** or **no domain-password** router configuration command to disable passwords. To disable LSP MTU settings, use the **no lsp mtu** router configuration command. To return to the default conditions for summary addressing, LSP refresh interval, LSP lifetime, LSP timers, SFP timers, and PRC timers, use the **no** form of the commands. Use the **no partition avoidance** router configuration command to disable the output format.

EXAMPLE

```
Switch(config)# clns routing
Switch(config)# router isis
Switch(config-router)# set-overloadbit on-startup 360
Switch(config-router)# log-adjacency-changes
Switch(config-router)# ignore-lsp-errors
Switch(config-router)# max-lsp-lifetime 65535
Switch(config-router)# lsp-refresh-interval 65000
Switch(config-router)# spf-interval 5 1 50
Switch(config-router)# prc-interval 5 1 50
Switch(config-router)# lsp-gen-interval 5 1 50
Switch(config-router)# end
```

Configuring IS-IS Interface Parameters

You can optionally configure certain interface-specific IS-IS parameters, independently from other attached routers. However, if you change some values from the defaults, such as multipliers and time intervals, it makes sense to also change them on multiple routers and interfaces. Most of the interface parameters can be configured for level 1, level 2, or both.

These are some interface level parameters you can configure:

- The default metric on the interface, which is used as a value for the IS-IS metric and assigned when there is no quality of service (QoS) routing performed.
- The hello interval (length of time between hello packets sent on the interface) or the default hello packet multiplier used on the interface to determine the hold time sent in IS-IS hello packets. The hold time determines how long a neighbor waits for another hello packet before declaring the neighbor down. This determines how quickly a failed link or neighbor is detected so that routes can be recalculated. Change the hello multiplier in circumstances where hello packets are lost frequently and IS-IS adjacencies are failing unnecessarily. You can raise the hello multiplier and lower the hello interval correspondingly to make the hello protocol more reliable without increasing the time required to detect a link failure.
- Other time intervals:
  - Complete sequence number PDU (CSNP) interval. CSNPs are sent by the designated router to maintain database synchronization.
  - Retransmission interval. This is the time between retransmission of IS-IS LSPs for point-to-point links.
  - IS-IS LSP retransmission throttle interval. This is the maximum rate (number of milliseconds between packets) at which IS-IS LSPs are re-sent on point-to-point links This interval is different from the retransmission interval, which is the time between successive retransmissions of the *same* LSP.

- Designated router election priority, which allows you to reduce the number of adjacencies required on a multiaccess network, which in turn reduces the amount of routing protocol traffic and the size of the topology database.
- The interface circuit type, which is the type of adjacency desired for neighbors on the specified interface.
- Password authentication for the interface.

BEFORE YOU BEGIN

Enable IS-IS routing as described in the Enabling IS-IS Routing, page 915.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Specify the interface to be configured and enter interface configuration mode. If the interface is not already configured as a Layer 3 interface, enter the **no switchport** command to put it into Layer 3 mode.
3.	**no shutdown**	Enable the interface if necessary. By default, UNIs and ENIs are disabled and NNIs are enabled.
4.	**isis metric** *default-metric* [**level-1** | **level-2**]	(Optional) Configure the metric (or cost) for the specified interface. The range is from 0 to 63. The default is 10. If no level is entered, the default is to apply to both Level 1 and Level 2 routers.
5.	**isis hello-interval** {*seconds* | **minimal**} [**level-1** | **level-2**]	(Optional) Specify the length of time between hello packets sent by the switch. By default, a value three times the hello interval *seconds* is advertised as the *holdtime* in the hello packets sent. With smaller hello intervals, topological changes are detected faster, but there is more routing traffic. ■ **minimal–**causes the system to compute the hello interval based on the hello multiplier so that the resulting hold time is 1 second. ■ *seconds*–the range is from 1 to 65535. The default is 10 seconds.
6.	**isis hello-multiplier** *multiplier* [**level-1** | **level-2**]	(Optional) Specify the number of IS-IS hello packets a neighbor must miss before the router should declare the adjacency as down. The range is from 3 to 1000. The default is 3. Using a smaller hello multiplier causes fast convergence, but can result in more routing instability.
7.	**isis csnp-interval** *seconds* [**level-1** | **level-2**]	(Optional) Configure the IS-IS complete sequence number PDU (CSNP) interval for the interface. The range is from 0 to 65535. The default is 10 seconds.
8.	**isis retransmit-interval** *seconds*	(Optional) Configure the number of seconds between retransmission of IS-IS LSPs for point-to-point links. The value you specify should be an integer greater than the expected round-trip delay between any two routers on the network. The range is from 0 to 65535. The default is 5 seconds.
9.	**isis retransmit-throttle-interval** *milliseconds*	(Optional) Configure the IS-IS LSP retransmission throttle interval, which is the maximum rate (number of milliseconds between packets) at which IS-IS LSPs will be re-sent on point-to-point links. The range is from 0 to 65535. The default is determined by the **isis lsp-interval** command.
10.	**isis priority** *value* [**level-1** | **level-2**]	(Optional) Configure the priority to use for designated router election. The range is from 0 to 127. The default is 64.

	Command	Purpose
11.	**isis circuit-type** {**level-1** | **level-1-2** | **level-2-only**}	(Optional) Configure the type of adjacency desired for neighbors on the specified interface (specify the interface circuit type). ■ **level-1**—a Level 1 adjacency is established if there is at least one area address common to both this node and its neighbors. ■ **level-1-2**—a Level 1 and 2 adjacency is established if the neighbor is also configured as both Level 1 and Level 2 and there is at least one area in common. If there is no area in common, a Level 2 adjacency is established. This is the default. ■ **level 2**—a Level 2 adjacency is established. If the neighbor router is a Level 1 router, no adjacency is established.
12.	**isis password** *password* [**level-1** | **level-2**]	(Optional) Configure the authentication password for an interface. By default, authentication is disabled. Specifying Level 1 or Level 2 enables the password only for Level 1 or Level 2 routing, respectively. If you do not specify a level, the default is Level 1 and Level 2.
13.	**end**	Return to privileged EXEC mode.
14.	**show clns interface** *interface-id*	Verify your entries.
15.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default settings, use the **no** forms of the commands.

EXAMPLE

The following configuration example for an IS-IS routing process called area1 sets a global default metric of 111 for the IS-IS interfaces:

```
interface Ethernet3/1
ip address 172.16.10.2 255.255.0.0
ip router isis area1
no ip route-cache
duplex half
!
interface Ethernet3/2
ip address 192.168.242.2 255.255.255.0
ip router isis area1
no ip route-cache
duplex half
router isis area1
net 01.0000.0309.1234.00
metric-style wide
metric 111
```

Monitoring and Maintaining IS-IS

You can remove all contents of a CLNS cache or remove information for a particular neighbor or route. You can display specific CLNS or IS-IS statistics, such as the contents of routing tables, caches, and databases. You can also display information about specific interfaces, filters, or neighbors.

Command	Purpose
clear clns cache	Clear and reinitialize the CLNS routing cache.
clear clns es-neighbors	Remove end system (ES) neighbor information from the adjacency database.
clear clns is-neighbors	Remove intermediate system (IS) neighbor information from the adjacency database.
clear clns neighbors	Remove CLNS neighbor information from the adjacency database.
clear clns route	Remove dynamically derived CLNS routing information.
show clns	Display information about the CLNS network.
show clns cache	Display the entries in the CLNS routing cache.
show clns es-neighbors	Display ES neighbor entries, including the associated areas.
show clns filter-expr	Display filter expressions.
show clns filter-set	Display filter sets.
show clns interface [*interface-id*]	Display the CLNS-specific or ES-IS information about each interface.
show clns neighbor	Display information about IS-IS neighbors.
show clns protocol	List the protocol-specific information for each IS-IS or ISO IGRP routing process in this router.
show clns route	Display all the destinations to which this router knows how to route CLNS packets.
show clns traffic	Display information about the CLNS packets this router has seen.
show ip route isis	Display the current state of the IS-IS IP routing table.
show isis database	Display the IS-IS link-state database.
show isis routes	Display the IS-IS Level 1 routing table.
show isis spf-log	Display a history of the shortest path first (SPF) calculations for IS-IS.
show isis topology	Display a list of all connected routers in all areas.
show route-map	Display all route maps configured or only the one specified.
trace clns *destination*	Discover the paths taken to a specified destination by packets in the network.
which-route {*nsap-address* | *clns-name*}	Display the routing table in which the specified CLNS destination is found.

Configuring BFD

The Bidirectional Forwarding Detection (BFD) Protocol quickly detects forwarding-path failures for a variety of media types, encapsulations, topologies, and routing protocols. It operates in a unicast, point-to-point mode on top of any data protocol being forwarded between two systems to track IPv4 connectivity between directly connected neighbors. BFD packets are encapsulated in UDP packets with a destination port number of 3784 or 3785.

In EIGRP, IS-IS, and OSPF deployments, the closest alternative to BFD is the use of modified failure-detection mechanisms. Although reducing the EIGRP, IS-IS, and OSPF timers can result in a failure-detection rate of 1 to 2 seconds, BFD can provide failure detection in less than 1 second. BFD can be less CPU-intensive than the reduced timers and, because it is not tied to any particular routing protocol, it can be used as a generic and consistent failure detection mechanism for multiple routing protocols.

To create a BFD session, you must configure BFD on both systems (BFD peers). Enabling BFD at the interface and routing protocol level on BFD peers creates a BFD session. BFD timers are negotiated and the BFD peers send control packets to each other at the negotiated intervals. If the neighbor is not directly connected, BFD neighbor registration is rejected.

Figure 104 on page 926 shows a simple network with two routers running OSPF and BFD. When OSPF discovers a neighbor (1), it sends a request to the BFD process to initiate a BFD neighbor session with the neighbor OSPF router (2), establishing the BFD neighbor session (3).

Figure 104 Establishing a BFD Session

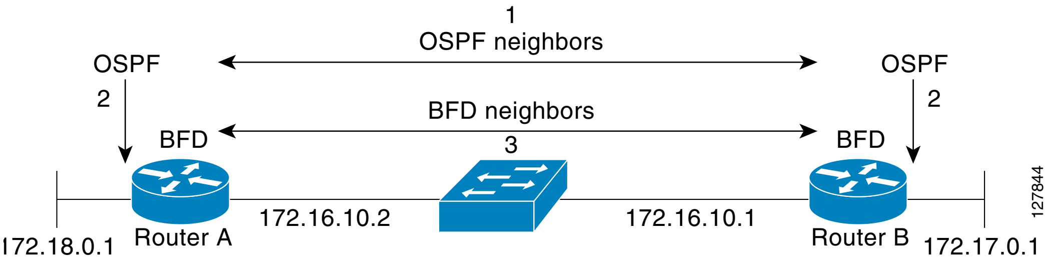
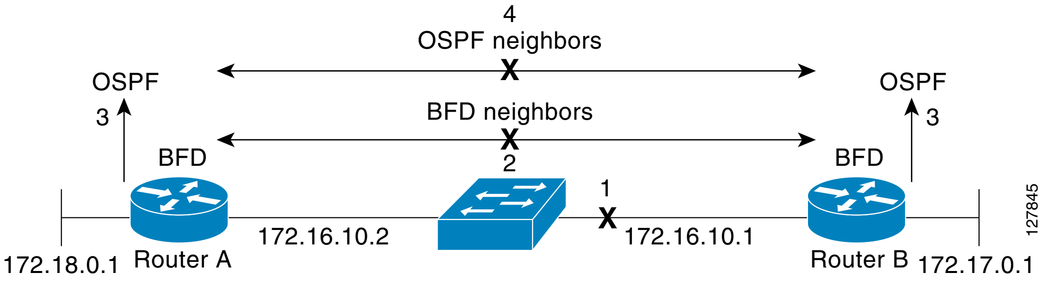


Figure 105 on page 926 shows what happens when a failure occurs in the network (1). The BFD neighbor session with the OSPF neighbor closes (2). BFD notifies the OSPF process that the BFD neighbor is no longer reachable, and the OSPF process breaks the OSPF neighbor relationship (4). If an alternative path is available, the routers start converging on it.

Figure 105 Breaking an OSPF Neighbor Relationship



BFD clients are routing protocols that register neighbors with BFD. The switch supports IS-IS, OSPF v1 and v2, BGP, EIGRP, and HSRP clients. You can use one BFD session for multiple client protocols. For example, if a network is running OSPF and EIGRP across the same link to the same peer, you need to create only one BFD session, and information is shared with both routing protocols.

The switch supports BFD version 0 and version 1. BFD neighbors automatically negotiate the version and the protocol always runs at the higher version. The default version is version 1.

By default, BFD neighbors exchange both control packets and echo packets for detecting forwarding failures. The switch sends echo packets at the configured BFD interval rate (from 50 to 999 ms), and control packets at the BFD slow-timer rate (from 1000 to 3000 ms).

Failure-rate detection can be faster in BFD echo mode, which is enabled by default when you configure BFD session. In this mode, the switch sends echo packets from the BFD software layer, and the BFD neighbor responds to the echo packets through its fast-switching layer. The echo packets do not reach the BFD neighbor software layer, but are reflected back over the forwarding path for failure detection. You configure the rate at which each BFD interface sends BFD echo packets by entering the **bfd interval** interface configuration command.

To reduce bandwidth consumption, you can disable the sending of echo packets by entering the **no bfd echo** interface configuration command. When echo mode is disabled, control packets are used to detect forwarding failures. Control packets are exchanged at the configured slow-timer rate, which could result in longer failure-detection time. You configure this rate by entering the **bfd slow-timer** global configuration command. The range is from 1000 to 3000 ms; the default rate is every 1000 ms.

You can enable or disable echo processing at a switch interface independent of the BFD neighbor configuration. Disabling echo mode only disables the sending of echo packets by the interface. The fast-switching layer that receives an echo packet always reflects it back to the sender.

To run BFD on a switch, you need to configure basic BFD interval parameters on BFD interfaces, enable routing on the switch, and enable one or more one routing protocol clients for BFD. You also need to confirm that Cisco Express Forwarding (CEF) is enabled (the default) on participating switches.

For more information on the configuration and commands, see the BFD documents listed in the Related Documents, page 966.

This section includes the following topics:

- Default BFD Configuration, page 927
- Default BFD Configuration Guidelines, page 927
- Configuring BFD Session Parameters on an Interface, page 928
- Enabling BFD Routing Protocol Clients, page 929

Default BFD Configuration

- No BFD sessions are configured. BFD is disabled on all interfaces.
- When configured, BFD version 1 is the default, but switches negotiate for version. Version 0 is also supported.
- Standby BFD (for HSRP) is enabled by default.
- Asynchronous BFD echo mode is enabled when a BFD session is configured.

Default BFD Configuration Guidelines

The switch supports a maximum of 28 BFD sessions at one time.

To run BFD on a switch:

- Configure basic BFD interval parameters on each interface over which you want to run BFD sessions.
- Enable routing on the switch. You can configure BFD without enabling routing, but BFD sessions do not become active unless routing is enabled on the switch and on the BFD interfaces.
- Enable one or more one routing protocol clients for BFD. You should implement fast convergence for the routing protocol that you are using.

Note: We recommend that you configure the BFD interval parameters on an interface before configuring the routing protocol commands, especially when using EIGRP.

Confirm that CEF is enabled on participating switches (the default) as well as IP routing.

BFD is supported on physical interfaces that are configured as routing interfaces. It is not supported on Layer 2 interfaces, pseudowires, static routes, SVI interfaces, or port channels.

Although you can configure BFD interface commands on a Layer 2 port, BFD sessions do not operate on the interface unless it is configured as a Layer 3 interface (no switchport) and assigned an IP address.

In HSRP BFD, standby BFD is enabled globally by default and on all interfaces. If you disable it on an interface, you then must disable and reenable it globally for BFD sessions to be active.

When using BFD echo mode (the default), you should disable sending of ICMP redirect messages by entering the **no ip redirects** interface configuration command on the BFD interface.

Configuring BFD Session Parameters on an Interface

Before you can start a BFD session on an interface, you must put the interface into Layer 3 mode and set the baseline BFD parameters on it.

Note: Although you can configure BFD on Layer 2 interfaces, a BFD session cannot start until both interfaces are in Layer 3 mode and routing is enabled on the switch.

BEFORE YOU BEGIN

See Default BFD Configuration Guidelines, page 927.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Specify an interface for a BFD session, and enter interface configuration mode. Only physical interfaces support BFD.
3.	**no shutdown**	Enable the interface if necessary. User network interfaces (UNIs) and enhanced network interfaces (ENIs) are disabled by default; network node interfaces (NNIs) are enabled by default.
4.	**no switchport**	Remove the interface from Layer 2 configuration mode.
5.	**ip address** *ip-address subnet-mask*	Configure the IP address and IP subnet mask.
6.	**bfd interval** *milliseconds* **min_rx** *milliseconds* **multiplier** *value*	Set BFD parameters for echo packets on the interface. ■ **interval**–Specify the rate at which BFD echo packets are sent to BFD peers. The range is from 50 to 999 milliseconds (ms). ■ **min_rx**–Specify the rate at which BFD echo packets are expected to be received from BFD peers. The range is from 50 to 999 ms. ■ **multiplier**–Specify the number of consecutive BFD echo packets that must be missed from a BFD peer before BFD declares that it is unavailable and informs the other BFD peer of the failure. The range is from 3 to 50. **Note:** There are no baseline BFD parameter defaults.
7.	**end**	Return to privileged EXEC mode.
8.	**show running-config**	Verify your entries.
9.	**show bfd neighbor detail**	(Optional) Display the final configured or negotiated values when the session is created with a neighbor.
10.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the BFD parameter configuration, enter the **no bfd interval** interface configuration command.

Enabling BFD Routing Protocol Clients

After you configure BFD parameters on an interface, you can start a BFD session for one or more routing protocols. You must first enable routing by entering the **ip routing** global configuration command on the switch. Note that there can be more than one way to start a BFD session on an interface, depending on the routing protocol.

- Configuring BFD for OSPF, page 929
- Configuring BFD for IS-IS, page 930
- Configuring BFD for BGP, page 932
- Configuring BFD for EIGRP, page 933
- Configuring BFD for HSRP, page 934

Configuring BFD for OSPF

When you start BFD sessions for OSPF, OSPF must be running on all participating devices. You can enable BFD support for OSPF by enabling it globally on all OSPF interfaces or by enabling it on one or more interfaces.

Configuring BFD for OSPF Globally

BEFORE YOU BEGIN

- Configure BFD parameters as described in the Configuring BFD Session Parameters on an Interface, page 928.
- Configure OSPF as described in the Configuring OSPF, page 865.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**router ospf** *process- id*	Specify an OSPF process, and enter router configuration mode.
3.	**bfd all-interfaces**	Enable BFD globally on all interfaces associated with the OSPF routing process.
4.	**exit**	(Optional) Return to global configuration mode if you want to disable BFD on one or more OSPF interfaces.
5.	**interface** *interface-id*	(Optional) Specify an interface, and enter interface configuration mode.
6.	**ip ospf bfd disable**	(Optional) Disable BFD on the specified OSPF interface. Repeat Steps 5 and 6 for all OSPF interfaces on which you do not want to run BFD sessions.
7.	**end**	Return to privileged EXEC mode.
8.	**show bfd neighbors** [**detail**]	Verify the configuration.
9.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable OSPF BFD on all interfaces, enter the **no bfd all-interfaces** router configuration command.To disable it on an interface, enter the **no ip osfp bfd** or the **ip ospf bfd disable** interface configuration command on the interface.

If you want to run OSPF BFD on only one or a few interfaces, you can enter the **ip ospf bfd** interface configuration command on those interfaces instead of enabling it globally. See the next procedure.

Note: If you try to configure OSPF BFD on a Layer 2 interface, the configuration is not recognized.

EXAMPLE

This is an example of enabling BFD for OSPF on all OSPF interfaces:

```
Switch(config)# router ospf 109
Switch(config-router)# bfd all-interfaces
Switch(config-router)# exit
```

Configuring BFD for OSPF on an Interface

BEFORE YOU BEGIN

- Configure BFD parameters on the interface as described in the Configuring BFD Session Parameters on an Interface, page 928.
- Configure OSPF as described in the Configuring OSPF, page 865.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**router ospf** *process- id*	Specify an OSPF process, and enter router configuration mode.
3.	**exit**	Return to global configuration mode.
4.	**interface** *interface-id*	Specify an interface, and enter interface configuration mode.
5.	**ip ospf bfd**	Enable BFD on the specified OSPF interface. Repeat Steps 3 and 4 for all OSPF interfaces on which you want to run BFD sessions.
6.	**end**	Return to privileged EXEC mode.
7.	**show bfd neighbors** [**detail**]	Verify the configuration.
8.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable OSPF BFD on an interface, enter the **no ip osfp bfd** or the **ip ospf bfd disable** interface configuration command on the interface.

EXAMPLE

This is an example of enabling BFD for OSPF on a single interface:

```
Switch(config)# router ospf 109
Switch(config-router)# exit
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ip ospf bfd
```

Configuring BFD for IS-IS

When you start BFD sessions for IS-IS, IS-IS must be running on all devices participating in BFD. You can enable BFD support for IS-IS by enabling it globally on all IS-IS interfaces or by enabling it on one or more interfaces.

Configuring BFD for IS-IS Globally

BEFORE YOU BEGIN

- Configure BFD parameters on the interface as described in the Configuring BFD Session Parameters on an Interface, page 928.

- Configure IS-IS as described in the Configuring IS-IS Dynamic Routing, page 913.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**router is-is** *area-tag*	Specify an IS-IS process and enter router configuration mode.
3.	**bfd all-interfaces**	Enable BFD globally on all interfaces associated with the IS-IS routing process.
4.	**exit**	(Optional) Return to global configuration mode if you want to disable BFD on one or more IS-IS interfaces.
5.	**interface** *interface-id*	(Optional) Specify an interface and enter interface configuration mode.
6.	**ip router isis**	(Optional) Enable IPv4 IS-IS routing on the interface.
7.	**isis bfd disable**	(Optional) Disable BFD on the IS-IS interface. Repeat Steps 5 through 7 for all IS-IS interfaces on which you do not want to run BFD sessions.
8.	**end**	Return to privileged EXEC mode.
9.	**show bfd neighbors** [**detail**]	Verify the configuration.
10.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable IS-IS BFD on all interfaces, enter the **no bfd all-interfaces** router configuration command. To disable it on the specified interface, enter the **no isis bfd** or the **isis bfd disable** interface configuration command on the interface.

If you only want to run IS-IS BFD on a few interfaces, instead of enabling it globally, you can enter the **isis bfd** interface configuration command on those interfaces. See the next procedure.

Note: Although IS-IS BFD operates only on Layer 3 interfaces, you can configure it on interfaces in Layer 2 or Layer 3 mode. When you enable it, you see this message:

```
%ISIS BFD is reverting to router mode configuration, and remains disabled.
```

EXAMPLE

This is an example of setting fast convergence and enabling BFD for IS-IS on all IS-IS interfaces:

```
Switch(config)# router is-is tag1
Switch(config-router)# bfd all-interfaces
Switch(config-router)# exit
```

Configuring BFD for IS-IS on an Interface

BEFORE YOU BEGIN

- Configure BFD parameters on the interface as described in the Configuring BFD Session Parameters on an Interface, page 928.
- Configure IS-IS as described in the Configuring IS-IS Dynamic Routing, page 913.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**router is-is** *area-tag*	Specify an IS-IS process and enter router configuration mode.
3.	**exit**	Return to global configuration mode.
4.	**interface** *interface-id*	Specify an interface, and enter interface configuration mode.
5.	**isis bfd**	Enable BFD on the specified IS-IS interface. Repeat Steps 3 and 4 for all IS-IS interfaces on which you want to run BFD sessions.
6.	**end**	Return to privileged EXEC mode.
7.	**show bfd neighbors** [**detail**]	Verify the configuration.
8.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable IS-IS BFD on an interface, enter the **no isis bfd** or the **isis bfd disable** interface configuration command on the interface.

EXAMPLE

This is an example of enabling BFD for IS-IS on a single interface:

```
Switch(config)# router is-is tag1
Switch(config-router)# exit
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# isis bfd
```

Configuring BFD for BGP

When you start BFD sessions for BGP, BGP must be running on all participating devices. You enter the IP address of the BFD neighbor to enable BFD for BGP.

BEFORE YOU BEGIN

- Configure BFD parameters on the interface as described in the Configuring BFD Session Parameters on an Interface, page 928.

- Configure BGP as described in the Configuring BGP, page 888.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**router bgp** *as-tag*	Specify a BGP autonomous system, and enter router configuration mode.
3.	**neighbor** *ip-address* **fall-over bfd**	Enable BFD support for fallover on the BFD neighbor.
4.	**end**	Return to privileged EXEC mode.
5.	**show bfd neighbors** [**detail**] > **show ip bgp neighbor**	Verify the configuration. Display information about BGP connections to neighbors.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable BGP BFD, enter the **no neighbor** *ip-address* **fall-over bfd** router configuration command.

Configuring BFD for EIGRP

When you start BFD sessions for EIGRP, EIGRP must be running on all participating devices.You can enable BFD support for EIGRP by globally enabling it on all EIGRP interfaces or by enabling it on one or more interfaces.

BEFORE YOU BEGIN

- Configure BFD parameters on the interface as described in the Configuring BFD Session Parameters on an Interface, page 928.
- Configure EIGRP as described in the Configuring EIGRP, page 880.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**router eigrp** *as-number*	Specify an EIGRP autonomous system number, and enter router configuration mode.
3.	**log-adjacency changes** [**detail**]	Configure the switch to send a system logging message when an EIGRP neighbor goes up or down.
4.	**bfd** {**all-interfaces |** interface *interface-id*}	Enable BFD for EIGRP. ■ Enter **all-interfaces** to globally enable BFD on all interfaces associated with the EIGRP routing process. ■ Enter **interface** *interface-id* to enable BFD on a per-interface basis for one or more interfaces associated with the EIGRP routing process.
5.	**end**	Return to privileged EXEC mode.
6.	**show bfd neighbors** [**detail**]	Verify the configuration.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable EIGRP BFD on all interfaces, enter the **no bfd all-interfaces** router configuration command. To disable it on an interface, enter the **no** bfd interface *interface-id* router configuration command.

Configuring BFD for HSRP

HSRP supports BFD by default; it is globally enabled on all interfaces. If HSRP support has been manually disabled, you can reenable it in interface or global configuration mode.

BEFORE YOU BEGIN

- Configure BFD parameters on the interface as described in the Configuring BFD Session Parameters on an Interface, page 928.
- Ensure that all participating devices have HSRP enabled and CEF enabled (the default).

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Specify an interface for a BFD session, and enter interface configuration mode. Only physical interfaces support BFD.
3.	**ip address** *ip-address subnet-mask*	Configure the IP address and IP subnet mask for the interface.
4.	**standby** [*group-number*] **ip** [*ip-address*] [**secondary**]]	Activate HSRP.
5.	**standby bfd**	(Optional) Enable HSRP support for BFD on the interface.
6.	exit	Return to global configuration mode.
7.	**standby bfd all-interfaces**	(Optional) Enable HSRP support for BFD on all interfaces.
8.	**end**	Return to privileged EXEC mode.
9.	**show standby neighbors**	Verify your entries.
10.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable HSRP support for BFD on all interfaces, enter the **no standby bfd all-interfaces** global configuration command. To disable it on an interface, enter the **no standby bfd** interface configuration command.

Note: If you disable standby BFD on an interface by entering the **no standby bfd** interface configuration command, to activate BFD sessions on other interfaces, you must disable and reenable it globally by entering the **no standby bfd all-interfaces** global configuration command followed by the **standby bfd all-interfaces** global configuration command.

EXAMPLE

The following example shows how to reenable HSRP BFD peering if it has been disabled on a switch:

```
Switch(config)# standby bfd all-interfaces
```

Disabling BFD Echo Mode

When you configure a BFD session, BFD echo mode is enabled by default on BFD interfaces. You can disable echo mode on an interface so it sends no echo packets and but only sends back echo packets received from a neighbor. When echo mode is disabled, control packets are used to detect forwarding failures. You can configure slow timers to reduce the frequency of BFD control packets.

BEFORE YOU BEGIN

Configure BFD parameters on the interface as described in the Configuring BFD Session Parameters on an Interface, page 928.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Enter a BFD interface and enter interface configuration mode.
3.	**no bfd echo**	Disable BFD echo mode on the interface. It is enabled by default, but can be disabled independently on BFD neighbors.
4.	**exit**	Return to global configuration mode.
5.	**bfd slow-timer** [*milliseconds*]	(Optional) Configure a BFD slow-timer value. The range is from 1000 to 30000 milliseconds. The default is 1000 milliseconds.
6.	**end**	Return to privileged EXEC mode.
7.	**show bfd neighbors detail**	Verify your entries.
8.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To reenable echo mode on the switch, enter the **bfd echo** global configuration command.

EXAMPLE

The following example disables echo mode between BFD neighbors:

```
Switch# configure terminal
Switch(config)# interface Ethernet 0/1
Switch(config-if)# no bfd echo
```

Configuring Multi-VRF CE

Virtual Private Networks (VPNs) provide a secure way for customers to share bandwidth over an ISP backbone network. A VPN is a collection of sites sharing a common routing table. A customer site is connected to the service-provider network by one or more interfaces, and the service provider associates each interface with a VPN routing table, called a VPN routing/forwarding (VRF) table.

The switch supports multiple VPN routing/forwarding (multi-VRF) instances in customer edge (CE) devices (multi-VRF CE). With multi-VRF CE, a service provider can support two or more VPNs with overlapping IP addresses.

Note: The switch does not use Multiprotocol Label Switching (MPLS) to support VPNs. For information about MPLS VRF, refer to the MPLS: Layer 3 VPNs Configuration Guide, Cisco IOS Release 15M&T.

Information About Multi-VRF CE

Multi-VRF CE allows a service provider to support two or more VPNs, where IP addresses can be overlapped among the VPNs. Multi-VRF CE uses input interfaces to distinguish routes for different VPNs and forms virtual packet-forwarding tables by associating one or more Layer 3 interfaces with each VRF. Interfaces in a VRF can be either physical, such as Ethernet ports, or logical, such as VLAN SVIs, but an interface cannot belong to more than one VRF at any time.

Note: Multi-VRF CE interfaces must be Layer 3 interfaces.

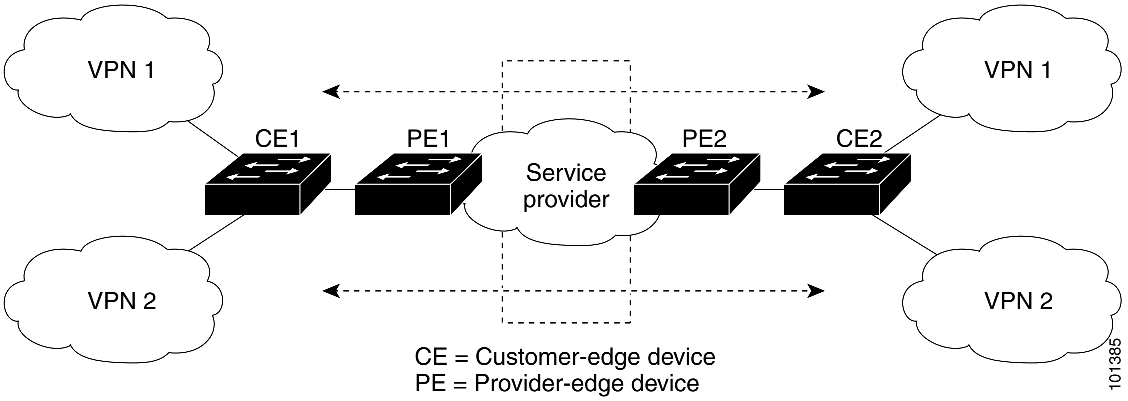
Multi-VRF CE includes these devices:

- Customer edge (CE) devices provide customers access to the service-provider network over a data link to one or more provider edge routers. The CE device advertises the site local routes to the router and learns the remote VPN routes from it. The Cisco Connected Grid switch can be a CE.
- Provider edge (PE) routers exchange routing information with CE devices by using static routing or a routing protocol such as BGP, RIPv2, OSPF, or EIGRP. The PE is only required to maintain VPN routes for those VPNs to which it is directly attached, eliminating the need for the PE to maintain all of the service-provider VPN routes. Each PE router maintains a VRF for each of its directly connected sites. Multiple interfaces on a PE router can be associated with a single VRF if all of these sites participate in the same VPN. Each VPN is mapped to a specified VRF. After learning local VPN routes from CEs, a PE router exchanges VPN routing information with other PE routers by using internal BGP (IBPG).
- Provider routers or core routers are any routers in the service provider network that do not attach to CE devices.

With multi-VRF CE, multiple customers can share one CE, and only one physical link is used between the CE and the PE. The shared CE maintains separate VRF tables for each customer and switches or routes packets for each customer based on its own routing table. Multi-VRF CE extends limited PE functionality to a CE device, giving it the ability to maintain separate VRF tables to extend the privacy and security of a VPN to the branch office.

Figure 106 on page 937 shows a configuration using Cisco Connected Grid switches as multiple virtual CEs. This scenario is suited for customers who have low bandwidth requirements for their VPN service, for example, small companies. In this case, multi-VRF CE support is required in the Cisco Connected Grid switches. Because multi-VRF CE is a Layer 3 feature, each interface in a VRF must be a Layer 3 interface.

Figure 106 Switches Acting as Multiple Virtual CEs



When the CE switch receives a command to add a Layer 3 interface to a VRF, it sets up the appropriate mapping between the VLAN ID and the policy label (PL) in multi-VRF-CE-related data structures and adds the VLAN ID and PL to the VLAN database.

When multi-VRF CE is configured, the Layer 3 forwarding table is conceptually partitioned into two sections:

- The multi-VRF CE routing section contains the routes from different VPNs.

- The global routing section contains routes to non-VPN networks, such as the Internet.

VLAN IDs from different VRFs are mapped into different policy labels, which are used to distinguish the VRFs during processing. If no route is found in the multi-VRF CE section of the Layer 3 forwarding table, the global routing section is used to determine the forwarding path. For each new VPN route learned, the Layer 3 setup function retrieves the policy label by using the VLAN ID of the ingress port and inserts the policy label and new route to the multi-VRF CE routing section. If the packet is received from a routed port, the port internal VLAN ID number is used; if the packet is received from an SVI, the VLAN number is used.

This is the packet-forwarding process in a multi-VRF-CE-enabled network:

- When the switch receives a packet from a VPN, the switch looks up the routing table based on the input policy label number. When a route is found, the switch forwards the packet to the PE.
- When the ingress PE receives a packet from the CE, it performs a VRF lookup. When a route is found, the router adds a corresponding MPLS label to the packet and sends it to the MPLS network.
- When an egress PE receives a packet from the network, it strips the label and uses the label to identify the correct VPN routing table. Then it performs the normal route lookup. When a route is found, it forwards the packet to the correct adjacency.
- When a CE receives a packet from an egress PE, it uses the input policy label to look up the correct VPN routing table. If a route is found, it forwards the packet within the VPN.

To configure VRF, you create a VRF table and specify the Layer 3 interface associated with the VRF. Then configure the routing protocols in the VPN and between the CE and the PE. BGP is the preferred routing protocol used to distribute VPN routing information across the provider’s backbone. The multi-VRF CE network has three major components:

- VPN route target communities—lists of all other members of a VPN community. You need to configure VPN route targets for each VPN community member.
- Multiprotocol BGP peering of VPN community PE routers—propagates VRF reachability information to all members of a VPN community. You need to configure BGP peering in all PE routers within a VPN community.
- VPN forwarding—transports all traffic between all VPN community members across a VPN service-provider network.

Default Multi-VRF CE Configuration

Feature	Default Setting
VRF	Disabled. No VRFs are defined.
Maps	No import maps, export maps, or route maps are defined.
VRF maximum routes	5000
Forwarding table	The default for an interface is the global routing table.

Multi-VRF CE Configuration Guidelines

These are considerations when configuring VRF in your network:

- A switch with multi-VRF CE is shared by multiple customers, and each customer has its own routing table.
- Because customers use different VRF tables, the same IP addresses can be reused. Overlapped IP addresses are allowed in different VPNs.
- Multi-VRF CE lets multiple customers share the same physical link between the PE and the CE. Trunk ports with multiple VLANs separate packets among customers. Each customer has its own VLAN.

- Multi-VRF CE does not support all MPLS-VRF functionality. It does not support label exchange, LDP adjacency, or labeled packets.
- For the PE router, there is no difference between using multi-VRF CE or using multiple CEs. In Figure 106 on page 937, multiple virtual Layer 3 interfaces are connected to the multi-VRF CE device.
- The switch supports configuring VRF by using physical ports, VLAN SVIs, or a combination of both. The SVIs can be connected through an access port or a trunk port.
- A customer can use multiple VLANs as long as they do not overlap with those of other customers. A customer's VLANs are mapped to a specific routing table ID that is used to identify the appropriate routing tables stored on the switch.
- The switch supports one global network and up to 26 VRFs.
- Most routing protocols (BGP, OSPF, RIP, EIGRP, and static routing) can be used between the CE and the PE. However, we recommend using external BGP (EBGP) for these reasons:
  - BGP does not require multiple algorithms to communicate with multiple CEs.
  - BGP is designed for passing routing information between systems run by different administrations.
  - BGP makes it easy to pass attributes of the routes to the CE.
- Multi-VRF CE does not affect the packet switching rate.
- If no VRFs are configured, up to 105 policies can be configured.
- If even one VRF is configured than 41 policies can be configured.
- If more than 41 policies are configured then VRF cannot be configured.
- VRF and private VLANs are mutually exclusive. You cannot enable VRF on a private VLAN. Similarly, you cannot enable private VLAN on a VLAN with VRF configured on the VLAN interface.
- VRF and policy-based routing (PBR) are mutually exclusive on a switch interface. You cannot enable VRF when PBR is enabled on an interface. In contrast, you cannot enable PBR when VRF is enabled on an interface.

Configuring VRFs

Follow the steps in this procedure to configure one or more VRFs.

BEFORE YOU BEGIN

See Multi-VRF CE Configuration Guidelines, page 938.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip routing**	Enable IP routing.
3.	**ip vrf** *vrf-name*	Name the VRF, and enter VRF configuration mode.
4.	**rd** *route-distinguisher*	Create a VRF table by specifying a route distinguisher. Enter either an AS number and an arbitrary number (xxx:y) or an IP address and arbitrary number (A.B.C.D:y).
5.	**route-target** {**export** | **import** | **both**} *route-target-ext-community*	Create a list of import, export, or import and export route target communities for the specified VRF. Enter either an AS system number and an arbitrary number (xxx:y) or an IP address and an arbitrary number (A.B.C.D:y). The *route-target-ext-community* should be the same as the *route-distinguisher* entered in Step 4.
6.	**import map** *route-map*	(Optional) Associate a route map with the VRF.
7.	**interface** *interface-id*	Specify the Layer 3 interface to be associated with the VRF, and enter interface configuration mode. The interface can be a routed port or SVI.
8.	**no shutdown**	Enable the interface if necessary. By default, UNIs and ENIs are disabled and NNIs are enabled.
9.	**ip vrf forwarding** *vrf-name*	Associate the VRF with the Layer 3 interface.
10.	**end**	Return to privileged EXEC mode.
11.	**show ip vrf** [**brief** | **detail** | **interfaces**] [*vrf-name*]	Verify the configuration. Display information about the configured VRFs.
12.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip vrf** *vrf-name* global configuration command to delete a VRF and to remove all interfaces from it. Use the **no ip vrf forwarding** interface configuration command to remove an interface from the VRF.

EXAMPLE

The following example shows how to import a route map to a VRF instance named VPN1:

```
Switch(config)# ip vrf vpn1
Switch(config-vrf)# rd 100:2
Switch(config-vrf)# route-target both 100:2
Switch(config-vrf)# route-target import 100:1
```

Configuring VRF-Aware Services

IP services can be configured on global interfaces, and these services run within the global routing instance. IP services are enhanced to run on multiple routing instances; they are VRF-aware. Any configured VRF in the system can be specified for a VRF-aware service.

VRF-aware services are implemented in platform-independent modules. VRF means multiple routing instances in Cisco IOS. Each platform has its own limit on the number of VRFs it supports.

VRF-aware services have the following characteristics:

- The user can ping a host in a user-specified VRF.

- ARP entries are learned in separate VRFs. The user can display Address Resolution Protocol (ARP) entries for specific VRFs.

These services are VRF-aware:

- ARP
- Ping
- Simple Network Management Protocol (SNMP)
- Hot Standby Router Protocol (HSRP)
- Syslog
- Traceroute
- FTP and TFTP

Note: VRF-aware services are not supported for Unicast Reverse Path Forwarding (uRPF).

User Interface for ARP

Use the **arp** command in global configuration mode to add a VRF to the ARP cache.

BEFORE YOU BEGIN

Configure a VRF as described in the Configuring VRFs, page 939.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**arp vrf** *vrf-name hardware-address encap-type* [*interface-type*] [alias]	Add a VRF instance. The ***vrf-name*** argument is the name of the VRF table.
3.	end	Return to privileged EXEC mode.
4.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

```
switch(config)# arp vrf vpn1 0800.0900.1834
```

User Interface for PING

To check if a configured VRF is working, you can use the **ping vrf** command.

When attempting to ping from a provider edge (PE) router to a customer edge (CE) router, or from a PE router to PE router, the standard ping command will not usually work. The **ping vrf** command allows you to ping the IP addresses of LAN interfaces on CE routers.

If you are on a PE router, be sure to indicate the specific VRF (VPN) name, as shown in the “Examples” section.

If all required information is not provided at the command line, the system will enter the interactive dialog (extended mode) for ping.

BEFORE YOU BEGIN

Configure a VRF as described in the Configuring VRFs, page 939.

DETAILED STEPS

	Command	Purpose
1.	**ping vrf** *vrf-name* **ip-host**	Tests a connection in the context of a specific VPN connection.

EXAMPLE

In the following example, the target host in the domain 209.165.201.1 is pinged (using IP/ICMP) in the context of the “CustomerA” VPN connection:

```
Switch# ping vrf CustomerA 209.165.201.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 209.165.201.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 176/264/576 ms
```

User Interface for SNMP

Follow the steps in this procedure to configure configure VRF-aware services for SNMP.

BEFORE YOU BEGIN

Configure a VRF as described in the Configuring VRFs, page 939.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**snmp-server trap authentication vrf**	Enable VRF instance context authentication notifications.
3.	**snmp-server engineID remote** *<host>* **vrf** *<vpn instance> <engine-id string>*	Configure a name for the remote SNMP engine on a switch.
4.	**snmp-server host** *<host>* **vrf** *<vpn instance>* **traps** *<community>*	Specify the recipient of an SNMP trap operation and specify the VRF table to be used for sending SNMP traps.
5.	**snmp-server host <***host>* **vrf** *<vpn instance>* **informs** *<community>*	Specify the recipient of an SNMP inform operation and specify the VRF table to be used for sending SNMP informs.
6.	**snmp-server user** *<user> <group>* **remote** *<host>* **vrf** *<vpn instance> <security model>*	Add a user to an SNMP group for a remote host on a VRF for SNMP access.
7.	**end**	Return to privileged EXEC mode.
8.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

The following example specifies the SNMP engine ID and configures the VRF name traps-vrf for SNMP communications with the remote device at 172.16.20.3:

```
Switch(config)# snmp-server engineID remote 172.16.20.3 vrf trap-vrf 80000009030000B064EFE100
```

The following example shows how to send all SNMP notifications to example.com over the VRF named trap-vrf using the community string public:

```
Switch(config)# snmp-server host example.com vrf trap-vrf public
```

User Interface for HSRP

Hot Standby Router Protocol (HSRP) support for VRFs ensures that HSRP virtual IP addresses are added to the correct IP routing table.

BEFORE YOU BEGIN

Configure a VRF as described in the Configuring VRFs, page 939.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
3.	**no switchport**	Remove the interface from Layer 2 configuration mode if it is a physical interface.
4.	**ip vrf forwarding** <*vrf-name*>	Configure VRF on the interface. Executing this command on an interface removes the IP address.
5.	**ip address** *ip address*	Enter the IP address for the interface.
6.	**standby 1 ip** *ip address*	Enable HSRP and configure the virtual IP address.
7.	**end**	Return to privileged EXEC mode.
8.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

```
Switch(config)# interface ethernet 0
Switch(config-if)# no switchport
Switch(config-if)# ip vrf forwarding vpn1
Switch(config-if)# ip address 172.16.1.3
Switch(config-if)# standby 1 ip
```

User Interface for Syslog

Follow the steps in this procedure to configure VRF-aware services for Syslog.

BEFORE YOU BEGIN

Configure a VRF as described in the Configuring VRFs, page 939.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**logging on**	Enable or temporarily disable logging of storage router event message.
3.	**logging host** ***ip address*** **vrf** ***vrf name***	Specify the host address of the syslog server where logging messages are to be sent.
4.	**logging buffered** *logging buffered size* **debugging**	Log messages to an internal buffer.
5.	**logging trap debugging**	Limit the logging messages sent to the syslog server.
6.	**logging facility** *facility*	Send system logging messages to a logging facility.
7.	**end**	Return to privileged EXEC mode.
8.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

The following example specifies a VRF that connects to the syslog server host:

```
Switch(config)# logging host 192.168.200.225 vrf vpn1
```

User Interface for Traceroute

Follow the steps in this procedure to find the destination address in a VRF.

BEFORE YOU BEGIN

Configure a VRF as described in the Configuring VRFs, page 939.

DETAILED STEPS

Command	Purpose
traceroute vrf *vrf-name ipaddress*	Specify the name of a VPN VRF in which to find the destination address.

EXAMPLE

The following example displays output of the traceroute command with the vrf keyword. Output includes the incoming VRF name/tag and the outgoing VRF name/tag.

```
Switch# traceroute vrf red 10.0.10.12
Type escape sequence to abort.
Tracing the route to 10.0.10.12
VRF info: (vrf in name/id, vrf out name/id)
  1 10.1.13.15 (red/13,red/13) 0 msec
    10.1.16.16 (red/13,red/13) 0 msec
    10.1.13.15 (red/13,red/13) 1 msec
  2 10.1.8.13 (red/13,red/13) 0 msec
    10.1.7.13 (red/13,red/13) 0 msec
    10.1.8.13 (red/13,red/13) 0 msec
  3 10.1.2.11 (red/13,blue/10) 1 msec 0 msec 0 msec
  4  *  *  *
```

User Interface for FTP and TFTP

FTP and TFTP are VRF-aware, which means that file transfer is supported across an interface within a VRF instance. To specify a VRF as a source for FTP or TFTP connections, the VRF must be associated with the same interface that you configure with the **ip ftp source-interface** command. In this configuration, FTP looks for the destination IP address for file transfer in the specified VRF table. If the specified source interface is not up, Cisco IOS software selects the address of the interface closest to the destination as the source address.

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip ftp source-interface** *interface-type interface-number*	Specify the source IP address for FTP connections.
3.	**end**	Return to privileged EXEC mode.
4.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To specify the IP address of an interface as the source address for TFTP connections, use the **ip tftp source-interface** show mode command. To return to the default, use the **no** form of this command.

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip tftp source-interface** *interface-type interface-number*	Specify the source IP address for TFTP connections.
3.	**end**	Return to privileged EXEC mode.
4.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

The following example shows how to configure the switch to use the VRF table named vpn1 to look for the destination IP address for the transfer of FTP packets:

```
Switch# configure terminal
Switch(config)# ip ftp source-interface ethernet 0
Switch(config)# ip vrf vpn1
Switch(config-vrf)# rd 200:1
Switch(config-vrf)# route-target both 200:1
Switch(config-vrf)# interface ethernet 0
Switch(config-if)# ip vrf forwarding vpn1
Switch(config-if)# end
```

User Interface for VRF-Aware RADIUS

To configure VRF-aware RADIUS, you must first enable AAA on a RADIUS server. The switch supports the **ip vrf forwarding** *vrf-name* server-group configuration and the **ip radius source-interface** global configuration commands.

Configuring a VPN Routing Session

Routing within the VPN can be configured with any supported routing protocol (RIP, OSPF, EIGRP, or BGP) or with static routing. The configuration shown here is for OSPF, but the process is the same for other protocols.

Note: To configure an EIGRP routing process to run within a VRF instance, you must configure an autonomous-system number by entering the **autonomous-system** *autonomous-system-number* address-family configuration mode command.

BEFORE YOU BEGIN

Configure a VRF as described in the Configuring VRFs, page 939.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**router ospf** *process-id* **vrf** *vrf-name*	Enable OSPF routing, specify a VPN forwarding table, and enter router configuration mode.
3.	**log-adjacency-changes**	(Optional) Log changes in the adjacency state. This is the default state.
4.	**redistribute bgp** *autonomous-system-number* subnets	Set the switch to redistribute information from the BGP network to the OSPF network.
5.	**network** *network-number* **area** *area-id*	Define a network address and mask on which OSPF runs and the area ID for that network address.
6.	**end**	Return to privileged EXEC mode.
7.	**show ip ospf** *process-id*	Verify the configuration of the OSPF network.
8.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no router ospf** *process-id* **vrf** *vrf-name* global configuration command to disassociate the VPN forwarding table from the OSPF routing process.

EXAMPLE

This example shows a basic OSPF configuration using the **router ospf** command to configure OSPF VRF processes for the VRFs first, second, and third:

```
Switch# configure terminal
Switch(config)# router ospf 12 vrf first
Switch(config)# router ospf 13 vrf second
Switch(config)# router ospf 14 vrf third
Switch(config)# exit
```

Configuring BGP PE to CE Routing Sessions

BEFORE YOU BEGIN

- Complete the BGP network strategy and planning for your network.
- Configure OSPF as described in the Configuring OSPF, page 865.
- Configure a VRF as described in the Configuring VRFs, page 939.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**router bgp** *autonomous-system-number*	Configure the BGP routing process with the AS number passed to other BGP routers, and enter router configuration mode.
3.	**network** *network-number* **mask** *network-mask*	Specify a network and mask to announce using BGP.
4.	**redistribute ospf** *process-id* **match internal**	Set the switch to redistribute OSPF internal routes.
5.	**network** *network-number* **area** *area-id*	Define a network address and mask on which OSPF runs and the area ID for that network address.
6.	**address-family ipv4 vrf** *vrf-name*	Define BGP parameters for PE to CE routing sessions, and enter VRF address-family mode.
7.	**neighbor** *address* **remote-as** *as-number*	Define a BGP session between PE and CE routers.
8.	**neighbor** *address* **activate**	Activate the advertisement of the IPv4 address family.
9.	**end**	Return to privileged EXEC mode.
10.	**show ip bgp** [**ipv4**] [**neighbors**]	Verify BGP configuration.
11.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no router bgp** *autonomous-system-number* global configuration command to delete the BGP routing process. Use the command with keywords to delete routing characteristics.

EXAMPLE

The following example configures BGP for CE to PE routing:

```
Switch(config)# router bgp 800
Switch(config-router)# address-family ipv4 vrf vl2
Switch(config-router-af)# redistribute ospf 2 match internal
Switch(config-router-af)# neighbor 83.0.0.3 remote-as 100
Switch(config-router-af)# neighbor 83.0.0.3 activate
Switch(config-router-af)# network 8.8.2.0 mask 255.255.255.0
Switch(config-router-af)# exit
Switch(config-router)# address-family ipv4 vrf vl1
Switch(config-router-af)# redistribute ospf 1 match internal
Switch(config-router-af)# neighbor 38.0.0.3 remote-as 100
Switch(config-router-af)# neighbor 38.0.0.3 activate
Switch(config-router-af)# network 8.8.1.0 mask 255.255.255.0
Switch(config-router-af)# end
```

Displaying Multi-VRF CE Status

You can use the following privileged EXEC commands to display information about multi-VRF CE configuration and status.

Command	Purpose
show ip protocols vrf *vrf-name*	Display routing protocol information associated with a VRF.
show ip route vrf *vrf-name* [**connected**] [*protocol* [*as-number*]] [**list**] [**mobile**] [**odr**] [**profile**] [**static**] [**summary**] [**supernets-only**]	Display IP routing table information associated with a VRF.
show ip vrf [**brief** | **detail** | **interfaces**] [*vrf-name*]	Display information about the defined VRF instances.

Configuring Protocol-Independent Features

This section describes how to configure IP routing protocol-independent features. For a complete description of the IP routing protocol-independent commands in this chapter, see the *Cisco IOS IP Routing: Protocol-Independent Command Reference*.

This section includes the following topics:

Configuring Cisco Express Forwarding

Cisco Express Forwarding (CEF) is a Layer 3 IP switching technology used to optimize network performance. CEF implements an advanced IP look-up and forwarding algorithm to deliver maximum Layer 3 switching performance. CEF is less CPU-intensive than fast switching route caching, allowing more CPU processing power to be dedicated to packet forwarding. In dynamic networks, fast switching cache entries are frequently invalidated because of routing changes, which can cause traffic to be process switched using the routing table, instead of fast switched using the route cache. CEF uses the Forwarding Information Base (FIB) lookup table to perform destination-based switching of IP packets.

The two main components in CEF are the distributed FIB and the distributed adjacency tables.

- The FIB is similar to a routing table or information base and maintains a mirror image of the forwarding information in the IP routing table. When routing or topology changes occur in the network, the IP routing table is updated, and those changes are reflected in the FIB. The FIB maintains next-hop address information based on the information in the IP routing table. Because the FIB contains all known routes that exist in the routing table, CEF eliminates route cache maintenance, is more efficient for switching traffic, and is not affected by traffic patterns.
- Nodes in the network are said to be adjacent if they can reach each other with a single hop across a link layer. CEF uses adjacency tables to prepend Layer 2 addressing information. The adjacency table maintains Layer 2 next-hop addresses for all FIB entries.

Because the switch uses Application Specific Integrated Circuits (ASICs) to achieve Gigabit-speed line rate IP traffic, CEF forwarding applies only to the software-forwarding path, that is, traffic that is forwarded by the CPU.

CEF is enabled globally by default. If for some reason it is disabled, you can re-enable it by using the **ip cef** global configuration command.

The default configuration is CEF enabled on all Layer 3 interfaces. Entering the **no ip route-cache cef** interface configuration command disables CEF for traffic that is being forwarded by software. This command does not affect the hardware forwarding path. Disabling CEF and using the **debug ip packet detail** privileged EXEC command can be useful to debug software-forwarded traffic. To enable CEF on an interface for the software-forwarding path, use the **ip route-cache cef** interface configuration command.

Caution: Although the no ip route-cache cef interface configuration command to disable CEF on an interface is visible in the CLI, we strongly recommend that you do not disable CEF on interfaces except for debugging purposes.

BEFORE YOU BEGIN

- Cisco Express Forwarding requires a software image that includes Cisco Express Forwarding and IP routing enabled on the switch.
- If you enable Cisco Express Forwarding and then create an access list that uses the log keyword, the packets that match the access list are not Cisco Express Forwarding switched. They are process switched. Logging disables Cisco Express Forwarding.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip cef**	Enable CEF operation.
3.	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
4.	**no shutdown**	Enable the interface if necessary. By default, UNIs and ENIs are disabled and NNIs are enabled.
5.	**ip route-cache cef**	Enable CEF on the interface for software-forwarded traffic.
6.	**end**	Return to privileged EXEC mode.
7.	**show ip cef**	Display the CEF status on all interfaces.
8.	**show cef linecard** [**detail**]	Display CEF-related interface information.
9.	**show cef interface** [*interface-id*]	Display detailed CEF information for all interfaces or the specified interface.
10.	**show adjacency**	Display CEF adjacency table information.
11.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

```
Switch(config)# ip cef
Switch(config)# interface ethernet 0
Switch(config-if)# ip route-cache cef
Switch(config-if)# end
```

Configuring the Number of Equal-Cost Routing Paths

When a router has two or more routes to the same network with the same metrics, these routes can be thought of as having an equal cost. The term *parallel path* is another way to see occurrences of equal-cost routes in a routing table. If a router has two or more equal-cost paths to a network, it can use them concurrently. Parallel paths provide redundancy in case of a circuit failure and also enable a router to load balance packets over the available paths for more efficient use of available bandwidth.

Although the router automatically learns about and configures equal-cost routes, you can control the maximum number of parallel paths supported by an IP routing protocol in its routing table.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**router** {**bgp** | **rip** | **ospf** | **eigrp**}	Enter router configuration mode.
3.	**maximum-paths** *maximum*	Set the maximum number of parallel paths for the protocol routing table. The range is from 1 to 8; the default is 4 for most IP routing protocols, but only 1 for BGP.
4.	**end**	Return to privileged EXEC mode.
5.	**show ip protocols**	Verify the setting in the *Maximum path* field.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no maximum-paths** router configuration command to restore the default value.

EXAMPLE

The following example shows how to allow a maximum of 16 paths to a destination in an OSPF routing process:

```
Switch(config)# router ospf 3
Switch(config-router)# maximum-paths 16
```

Configuring Static Unicast Routes

Static unicast routes are user-defined routes that cause packets moving between a source and a destination to take a specified path. Static routes can be important if the router cannot build a route to a particular destination and are useful for specifying a gateway of last resort to which all unroutable packets are sent.

The switch retains static routes until you remove them. However, you can override static routes with dynamic routing information by assigning administrative distance values. Each dynamic routing protocol has a default administrative distance, as listed in Table 2. If you want a static route to be overridden by information from a dynamic routing protocol, set the administrative distance of the static route higher than that of the dynamic protocol.

Table 66 Default Administrative Distance Values

Route Source	Default Distance
Connected interface	0
Static route	1
Enhanced IRGP summary route	5
External BGP	20
Internal Enhanced IGRP	90
IGRP	100
OSPF	110
Internal BGP	200
Unknown	225

Static routes that point to an interface are advertised through RIP, IGRP, and other dynamic routing protocols, whether or not static **redistribute** router configuration commands were specified for those routing protocols. These static routes are advertised because static routes that point to an interface are considered in the routing table to be connected and hence lose their static nature. However, if you define a static route to an interface that is not one of the networks defined in a network command, no dynamic routing protocols advertise the route unless a **redistribute** static command is specified for these protocols.

When an interface goes down, all static routes through that interface are removed from the IP routing table. When the software can no longer find a valid next hop for the address specified as the forwarding router's address in a static route, the static route is also removed from the IP routing table.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip route** *prefix mask* {*address* | *interface*} [*distance*]	Establish a static route.
3.	**end**	Return to privileged EXEC mode.
4.	**show ip route**	Display the current state of the routing table to verify the configuration.
5.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip route** *prefix mask* {*address* | *interface*} global configuration command to remove a static route.

EXAMPLE

The following example shows how to choose an administrative distance of 110. In this case, packets for network 10.0.0.0 will be routed to a router at 172.31.3.4 if dynamic information with an administrative distance less than 110 is not available.

```
ip route 10.0.0.0 255.0.0.0 172.31.3.4 110
```

Specifying Default Routes and Networks

A router might not be able to learn the routes to all other networks. To provide complete routing capability, you can use some routers as smart routers and give the remaining routers default routes to the smart router. (Smart routers have routing table information for the entire internetwork.) These default routes can be dynamically learned or can be configured in the individual routers. Most dynamic interior routing protocols include a mechanism for causing a smart router to generate dynamic default information that is then forwarded to other routers.

If a router has a directly connected interface to the specified default network, the dynamic routing protocols running on that device generate a default route. In RIP, it advertises the pseudonetwork 0.0.0.0.s

A router that is generating the default for a network also might need a default of its own. One way a router can generate its own default is to specify a static route to the network 0.0.0.0 through the appropriate device.

When default information is passed through a dynamic routing protocol, no further configuration is required. The system periodically scans its routing table to choose the optimal default network as its default route. In IGRP networks, there might be several candidate networks for the system default. Cisco routers use administrative distance and metric information to set the default route or the gateway of last resort.

If dynamic default information is not being passed to the system, candidates for the default route are specified with the **ip default-network** global configuration command. If this network appears in the routing table from any source, it is flagged as a possible choice for the default route. If the router has no interface on the default network, but does have a path to it, the network is considered as a possible candidate, and the gateway to the best default path becomes the gateway of last resort.

BEFORE YOU BEGIN

The **ip default-network** command is a classful command. It is effective only if the network mask of the network that you wish to configure as a candidate route for computing the gateway of last resort matches the network mask in the Routing Information Base (RIB).

For example, if you configure **ip default-network 10.0.0.0**, then the mask considered by the routing protocol is 10.0.0.0/8, as it is a Class A network. The gateway of last resort is set only if the RIB contains a 10.0.0.0/8 route.

If you need to use the **ip default-network** command, ensure that the RIB contains a network route that matches the major mask of the network class.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ip default-network** *network number*	Specify a default network.
3.	**end**	Return to privileged EXEC mode.
4.	**show ip route**	Display the selected default route in the gateway of last resort display.
5.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ip default-network** *network number* global configuration command to remove the route.

EXAMPLE

The following example defines a static route to network 10.0.0.0 as the static default route:

```
ip route 10.0.0.0 255.0.0.0 10.108.3.4
ip default-network 10.0.0.0
```

Using Route Maps to Redistribute Routing Information

The switch can run multiple routing protocols simultaneously, and it can redistribute information from one routing protocol to another. Redistributing information from one routing protocol to another applies to all supported IP-based routing protocols.

You can also conditionally control the redistribution of routes between routing domains by defining enhanced packet filters or route maps between the two domains. The **match** and **set** route-map configuration commands define the condition portion of a route map. The **match** command specifies that a criterion must be matched. The **set** command specifies an action to be taken if the routing update meets the conditions defined by the match command. Although redistribution is a protocol-independent feature, some of the **match** and **set** route-map configuration commands are specific to a particular protocol.

One or more **match** commands and one or more **set** commands follow a **route-map** command. If there are no **match** commands, everything matches. If there are no **set** commands, nothing is done, other than the match. Therefore, you need at least one **match** or **set** command.

Note: A route map with no **set** route-map configuration commands is sent to the CPU, which causes high CPU utilization.

You can also identify route-map statements as **permit** or **deny**. If the statement is marked as a deny, the packets meeting the match criteria are sent back through the normal forwarding channels (destination-based routing). If the statement is marked as permit, set clauses are applied to packets meeting the match criteria. Packets that do not meet the match criteria are forwarded through the normal routing channel.

You can use the BGP route map **continue** clause to execute additional entries in a route map after an entry is executed with successful match and set clauses. You can use the **continue** clause to configure and organize more modular policy definitions so that specific policy configurations need not be repeated within the same route map. The switch supports the **continue** clause for outbound policies. For more information about using the route map **continue** clause, see the “BGP Route-Map Continue” section in the *IP Routing: BGP Configuration Guide, Cisco IOS Release 15M&T*.

Note: Although each of Steps 3 through 14 in the following section is optional, you must enter at least one **match** route-map configuration command and one **set** route-map configuration command.

BEFORE YOU BEGIN

You should know your network design and how you want traffic to flow through it before configuring route redistribution or policy-based routing.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**route-map** *map-tag* [**permit | deny**] [*sequence number*]	Define any route maps used to control redistribution and enter route-map configuration mode. ■ *map-tag*–A meaningful name for the route map. The **redistribute** router configuration command uses this name to reference this route map. Multiple route maps might share the same map tag name. ■ (Optional) If **permit** is specified and the match criteria are met for this route map, the route is redistributed as controlled by the set actions. If **deny** is specified, the route is not redistributed. ■ *sequence number* (Optional)– Number that indicates the position a new route map is to have in the list of route maps already configured with the same name.
3.	**match as-path** *path-list-number*	Match a BGP AS path access list.
4.	**match community-list** *community-list-number* [**exact**]	Match a BGP community list.
5.	**match ip address** {*access-list-number* | *access-list-name*} [*...access-list-number* | *...access-list-name*]	Match a standard access list by specifying the name or number. It can be an integer from 1 to 199.
6.	**match metric** *metric-value*	Match the specified route metric. The *metric-value* can be an EIGRP metric with a specified value from 0 to 4294967295.
7.	**match ip next-hop** {*access-list-number* | *access-list-name*} [*...access-list-number* | *...access-list-name*]	Match a next-hop router address passed by one of the access lists specified (numbered from 1 to 199).
8.	**match tag** *tag value* [*...tag-value*]	Match the specified tag value in a list of one or more route tag values. Each can be an integer from 0 to 4294967295.
9.	**match interface** *type number* [*...type number*]	Match the specified next hop route out one of the specified interfaces.
10.	**match ip route-source** {*access-list-number | access-list-name*} [*...access-list-number | ...access-list-name*]	Match the address specified by the specified advertised access lists.
11.	**match route-type** {**local** | **internal** | **external** [**type-1** | **type-2**]}	Match the specified **route-type**: ■ **local**–Locally generated BGP routes. ■ **internal**–OSPF intra-area and interarea routes or EIGRP internal routes. ■ **external**–OSPF external routes (Type 1 or Type 2) or EIGRP external routes.

	Command	Purpose
12.	**set dampening** *halflife reuse suppress max-suppress-time*	Set BGP route dampening factors.
13.	**set local-preference** *value*	Assign a value to a local BGP path.
14.	**set origin** {**igp** | **egp** *as* | **incomplete**}	Set the BGP origin code.
15.	**set as-path** {**tag** | **prepend** *as-path-string*}	Modify the BGP autonomous system path.
16.	**set level** {**level-1** / **level-2** / **level-1-2** / **stub-area** / **backbone**}	Set the level for routes that are advertised into the specified area of the routing domain. The **stub-area** and **backbone** are OSPF NSSA and backbone areas.
17.	**set metric** *metric value*	Set the metric value to give the redistributed routes (for EIGRP only). The *metric value* is an integer from -294967295 to 294967295.
18.	**set metric** *bandwidth delay reliability loading mtu*	Set the metric value to give the redistributed routes (for EIGRP only): ■ *bandwidth*—Metric value or IGRP bandwidth of the route in kilobits per second in the range 0 to 4294967295 ■ *delay*—Route delay in tens of microseconds in the range 0 to 4294967295. ■ *reliability*—Likelihood of successful packet transmission expressed as a number between 0 and 255, where 255 means 100 percent reliability and 0 means no reliability. ■ *loading*— Effective bandwidth of the route expressed as a number from 0 to 255 (255 is 100 percent loading). ■ *mtu*—Minimum maximum transmission unit (MTU) size of the route in bytes in the range 0 to 4294967295.
19.	**set metric-type** {**type-1** | **type-2**}	Set the OSPF external metric type for redistributed routes.
20.	**set metric-type internal**	Set the multi-exit discriminator (MED) value on prefixes advertised to external BGP neighbor to match the IGP metric of the next hop.
21.	**set weight**	Set the BGP weight for the routing table. The value can be from 1 to 65535.
22.	**end**	Return to privileged EXEC mode.
23.	**show route-map**	Display all route maps configured or only the one specified to verify configuration.
24.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete an entry, use the **no route-map** *map tag* global configuration command or the **no match** or **no set** route-map configuration commands.

EXAMPLE

The following example shows how to redistribute Routing Information Protocol (RIP) routes with a hop count equal to 1 to Open Shortest Path First (OSPF). These routes will be redistributed to OSPF as external link-state advertisements (LSAs) with a metric of 5, metric type of Type 1, and a tag equal to 1.

```
Switch(config)# router ospf 109
Switch(config-router)# redistribute rip route-map rip-to-ospf
Switch(config-router)# exit
Switch(config)# route-map rip-to-ospf permit
Switch(config-route-map)# match metric 1
Switch(config-route-map)# set metric 5
Switch(config-route-map)# set metric-type type1
Switch(config-route-map)# set tag 1
```

Controlling Route Redistribution

You can distribute routes from one routing domain into another and control route distribution. Note that the keywords in this procedure are the same as defined in the previous procedure.

The metrics of one routing protocol do not necessarily translate into the metrics of another. In these situations, an artificial metric is assigned to the redistributed route. Uncontrolled exchanging of routing information between different routing protocols can create routing loops and seriously degrade network operation.

If you have not defined a default redistribution metric that replaces metric conversion, some automatic metric translations occur between routing protocols:

- RIP can automatically redistribute static routes. It assigns static routes a metric of 1 (directly connected).
- Any protocol can redistribute other routing protocols if a default mode is in effect.

BEFORE YOU BEGIN

Review the usage guidelines and additional examples for the **redistribute** command in the *Cisco IOS IP Routing: Protocol-Independent Command Reference*.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**router** {**bgp** | **rip** | **ospf** | **eigrp**}	Enter router configuration mode.
3.	**redistribute** *protocol* [*process-id*] {**level-1** | **level-1-2** | **level-2**} [**metric** *metric-value*] [**metric-type** *type-value*] [**match internal** | **external** *type-value*] [**tag** *tag-value*] [**route-map** *map-tag*] [**weight** *weight*] [**subnets**]	Redistribute routes from one routing protocol to another routing protocol. If no route-maps are specified, all routes are redistributed. If the keyword **route-map** is specified with no *map-tag*, no routes are distributed.
4.	**default-metric** *number*	Cause the current routing protocol to use the same metric value for all redistributed routes (BGP, RIP, and OSPF).
5.	**default-metric** *bandwidth delay reliability loading mtu*	Cause the EIGRP routing protocol to use the same metric value for all non-EIGRP redistributed routes.
6.	**end**	Return to privileged EXEC mode.
7.	**show route-map**	Display all route maps configured or only the one specified to verify configuration.
8.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable redistribution, use the **no** form of the commands.

EXAMPLE

Given the following configuration, a RIP-learned route for network 160.89.0.0 and an ISO IGRP-learned route with prefix 49.0001.0002 will be redistributed into an IS-IS Level 2 link-state PDU with metric 5:

```
router isis
redistribute rip route-map ourmap
redistribute iso-igrp remote route-map ourmap
route-map ourmap permit
match ip address 1
match clns address ourprefix
set metric 5
set level level-2
access-list 1 permit 160.89.0.0 0.0.255.255
clns filter-set ourprefix permit 49.0001.0002...
```

Configuring Policy-Based Routing

You can use policy-based routing (PBR) to configure a defined policy for traffic flows. By using PBR, you can have more control over routing by reducing the reliance on routes derived from routing protocols. PBR can specify and implement routing policies that allow or deny paths based on:

- Identity of a particular end system
- Application
- Protocol

You can use PBR to provide equal-access and source-sensitive routing, routing based on interactive versus batch traffic, or routing based on dedicated links. For example, you could transfer stock records to a corporate office on a high-bandwidth, high-cost link for a short time while transmitting routine application data such as e-mail over a low-bandwidth, low-cost link.

With PBR, you classify traffic using access control lists (ACLs) and then make traffic go through a different path. PBR is applied to incoming packets. All packets received on an interface with PBR enabled are passed through route maps. Based on the criteria defined in the route maps, packets are forwarded (routed) to the appropriate next hop.

- If packets do not match any route map statements, all set clauses are applied.
- If a statement is marked as permit and the packets do not match any route-map statements, the packets are sent through the normal forwarding channels, and destination-based routing is performed.
- For PBR, route-map statements marked as deny are not supported.

For more information about configuring route maps, see Using Route Maps to Redistribute Routing Information, page 953.

You can use standard IP ACLs to specify match criteria for a source address or extended IP ACLs to specify match criteria based on an application, a protocol type, or an end station. The process proceeds through the route map until a match is found. If no match is found, normal destination-based routing occurs. There is an implicit deny at the end of the list of match statements.

If match clauses are satisfied, you can use a set clause to specify the IP addresses identifying the next hop router in the path.

For details about PBR commands and keywords, see *IP Routing: Protocol-Independent Configuration Guide, Cisco IOS Release 15M&T*.

PBR Configuration Guidelines

Before configuring PBR, you should be aware of this information:

- Multicast traffic is not policy-routed. PBR applies to only to unicast traffic.
- You can enable PBR on a routed port or an SVI.
- The switch does not support **route-map deny** statements for PBR.
- You can apply a policy route map to an EtherChannel port channel in Layer 3 mode, but you cannot apply a policy route map to a physical interface that is a member of the EtherChannel. If you try to do so, the command is rejected. When a policy route map is applied to a physical interface, that interface cannot become a member of an EtherChannel.
- You can define a maximum of 246 IP policy route maps on the switch.
- You can define a maximum of 512 access control entries (ACEs) for PBR on the switch.
- When configuring match criteria in a route map, follow these guidelines:
  - Do not match ACLs that permit packets destined for a local address. PBR would forward these packets, which could cause ping or Telnet failure or route protocol flapping.
  - Do not match ACLs with deny ACEs. Packets that match a deny ACE are sent to the CPU, which could cause high CPU utilization.
- To use PBR, you must first enable the default template by using the **sdm prefer default** global configuration command. PBR is not supported with the Layer 2 template.
- VRF and PBR are mutually-exclusive on a switch interface. You cannot enable VRF when PBR is enabled on an interface. In contrast, you cannot enable PBR when VRF is enabled on an interface.

- The number of TCAM entries used by PBR depends on the route map itself, the ACLs used, and the order of the ACLs and route-map entries.
- Policy-based routing based on packet length, IP precedence and TOS, set interface, set default next hop, or set default interface are not supported. Policy maps with no valid set actions or with set action set to *Don't Fragment* are not supported.

Enabling PBR

By default, PBR is disabled on the switch. To enable PBR, you must create a route map that specifies the match criteria and the resulting action if all of the match clauses are met. Then, you must enable PBR for that route map on an interface. All packets arriving on the specified interface matching the match clauses are subject to PBR.

PBR can be fast-switched or implemented at speeds that do not slow down the switch. Fast-switched PBR supports most match and set commands. PBR must be enabled before you enable fast-switched PBR. Fast-switched PBR is disabled by default.

Packets that are generated by the switch, or local packets, are not normally policy-routed. When you globally enable local PBR on the switch, all packets that originate on the switch are subject to local PBR. Local PBR is disabled by default.

BEFORE YOU BEGIN

See PBR Configuration Guidelines, page 958.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**route-map** *map-tag* [**permit**] [*sequence number*]	Define any route maps used to control where packets are output, and enter route-map configuration mode. ■ *map-tag*–A meaningful name for the route map. The **ip policy route-map** interface configuration command uses this name to reference the route map. Multiple route maps might share the same map tag name. ■ (Optional) If **permit** is specified and the match criteria are met for this route map, the route is policy-routed as controlled by the set actions. **Note:** The **route-map deny** statement is not supported in PBR route maps to be applied to an interface. ■ *sequence number* (Optional)– Number that shows the position of a new route map in the list of route maps already configured with the same name.
3.	**match ip address** {*access-list-number* | *access-list-name*} [*...access-list-number* | *...access-list-name*]	Match the source and destination IP address that is permitted by one or more standard or extended access lists. **Note:** Do not enter an ACL with a deny ACE or an ACL that permits a packet destined for a local address. If you do not specify a **match** command, the route map applies to all packets.
4.	**set ip next-hop** *ip-address* [*...ip-address*]	Specify the action to take on the packets that match the criteria. Set next hop to which to route the packet (the next hop must be adjacent).
5.	**exit**	Return to global configuration mode.
6.	**interface** *interface-id*	Enter interface configuration mode, and specify the interface to configure.
7.	**no shutdown**	Enable the interface if necessary. By default, UNIs and ENIs are disabled and NNIs are enabled.
8.	**ip policy route-map** *map-tag*	Enable PBR on a Layer 3 interface, and identify the route map to use. You can configure only one route map on an interface. However, you can have multiple route map entries with different sequence numbers. These entries are evaluated in sequence number order until the first match. If there is no match, packets are routed as usual. **Note:** If the IP policy route map contains a **deny** statement, the configuration fails.
9.	**ip route-cache policy**	(Optional) Enable fast-switching PBR. You must first enable PBR before enabling fast-switching PBR.
10.	**exit**	Return to global configuration mode.
11.	**ip local policy route-map** *map-tag*	(Optional) Enable local PBR to perform policy-based routing on packets originating at the switch. This applies to packets generated by the switch and not to incoming packets.

	Command	Purpose
12.	**end**	Return to privileged EXEC mode.
13.	**show route-map** [*map-name*]	(Optional) Display all route maps configured or only the one specified to verify configuration.
14.	**show ip policy**	(Optional) Display policy route maps attached to interfaces.
15.	**show ip local policy**	(Optional) Display whether or not local policy routing is enabled and, if so, the route map being used.
16.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no route-map** *map-tag* global configuration command or the **no match** or **no set** route-map configuration commands to delete an entry. Use the **no ip policy route-map** *map-tag* interface configuration command to disable PBR on an interface. Use the **no ip route-cache policy** interface configuration command to disable fast-switching PBR. Use the no **ip local policy route-map** *map-tag* global configuration command to disable policy-based routing on packets originating on the switch.

EXAMPLE

The following example sends packets with the destination IP address of 172.21.16.18 to a router at IP address 172.30.3.20:

```
interface serial 0
 ip policy route-map wethersfield
!
route-map wethersfield
 match ip address 172.21.16.18
 set ip next-hop 172.30.3.20
```

Filtering Routing Information

You can filter routing protocol information by performing the tasks described in this section.

Note: When routes are redistributed between OSPF processes, no OSPF metrics are preserved.

Setting Passive Interfaces

To prevent other routers on a local network from dynamically learning about routes, you can use the **passive-interface** router configuration command to keep routing update messages from being sent through a router interface. When you use this command in the OSPF protocol, the interface address you specify as passive appears as a stub network in the OSPF domain. OSPF routing information is neither sent nor received through the specified router interface.

In networks with many interfaces, to avoid having to manually set them as passive, you can set all interfaces to be passive by default by using the **passive-interface default** router configuration command and manually setting interfaces where adjacencies are desired.

BEFORE YOU BEGIN

You should know your network design and how you want traffic to flow through it before filtering routing information.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**router** {**bgp** | **rip** | **ospf** | **eigrp**}	Enter router configuration mode.
3.	**passive-interface** *interface-id*	Suppress sending routing updates through the specified Layer 3 interface.
4.	**passive-interface default**	(Optional) Set all interfaces as passive by default.
5.	**no passive-interface** *interface type*	(Optional) Activate only those interfaces that need to have adjacencies sent.
6.	**network** *network-address*	(Optional) Specify the list of networks for the routing process. The *network-address* is an IP address.
7.	**end**	Return to privileged EXEC mode.
8.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use a network monitoring privileged EXEC command such as **show ip ospf interface** to verify the interfaces that you enabled as passive, or use the **show ip interface** privileged EXEC command to verify the interfaces that you enabled as active.

To re-enable the sending of routing updates, use the **no passive-interface** *interface-id* router configuration command.

EXAMPLE

The following example sends EIGRP updates to all interfaces on network 10.108.0.0 except Ethernet interface 1:

```
router eigrp 109
 network 10.108.0.0
 passive-interface ethernet 1
```

The following example sets all interfaces as passive and then activates Ethernet interface 0:

```
router ospf 100
 passive-interface default
 no passive-interface ethernet0
 network 10.108.0.1 0.0.0.255 area 0
```

Controlling Advertising and Processing in Routing Updates

You can use the **distribute-list** router configuration command with access control lists to suppress routes from being advertised in routing updates and to prevent other routers from learning one or more routes. When used in OSPF, this feature applies to only external routes, and you cannot specify an interface name.

You can also use a **distribute-list** router configuration command to avoid processing certain routes listed in incoming updates. (This feature does not apply to OSPF.)

BEFORE YOU BEGIN

Configure an access list **defining which networks are to be sent or received and which are to be suppressed in routing updates.**

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**router** {**bgp** | **rip** | **eigrp**}	Enter router configuration mode.
3.	**distribute-list** {*access-list-number* | *access-list-name*} **out** [*interface-name* | *routing process* | *autonomous-system-number*]	Permit or deny routes from being advertised in routing updates, depending upon the action listed in the access list.
4.	**distribute-list** {*access-list-number* | *access-list-name*} **in** [*type-number*]	Suppress processing in routes listed in updates.
5.	**end**	Return to privileged EXEC mode.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no distribute-list in** router configuration command to change or cancel a filter. To cancel suppression of network advertisements in updates, use the **no distribute-list out** router configuration command.

EXAMPLE

In the following example, a prefix list and distribute list are defined to configure the BGP routing process to accept traffic from only network 10.1.1.0/24, network 192.168.1.0, and network 10.108.0.0. An inbound route refresh is initiated to activate the distribute-list.

```
Switch(config)# ip prefix-list RED permit 10.1.1.0/24
Switch(config)# ip prefix-list RED permit 10.108.0.0/16
Switch(config)# ip prefix-list RED permit 192.168.1.0/24
Switch(config)# router bgp 50000
Switch(config-router)# network 10.108.0.0
Switch(config-router)# distribute-list prefix RED in
Switch(config-router)# end
Switch# clear ip bgp in
```

Filtering Sources of Routing Information

Because some routing information might be more accurate than others, you can use filtering to prioritize information coming from different sources. An *administrative distance* is a rating of the trustworthiness of a routing information source, such as a router or group of routers. In a large network, some routing protocols can be more reliable than others. By specifying administrative distance values, you enable the router to intelligently discriminate between sources of routing information. The router always picks the route whose routing protocol has the lowest administrative distance.

Because each network has its own requirements, there are no general guidelines for assigning administrative distances.

BEFORE YOU BEGIN

- Always set the administrative distance from the least to the most specific network.
- Review the usage guidelines and additional examples for the **distance** command in the *Cisco IOS IP Routing: Protocol-Independent Command Reference*.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**router** {**bgp** | **rip** | **ospf** | **eigrp**}	Enter router configuration mode.
3.	**distance** *weight* {*ip-address* {*ip-address mask*}} [*ip access list*]	Define an administrative distance. ■ *weight*–The administrative distance as an integer from 10 to 255. Used alone, *weight* specifies a default administrative distance that is used when no other specification exists for a routing information source. Routes with a distance of 255 are not installed in the routing table. ■ (Optional) *ip access list*–An IP standard or extended access list to be applied to incoming routing updates.
4.	**end**	Return to privileged EXEC mode.
5.	**show ip protocols**	Display the default administrative distance for a specified routing process.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove a distance definition, use the **no distance** router configuration command.

EXAMPLE

In the following example, the **routereigrp** global configuration command sets up EIGRP routing in autonomous system number 109. The network router configuration commands specify EIGRP routing on networks 192.168.7.0 and 172.16.0.0. The first distance command sets the administrative distance to 90 for all routers on the Class C network 192.168.7.0. The second distance command sets the administrative distance to 120 for the router with the address 172.16.1.3.

```
Switch# configure terminal
Switch(config)# router eigrp 109
Switch(config-router)# network 192.168.7.0
Switch(config-router)# network 172.16.0.0
Switch(config-router)# distance 90 192.168.7.0 0.0.0.255
Switch(config-router)# distance 120 172.16.1.3 0.0.0.255
Switch(config-router)# end
```

In the following example, the set distance is from the least to the most specific network:

```
Switch# configure terminal
Switch(config)# router eigrp 109
Switch(config-router)# distance 22 10.0.0.0 0.0.0.255
Switch(config-router)# distance 33 10.11.0.0 0.0.0.255
Switch(config-router)# distance 44 10.11.12.0 0.0.0.255
Switch(config-router)# end
```

Managing Authentication Keys

Key management is a method of controlling authentication keys used by routing protocols. Not all protocols can use key management. Authentication keys are available for EIGRP and RIP Version 2.

To manage authentication keys, define a key chain, identify the keys that belong to the key chain, and specify how long each key is valid. Each key has its own key identifier (specified with the **key** *number* key chain configuration command), which is stored locally. The combination of the key identifier and the interface associated with the message uniquely identifies the authentication algorithm and Message Digest 5 (MD5) authentication key in use.

You can configure multiple keys with life times. Only one authentication packet is sent, regardless of how many valid keys exist. The software examines the key numbers in order from lowest to highest, and uses the first valid key it encounters. The lifetimes allow for overlap during key changes. Note that the router must know these lifetimes.

BEFORE YOU BEGIN

Before you manage authentication keys, you must enable authentication. See the appropriate protocol section to see how to enable authentication for that protocol.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**key chain** *name-of-chain*	Identify a key chain, and enter key chain configuration mode.
3.	**key** *number*	Identify the key number. The range is 0 to 2147483647.
4.	**key-string** *text*	Identify the key string. The string can contain from 1 to 80 uppercase and lowercase alphanumeric characters, but the first character cannot be a number.
5.	**accept-lifetime** *start-time* {**infinite** | *end-time* | **duration** *seconds*}	(Optional) Specify the time period during which the key can be received. The *start-time* and *end-time* syntax can be either *hh*:*mm*:*ss Month date year* or *hh*:*mm*:*ss date Month year*. The default is forever with the default *start-time* and the earliest acceptable date as January 1, 1993. The default *end-time* and **duration** is **infinite**.
6.	**send-lifetime** *start-time* {**infinite** | *end-time* | **duration** *seconds*}	(Optional) Specify the time period during which the key can be sent. The *start-time* and *end-time* syntax can be either *hh*:*mm*:*ss Month date year* or *hh*:*mm*:*ss date Month year*. The default is forever with the default *start-time* and the earliest acceptable date as January 1, 1993. The default *end-time* and **duration** is infinite.
7.	**end**	Return to privileged EXEC mode.
8.	**show key chain**	Display authentication key information.
9.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the key chain, use the **no key chain** *name-of-chain* global configuration command.

EXAMPLE

The following example configures a key chain named chain1. The key named key1 will be accepted from 1:30 p.m. to 3:30 p.m. and be sent from 2:00 p.m. to 3:00 p.m. The key named key2 will be accepted from 2:30 p.m. to 4:30 p.m. and be sent from 3:00 p.m. to 4:00 p.m. The overlap allows for migration of keys or a discrepancy in the set time of the router. There is a 30-minute leeway on each side to handle time differences.

```
Router(config)# interface ethernet 0
Router(config-if)# ip rip authentication key-chain chain1
Router(config-if)# ip rip authentication mode md5
!
Router(config)# router rip
Router(config-router)# network 172.19.0.0
Router(config-router)# version 2
!
Router(config)# key chain chain1
Router(config-keychain)# key 1
Router(config-keychain-key)# key-string key1
Router(config-keychain-key)# accept-lifetime 13:30:00 Jan 25 1996 duration 7200
Router(config-keychain-key)# send-lifetime 14:00:00 Jan 25 1996 duration 3600
Router(config-keychain-key)# exit
Router(config-keychain)# key 2
Router(config-keychain-key)# key-string key2
Router(config-keychain-key)# accept-lifetime 14:30:00 Jan 25 1996 duration 7200
Router(config-keychain-key)# send-lifetime 15:00:00 Jan 25 1996 duration 3600
```

Verifying Configuration

You can remove all contents of a particular cache, table, or database. You can also display specific statistics.

Command	Purpose
clear ip route {*network* [*mask* **|** *****]}	Clear one or more routes from the IP routing table.
show ip protocols	Display the parameters and state of the active routing protocol process.
show ip route [*address* [*mask*] [**longer-prefixes**]] | [*protocol* [*process-id*]]	Display the current state of the routing table.
show ip route summary	Display the current state of the routing table in summary form.
show ip route supernets-only	Display supernets.
show ip cache	Display the routing table used to switch IP traffic.
show route-map [*map-name*]	Display all route maps configured or only the one specified.

Related Documents

- Cisco IOS Master Command List, All Releases
- IP Addressing: ARP Configuration Guide, Cisco IOS Release 15M&T
- Cisco IOS IP Routing: RIP Command Reference
- IP Routing: RIP Configuration Guide, Cisco IOS Release 15M&T
- Cisco IOS IP Routing: OSPF Command Reference
- IP Routing: OSPF Configuration Guide, Cisco IOS Release 15M&T

- Cisco IOS IP Routing: EIGRP Command Reference
- IP Routing: EIGRP Configuration Guide, Cisco IOS Release 15M&T
- Cisco IOS IP Routing: BGP Command Reference
- IP Routing: BGP Configuration Guide, Cisco IOS Release 15M&T
- Cisco IOS ISO CLNS Command Reference
- ISO CLNS Configuration Guide, Cisco IOS Release 15M&T
- Cisco IOS IP Routing: ISIS Command Reference
- IP Routing: ISIS Configuration Guide, Cisco IOS Release 15M&T
- High Availability Configuration Guide, Cisco IOS Release 15S
- IP Routing: BFD Configuration Guide, Cisco IOS Release 15M&T
- Cisco IOS IP Routing: Protocol-Independent Command Reference
- IP Routing: Protocol-Independent Configuration Guide, Cisco IOS Release 15M&T
- *Internet Routing Architectures,* published by Cisco Press

Configuring IPv6 Unicast Routing

This chapter describes how to configure IPv6 unicast routing on the Cisco Industrial Ethernet Switches, hereafter referred to as “switch.”

To use this feature, the switch must be running the IP services image. To enable IPv6 routing, you must configure the switch to use a dual IPv4 and IPv6 switch database management (SDM) template. See Dual IPv4 and IPv6 Protocol Stacks, page 972.

Note: For complete syntax and usage information for the commands used in this chapter, see the Cisco IOS documentation listed in the Related Documents, page 998.

Information About IPv6

IPv4 users can move to IPv6 and receive services such as end-to-end security, quality of service (QoS), and globally unique addresses. The IPv6 address space reduces the need for private addresses and Network Address Translation (NAT) processing by border routers at network edges.

This section describes IPv6 implementation on the switch and includes the following topics:

IPv6 Addresses

The switch supports only IPv6 unicast addresses. It does not support site-local unicast addresses, anycast addresses, or multicast addresses.

The IPv6 128-bit addresses are represented as a series of eight 16-bit hexadecimal fields separated by colons in the format: n:n:n:n:n:n:n:n. This is an example of an IPv6 address:

2031:0000:130F:0000:0000:09C0:080F:130B

For easier implementation, leading zeros in each field are optional. This is the same address without leading zeros:

2031:0:130F:0:0:9C0:80F:130B

You can also use two colons (::) to represent successive hexadecimal fields of zeros, but you can use this short version only once in each address:

2031:0:130F::09C0:080F:130B

For more information about IPv6 address formats, address types, and the IPv6 packet header, see *IPv6 Addressing and Basic Connectivity Configuration Guide, Cisco IOS Release 15M&T* in the IPv6 Configuration Library, Cisco IOS Release 15M&T.

In the “Information About Implementing Basic Connectivity for IPv6” chapter, these sections apply to the switch:

- IPv6 Address Formats
- IPv6 Address Type: Unicast
- IPv6 Address Output Display
- Simplified IPv6 Packet Header

Supported IPv6 Unicast Routing Features

Support on the switch includes expanded address capability, header format simplification, improved support of extensions and options, and hardware parsing of the extension header. The switch supports hop-by-hop extension header packets, which are routed or bridged in software.

The switch provides IPv6 routing capability over 802.1Q trunk ports for static routes, Routing Information Protocol (RIP) for IPv6, and Open Shortest Path First (OSPF) Version 3 Protocol. It supports up to 16 equal-cost routes and can simultaneously forward IPv4 and IPv6 frames at line rate.

Note: For more information about the IPv6 unicast routing features described in this section, see IPv6 Configuration Library, Cisco IOS Release 15M&T and *IPv6 Implementation Guide, Cisco IOS Release 15.2M&T*.

- EIGRP IPv6, page 973
- Multiprotocol BGP for IPv6, page 973
- SNMP and Syslog Over IPv6, page 974
- HTTP(S) Over IPv6, page 974

128-Bit Unicast Addresses

The switch supports aggregatable global unicast addresses and link-local unicast addresses. It does not support site-local unicast addresses.

- Aggregatable global unicast addresses are IPv6 addresses from the aggregatable global unicast prefix. The address structure enables strict aggregation of routing prefixes and limits the number of routing table entries in the global routing table. These addresses are used on links that are aggregated through organizations and eventually to the Internet service provider.

  These addresses are defined by a global routing prefix, a subnet ID, and an interface ID. Current global unicast address allocation uses the range of addresses that start with binary value 001 (2000::/3). Addresses with a prefix of 2000::/3(001) through E000::/3(111) must have 64-bit interface identifiers in the extended unique identifier (EUI)-64 format.

- Link local unicast addresses can be automatically configured on any interface by using the link-local prefix FE80::/10(1111 1110 10) and the interface identifier in the modified EUI format. Link-local addresses are used in the neighbor discovery protocol (NDP) and the stateless autoconfiguration process. Nodes on a local link use link-local addresses and do not require globally unique addresses to communicate. IPv6 routers do not forward packets with link-local source or destination addresses to other links.

DNS for IPv6

IPv6 supports Domain Name System (DNS) record types in the DNS name-to-address and address-to-name lookup processes. The DNS AAAA resource record types support IPv6 addresses and are equivalent to an A address record in IPv4. The switch supports DNS resolution for IPv4 and IPv6.

Path MTU Discovery for IPv6 Unicast

The switch supports advertising the system maximum transmission unit (MTU) to IPv6 nodes and path MTU discovery. Path MTU discovery allows a host to dynamically discover and adjust to differences in the MTU size of every link along a given data path. In IPv6, if a link along the path is not large enough to accommodate the packet size, the source of the packet handles the fragmentation. The switch does not support path MTU discovery for multicast packets.

ICMPv6

The Internet Control Message Protocol (ICMP) in IPv6 generates error messages, such as ICMP destination unreachable messages, to report errors during processing and other diagnostic functions. In IPv6, ICMP packets are also used in the neighbor discovery protocol and path MTU discovery.

Neighbor Discovery

The switch supports NDP for IPv6, a protocol running on top of ICMPv6, and static neighbor entries for IPv6 stations that do not support NDP. The IPv6 neighbor discovery process uses ICMP messages and solicited-node multicast addresses to determine the link-layer address of a neighbor on the same network (local link), to verify the reachability of the neighbor, and to keep track of neighboring routers.

The switch supports ICMPv6 redirect for routes with mask lengths less than 64 bits. ICMP redirect is not supported for host routes or for summarized routes with mask lengths greater than 64 bits.

Neighbor discovery throttling ensures that the switch CPU is not unnecessarily burdened while it is in the process of obtaining the next hop forwarding information to route an IPv6 packet. The switch drops any additional IPv6 packets whose next hop is the same neighbor that the switch is actively trying to resolve. This drop avoids further load on the CPU.

Default Router Preference

The switch supports IPv6 default router preference (DRP), an extension in router advertisement messages. DRP improves the ability of a host to select an appropriate router, especially when the host is multihomed and the routers are on different links. The switch does not support the Route Information Option in RFC 4191.

An IPv6 host maintains a default router list from which it selects a router for traffic to offlink destinations. The selected router for a destination is then cached in the destination cache. NDP for IPv6 specifies that routers that are reachable or probably reachable are preferred over routers whose reachability is unknown or suspect. For reachable or probably reachable routers, NDP can either select the same router every time or cycle through the router list. By using DRP, you can configure an IPv6 host to prefer one router over another, provided both are reachable or probably reachable.

IPv6 Stateless Autoconfiguration and Duplicate Address Detection

The switch uses stateless autoconfiguration to manage link, subnet, and site addressing changes, such as management of host and mobile IP addresses. A host autonomously configures its own link-local address, and booting nodes send router solicitations to request router advertisements for configuring interfaces.

IPv6 Applications

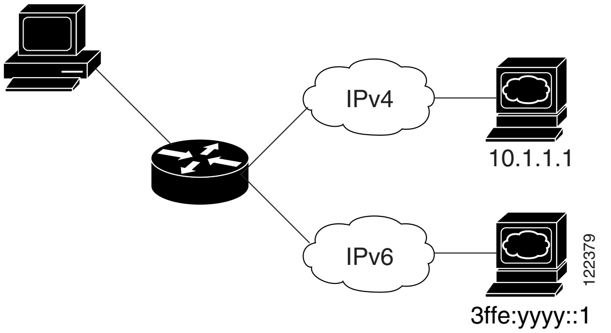
- Ping, traceroute, Telnet, TFTP, and FTP
- Secure Shell (SSH) over an IPv6 transport
- HTTP server access over IPv6 transport
- DNS resolver for AAAA over IPv4 transport
- Cisco Discovery Protocol (CDP) support for IPv6 addresses

Dual IPv4 and IPv6 Protocol Stacks

You must use the dual IPv4 and IPv6 template to allocate hardware memory usage to both IPv4 and IPv6 protocols.

Figure 107 shows a router forwarding both IPv4 and IPv6 traffic through the same interface, based on the IP packet and destination addresses.

Figure 107 Dual IPv4 and IPv6 Support on an Interface



Use the dual IPv4 and IPv6 switch database management (SDM) template to enable IPv6 routing dual stack environments (supporting both IPv4 and IPv6).

- If you try to configure IPv6 without first selecting a dual IPv4 and IPv6 template, a warning message appears.
- In IPv4-only environments, the switch routes IPv4 packets and applies IPv4 QoS and ACLs in hardware. IPv6 packets are not supported.
- In dual IPv4 and IPv6 environments, the switch routes both IPv4 and IPv6 packets and applies IPv4 QoS in hardware.
- IPv6 QoS is not supported.
- If you do not plan to use IPv6, do not use the dual stack template because it results in less hardware memory availability for each resource.

DHCP for IPv6 Address Assignment

DHCPv6 enables DHCP servers to pass configuration parameters, such as IPv6 network addresses, to IPv6 clients. The address assignment feature manages nonduplicate address assignment in the correct prefix based on the network where the host is connected. Assigned addresses can be from one or multiple prefix pools. Additional options, such as default domain and DNS name-server address, can be passed back to the client. Address pools can be assigned for use on a specific interface, on multiple interfaces, or the server can automatically find the appropriate pool.

Static Routes for IPv6

Static routes are manually configured and define an explicit route between two networking devices. Static routes are useful for smaller networks with only one path to an outside network or to provide security for certain types of traffic in a larger network.

RIP for IPv6

Routing Information Protocol (RIP) for IPv6 is a distance-vector protocol that uses hop count as a routing metric. It includes support for IPv6 addresses and prefixes and the all-RIP-routers multicast group address FF02::9 as the destination address for RIP update messages.

OSPF for IPv6

The switch supports Open Shortest Path First (OSPF) for IPv6, a link-state protocol for IP.

EIGRP IPv6

The switch supports Enhanced Interior Gateway Routing Protocol (EIGRP) for IPv6. It is configured on the interfaces on which it runs and does not require a global IPv6 address.

Before running, an instance of EIGRP IPv6 requires an implicit or explicit router ID. An implicit router ID is derived from a local IPv4 address, so any IPv4 node always has an available router ID. However, EIGRP IPv6 might be running in a network with only IPv6 nodes and therefore might not have an available IPv4 router ID.

Multiprotocol BGP for IPv6

Multiprotocol Border Gateway Protocol (BGP) is the supported exterior gateway protocol for IPv6. Multiprotocol BGP extensions for IPv6 support the same features and functionality as IPv4 BGP. IPv6 enhancements to multiprotocol BGP include support for IPv6 address family and network layer reachability information (NLRI) and next-hop (the next router in the path to the destination) attributes that use IPv6 addresses.

The switch does not support multicast BGP or non-stop forwarding (NSF) for IPv6 or for BGP IPv6.

SNMP and Syslog Over IPv6

To support both IPv4 and IPv6, IPv6 network management requires both IPv6 and IPv4 transports. Syslog over IPv6 supports address data types for these transports.

SNMP and syslog over IPv6 provide these features:

- Support for both IPv4 and IPv6
- IPv6 transport for SNMP and to modify the SNMP agent to support traps for an IPv6 host
- SNMP- and syslog-related MIBs to support IPv6 addressing
- Configuration of IPv6 hosts as trap receivers

For support over IPv6, SNMP modifies the existing IP transport mapping to simultaneously support IPv4 and IPv6. These SNMP actions support IPv6 transport management:

- Opens User Datagram Protocol (UDP) SNMP socket with default settings
- Provides a new transport mechanism called *SR_IPV6_TRANSPORT*
- Sends SNMP notifications over IPv6 transport
- Supports SNMP-named access lists for IPv6 transport
- Supports SNMP proxy forwarding using IPv6 transport
- Verifies SNMP Manager feature works with IPv6 transport

HTTP(S) Over IPv6

The HTTP client sends requests to both IPv4 and IPv6 HTTP servers, which respond to requests from both IPv4 and IPv6 HTTP clients. URLs with literal IPv6 addresses must be specified in hexadecimal using 16-bit values between colons.

The accept socket call chooses an IPv4 or IPv6 address family. The accept socket is either an IPv4 or IPv6 socket. The listening socket waits for both IPv4 and IPv6 signals that indicate a connection. The IPv6 listening socket is bound to an IPv6 wildcard address.

The underlying TCP/IP stack supports a dual-stack environment. HTTP relies on the TCP/IP stack and the sockets for processing network-layer interactions.

Basic network connectivity (**ping**) must exist between the client and the server hosts before HTTP connections can be made.

Unsupported IPv6 Unicast Routing Features

- IPv6 policy-based routing
- IPv6 virtual private network (VPN) routing and forwarding (VRF) table support
- Support for Intermediate System-to-Intermediate System (IS-IS) routing
- IPv6 packets destined to site-local addresses
- Tunneling protocols, such as IPv4-to-IPv6 or IPv6-to-IPv4
- The switch as a tunnel endpoint supporting IPv4-to-IPv6 or IPv6-to-IPv4 tunneling protocols
- IPv6 unicast reverse-path forwarding
- IPv6 general prefixes

- HSRP for IPv6

Prerequisites

Select a dual IPv4 and IPv6 template as described in the Dual IPv4 and IPv6 Protocol Stacks, page 972.

Guidelines and Limitations

Because IPv6 is implemented in switch hardware, some limitations occur due to the IPv6 compressed addresses in the hardware memory. This results in some loss of functionality and some feature limitations.

- When using user-network interface (UNI) or enhanced network interface (ENI) ports for any IPv6-related features, you must first globally enable IP routing and IPv6 routing on the switch by entering the **ip routing** and **ipv6 unicast-routing** global configuration commands even if you are not using IPv6 routing.
- ICMPv6 redirect functionality is not supported for IPv6 host routes (routes used to reach a specific host) or for IPv6 routes with masks greater than 64 bits. The switch cannot redirect hosts to a better first-hop router for a specific destination that is reachable through a host route or through a route with masks greater than 64 bits.
- Load balancing using equal cost and unequal cost routes is not supported for IPv6 host routes or for IPv6 routes with a mask greater than 64 bits.
- The switch cannot forward SNAP-encapsulated IPv6 packets.

  There is a similar limitation for IPv4 SNAP-encapsulated packets, but the packets are dropped at the switch.
- The switch routes IPv6-to-IPv4 and IPv4-to-IPv6 packets in hardware, but the switch cannot be an IPv6-to-IPv4 or IPv4-to-IPv6 tunnel endpoint.
- Bridged IPv6 packets with hop-by-hop extension headers are forwarded in software. In IPv4, these packets are routed in software but bridged in hardware.
- In addition to the normal SPAN and RSPAN limitations defined in the software configuration guide, these limitations are specific to IPv6 packets:
  - When you send RSPAN IPv6-routed packets, the source MAC address in the SPAN output packet might be incorrect.
  - When you send RSPAN IPv6-routed packets, the destination MAC address might be incorrect. Normal traffic is not affected.
- The switch cannot apply QoS classification or policy-based routing on source-routed IPv6 packets in hardware.
- The switch cannot generate ICMPv6 `Packet Too Big` messages for multicast packets.

Default Settings

Feature	Default Setting
SDM template	Default.
IPv6 routing	Disabled globally and on all interfaces.
CEFv6	Disabled (IPv4 CEF is enabled by default). **Note:** When IPv6 routing is enabled, CEFv6 is automatically enabled.
IPv6 addresses	None configured.

Configuring IPv6

Configuring IPv6 Addressing and Enabling IPv6 Routing

To forward IPv6 traffic on an interface, you must configure a global IPv6 address on that interface. Configuring an IPv6 address on an interface automatically configures a link-local address and activates IPv6 for the interface. The configured interface automatically joins these required multicast groups for that link:

- solicited-node multicast group FF02:0:0:0:0:1:ff00::/104 for each unicast address assigned to the interface (the address for the neighbor discovery process)
- all-nodes link-local multicast group FF02::1
- all-routers link-local multicast group FF02::2

For more information about configuring IPv6 routing, see the “Implementing Addressing and Basic Connectivity for IPv6” chapter in the *IPv6 Implementation Guide, Cisco IOS Release 15.2M&T*.

BEFORE YOU BEGIN

- Be sure to select a dual IPv4 and IPv6 SDM template.
- Not all features discussed in this chapter are supported by the switch. See Unsupported IPv6 Unicast Routing Features, page 974.

- In the **ipv6 address** interface configuration command, you must enter the *ipv6-address* and *ipv6-prefix* variables with the address specified in hexadecimal using 16-bit values between colons. The *prefix-length* variable (preceded by a slash [/]) is a decimal value that shows how many of the high-order contiguous bits of the address comprise the prefix (the network portion of the address).

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**sdm prefer dual-ipv4-and-ipv6** {**default** | **routing** | **vlan**}	Select an SDM template that supports IPv4 and IPv6. ■ **default**–Set the switch to the default template to balance system resources. ■ **routing**–Set the switch to the routing template to support IPv4 and IPv6 routing, including IPv4 policy-based routing. ■ **vlan**–Maximize VLAN configuration on the switch with no routing supported in hardware.
3.	**end**	Return to privileged EXEC mode.
4.	**reload**	Reload the operating system.
5.	**configure terminal**	Enter global configuration mode.
6.	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure. The interface can be a physical interface, a switch virtual interface (SVI), or a Layer 3 EtherChannel.
7.	**no switchport**	Remove the interface from Layer 2 configuration mode (if it is a physical interface).
8.	**ipv6 address** *ipv6-prefix/prefix length* **eui-64** or **ipv6 address** *ipv6-address* **link-local** or **ipv6 enable**	Specify a global IPv6 address with an extended unique identifier (EUI) in the low-order 64 bits of the IPv6 address. Specify only the network prefix; the last 64 bits are automatically computed from the switch MAC address. This enables IPv6 processing on the interface. Specify a link-local address on the interface to be used instead of the link-local address that is automatically configured when IPv6 is enabled on the interface. This command enables IPv6 processing on the interface. Automatically configure an IPv6 link-local address on the interface, and enable the interface for IPv6 processing. The link-local address can only be used to communicate with nodes on the same link.
9.	**exit**	Return to global configuration mode.
10.	**ip routing**	Enable IP routing on the switch.
11.	**ipv6 unicast-routing**	Enable forwarding of IPv6 unicast data packets.
12.	**end**	Return to privileged EXEC mode.
13.	**show ipv6 interface** *interface-id*	Verify your entries.
14.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove an IPv6 address from an interface, use the **no ipv6 address** *ipv6-prefix/prefix length* **eui-64** or **no ipv6 address** *ipv6-address* **link-local** interface configuration command. To remove all manually configured IPv6 addresses from an interface, use the **no ipv6 address** interface configuration command without arguments. To disable IPv6 processing on an interface that has not been explicitly configured with an IPv6 address, use the **no ipv6 enable** interface configuration command. To globally disable IPv6 routing, use the **no ipv6 unicast-routing** global configuration command.

EXAMPLE

This example shows how to enable IPv6 with both a link-local address and a global address based on the IPv6 prefix 2001:0DB8:c18:1::/64. The EUI-64 interface ID is used in the low-order 64 bits of both addresses. Output from the **show ipv6 interface** EXEC command is included to show how the interface ID (20B:46FF:FE2F:D940) is appended to the link-local prefix FE80::/64 of the interface.

```
Switch(config)# sdm prefer dual-ipv4-and-ipv6 default
Switch(config)# ipv6 unicast-routing
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# no switchport
Switch(config-if)# ipv6 address 2001:0DB8:c18:1::/64 eui 64
Switch(config-if)# end
Switch# show ipv6 interface gigabitethernet0/11
GigabitEthernet0/2 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::20B:46FF:FE2F:D940
  Global unicast address(es):
  2001:0DB8:c18:1:20B:46FF:FE2F:D940, subnet is 2001:0DB8:c18:1::/64 [EUI]
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::1:FF2F:D940
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds
  ND advertised reachable time is 0 milliseconds
  ND advertised retransmit interval is 0 milliseconds
  ND router advertisements are sent every 200 seconds
  ND router advertisements live for 1800 seconds
  Hosts use stateless autoconfig for addresses.
```

Configuring Default Router Preference

Router advertisement messages are sent with the default router preference (DRP) configured by the **ipv6 nd router-preference** interface configuration command. If no DRP is configured, router advertisements are sent with a medium preference.

A DRP is useful when two routers on a link might provide equivalent, but not equal-cost routing, and policy might dictate that hosts should prefer one of the routers.

BEFORE YOU BEGIN

Complete the Configuring IPv6 Addressing and Enabling IPv6 Routing, page 976.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Enter interface configuration mode, and enter the Layer 3 interface on which you want to specify the DRP.
3.	**ipv6 nd router-preference** {**high** | **medium** | **low**}	Specify a DRP for the router on the switch interface.
4.	**end**	Return to privileged EXEC mode.
5.	**show ipv6 interface**	Verify the configuration.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no ipv6 nd router-preference** interface configuration command to disable an IPv6 DRP.

EXAMPLE

This example shows how to configure a DRP of *high* for the router on an interface:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ipv6 nd router-preference high
Switch(config-if)# end
```

Configuring IPv4 and IPv6 Protocol Stacks

Follow this procedure to configure a Layer 3 interface to support both IPv4 and IPv6 and to enable IPv6 routing.

BEFORE YOU BEGIN

Before configuring IPv6 routing, you must select an SDM template that supports IPv4 and IPv6. If not already configured, use the **sdm prefer dual-ipv4-and-ipv6** {**default** | **routing** | **vlan**} **global** configuration command to configure a template that supports IPv6. When you select a new template, you must reload the switch by using the **reload** privileged EXEC command so that the template takes effect.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**sdm prefer dual-ipv4-and-ipv6** {**default** | **routing** | **vlan**}	Select an SDM template that supports IPv4 and IPv6. ■ **default**–Set the switch to the default template to balance system resources. ■ **routing**–Set the switch to the routing template to support IPv4 and IPv6 routing, including IPv4 policy-based routing. ■ **vlan**–Maximize VLAN configuration on the switch with no routing supported in hardware.
3.	**end**	Return to privileged EXEC mode.
4.	**reload**	Reload the operating system.
5.	**configure terminal**	Enter global configuration mode.
6.	**ip routing**	Enable IPv4 routing on the switch.
7.	**ipv6 unicast-routing**	Enable forwarding of IPv6 data packets on the switch.
8.	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
9.	**no switchport**	Remove the interface from Layer 2 configuration mode (if it is a physical interface).
10.	**ip address** *ip-address mask* [**secondary**]	Specify a primary or secondary IPv4 address for the interface.
11.	**ipv6 address** *ipv6-prefix/prefix length* **eui-64** or **ipv6 address** *ipv6-address* **link-local** or **ipv6 enable**	Specify a global IPv6 address. Specify only the network prefix; the last 64 bits are automatically computed from the switch MAC address. Specify a link-local address on the interface to be used instead of the automatically configured link-local address when IPv6 is enabled on the interface. Automatically configure an IPv6 link-local address on the interface, and enable the interface for IPv6 processing. The link-local address can only be used to communicate with nodes on the same link.
12.	**end**	Return to privileged EXEC mode.
13.	**show interface** *interface-id* **show ip interface** *interface-id* **show ipv6 interface** *interface-id*	Verify your entries.
14.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable IPv4 routing, use the **no ip routing** global configuration command. To disable IPv6 routing, use the **no ipv6 unicast-routing** global configuration command. To remove an IPv4 address from an interface, use the **no ip address** *ip-address mask* interface configuration command. To remove an IPv6 address from an interface, use the **no ipv6 address** *ipv6-prefix/prefix length* **eui-64** or **no ipv6 address** *ipv6-address* **link-local** interface configuration command.

To remove all manually configured IPv6 addresses from an interface, use the **no ipv6 address** interface configuration command without arguments. To disable IPv6 processing on an interface that has not been explicitly configured with an IPv6 address, use the **no ipv6 enable** interface configuration command.

EXAMPLE

This example shows how to enable IPv4 and IPv6 routing on an interface:

```
Switch(config)# sdm prefer dual-ipv4-and-ipv6 default
Switch(config)# ip routing
Switch(config)# ipv6 unicast-routing
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# no switchport
Switch(config-if)# ip address 192.168.99.1 244.244.244.0
Switch(config-if)# ipv6 address 2001:0DB8:c18:1::/64 eui 64
Switch(config-if)# end
```

Configuring DHCP for IPv6 Address Assignment

This document describes only the DHCPv6 address assignment. For more information about configuring the DHCPv6 client, server, or relay agent functions, see the “Implementing DHCP for IPv6” chapter in the *IPv6 Implementation Guide, Cisco IOS Release 15.2M&T*.

- Default DHCPv6 Address Assignment Configuration, page 982
- DHCPv6 Address Assignment Configuration Guidelines, page 982
- Enabling the DHCPv6 Server Function, page 982
- Enabling the DHCPv6 Client Function, page 985

Default DHCPv6 Address Assignment Configuration

By default, no Dynamic Host Configuration Protocol for IPv6 (DHCPv6) features are configured on the switch.

DHCPv6 Address Assignment Configuration Guidelines

When configuring a DHCPv6 address assignment, consider these guidelines:

- In the procedures, the specified interface must be one of these Layer 3 interfaces:
  - DHCPv6 IPv6 routing must be enabled on a Layer 3 interface.
  - SVI: a VLAN interface created by using the **interface vlan** *vlan_id* command.
  - EtherChannel port channel in Layer 3 mode: a port-channel logical interface created by using the **interface port-channel port-channel-number** command.
- Before configuring DHCPv6, you must select a Switch Database Management (SDM) template that supports IPv4 and IPv6.
- The switch can act as a DHCPv6 client, server, or relay agent. The DHCPv6 client, server, and relay function are mutually exclusive on an interface.

Enabling the DHCPv6 Server Function

BEFORE YOU BEGIN

See DHCPv6 Address Assignment Configuration Guidelines, page 982.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ipv6 dhcp pool** *poolname*	Enter DHCP pool configuration mode, and define the name for the IPv6 DHCP pool. The pool name can be a symbolic string (such as Engineering) or an integer (such as 0).
3.	**address prefix** *IPv6-prefix* **lifetime** *{t1 t1* | **infinite***}*	(Optional) Specify an address prefix for address assignment. This address must be in hexadecimal, using 16-bit values between colons. ■ **lifetime** *t1 t1*—Specify a time interval (in seconds) that an IPv6 address prefix remains in the valid state. The range is 5 to 4294967295 seconds. Specify **infinite** for no time interval.
4.	**link-address** *IPv6-prefix*	(Optional) Specify a link-address IPv6 prefix. When an address on the incoming interface or a link-address in the packet matches the specified IPv6 prefix, the server uses the configuration information pool. This address must be in hexadecimal, using 16-bit values between colons.
5.	**vendor-specific** *vendor-id*	(Optional) Enter vendor-specific configuration mode, and enter a vendor-specific identification number. This number is the vendor IANA Private Enterprise Number. The range is 1 to 4294967295.
6.	**suboption** *number {***address** *IPv6-address* | **ascii** *ASCII-string* | **hex** *hex-string}*	(Optional) Enter a vendor-specific suboption number. The range is 1 to 65535. Enter an IPv6 address, ASCII text, or a hex string as defined by the suboption parameters.
7.	**exit**	Return to DHCP pool configuration mode.
8.	**exit**	Return to global configuration mode.
9.	**interface** *interface-id*	Enter interface configuration mode, and specify the interface to configure.

	Command	Purpose
10.	**ipv6 dhcp server** [*poolname* | **automatic**] [**rapid-commit**] [**preference** *value*] [**allow-hint**]	Enable the DHCPv6 server function on an interface. ■ *poolname*–(Optional) User-defined name for the IPv6 DHCP pool. The pool name can be a symbolic string (such as Engineering) or an integer (such as 0). ■ **automatic**–(Optional) Enables the system to automatically determine which pool to use when allocating addresses for a client. ■ **rapid-commit**–(Optional) Allow two-message exchange method. ■ **preference** *value*–(Optional) The preference value carried in the preference option in the advertise message sent by the server. The range is from 0 to 255. The preference value default is 0. ■ **allow-hint**–(Optional) Specifies whether the server should consider client suggestions in the SOLICIT message. By default, the server ignores client hints.
11.	**end**	Return to privileged EXEC mode.
12.	**show ipv6 dhcp pool** or **show ipv6 dhcp interface**	Verify DHCPv6 pool configuration. Verify that the DHCPv6 server function is enabled on an interface.
13.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete a DHCPv6 pool, use the **no ipv6 dhcp pool** *poolname* global configuration command. Use the **no** form of the DHCP pool configuration mode commands to change the DHCPv6 pool characteristics. To disable the DHCPv6 server function on an interface, use the **no ipv6 dhcp server** interface configuration command.

EXAMPLE

This example shows how to configure a pool called *engineering* with an IPv6 address prefix:

```
Switch# configure terminal
Switch(config)# ipv6 dhcp pool engineering
Switch(config-dhcpv6)#address prefix 2001:1000::0/64
Switch(config-dhcpv6)# end
```

This example shows how to configure a pool called *testgroup* with three link-addresses and an IPv6 address prefix:

```
Switch# configure terminal
Switch(config)# ipv6 dhcp pool testgroup
Switch(config-dhcpv6)# link-address 2001:1001::0/64
Switch(config-dhcpv6)# link-address 2001:1002::0/64
Switch(config-dhcpv6)# link-address 2001:2000::0/48
Switch(config-dhcpv6)# address prefix 2001:1003::0/64
Switch(config-dhcpv6)# end
```

This example shows how to configure a pool called *350* with vendor-specific options:

```
Switch# configure terminal
Switch(config)# ipv6 dhcp pool 350
Switch(config-dhcpv6)# address prefix 2001:1005::0/48
Switch(config-dhcpv6)# vendor-specific 9
Switch(config-dhcpv6-vs)# suboption 1 address 1000:235D::1
Switch(config-dhcpv6-vs)# suboption 2 ascii "IP-Phone"
Switch(config-dhcpv6-vs)# end
```

Enabling the DHCPv6 Client Function

BEFORE YOU BEGIN

See DHCPv6 Address Assignment Configuration Guidelines, page 982.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Enter interface configuration mode, and specify the interface to configure.
3.	**ipv6 address dhcp** [**rapid-commit**]	Enable the interface to acquire an IPv6 address from the DHCPv6 server. **rapid-commit**–(Optional) Allow two-message exchange method for address assignment.
4.	**ipv6 dhcp client request** [**vendor-specific**]	(Optional) Enable the interface to request the vendor-specific option.
5.	**end**	Return to privileged EXEC mode.
6.	**show ipv6 dhcp interface**	Verify that the DHCPv6 client is enabled on an interface.

To disable the DHCPv6 client function, use the **no ipv6 address dhcp** interface configuration command. To remove the DHCPv6 client request, use the **no ipv6 address dhcp client request** interface configuration command.

EXAMPLE

This example shows how to acquire an IPv6 address and to enable the rapid-commit option:

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# ipv6 address dhcp rapid-commit
```

Configuring IPv6 ICMP Rate Limiting

ICMP rate limiting is enabled by default with a default interval between error messages of 100 milliseconds and a bucket size (maximum number of tokens to be stored in a bucket) of 10.

BEFORE YOU BEGIN

Complete the Configuring IPv6 Addressing and Enabling IPv6 Routing, page 976.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ipv6 icmp error-interval** *interval* [*bucketsize*]	Configure the interval and bucket size for IPv6 ICMP error messages: ■ *interval*–The interval (in milliseconds) between tokens being added to the bucket. The range is from 0 to 2147483647 milliseconds. ■ *bucketsize*–(Optional) The maximum number of tokens stored in the bucket. The range is from 1 to 200.
3.	**end**	Return to privileged EXEC mode.
4.	**show ipv6 interface** [*interface-id*]	Verify your entries.
5.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default configuration, use the **no ipv6 icmp error-interval** global configuration command.

EXAMPLE

This example shows how to configure an IPv6 ICMP error message interval of 50 milliseconds and a bucket size of 20 tokens:

```
Switch(config)#ipv6 icmp error-interval 50 20
```

Configuring CEF for IPv6

Cisco Express Forwarding (CEF) is a Layer 3 IP switching technology, allowing more CPU processing power to be dedicated to packet forwarding. IPv4 CEF is enabled by default. IPv6 CEF is disabled by default, but automatically enabled when you configure IPv6 routing.

To route IPv6 unicast packets, first globally configure forwarding of IPv6 unicast packets by using the **ipv6 unicast-routing** global configuration command. You must also configure an IPv6 address and IPv6 processing on an interface by using the **ipv6 address** interface configuration command.

To disable IPv6 CEF, use the **no ipv6 cef** global configuration command. To reenable IPv6 CEF, use the **ipv6 cef** global configuration command. You can verify the IPv6 state by entering the **show ipv6 cef** privileged EXEC command.

For more information about configuring CEF, see the “Implementing IPv6 Addressing and Basic Connectivity” chapter in the *IPv6 Implementation Guide, Cisco IOS Release 15.2M&T*.

Configuring Static Routing for IPv6

BEFORE YOU BEGIN

Before configuring a static IPv6 route, you must:

- Enable routing by using the **ip routing** global configuration command.
- Enable the forwarding of IPv6 packets by using the **ipv6 unicast-routing** global configuration command.
- Enable IPv6 on at least one Layer 3 interface by configuring an IPv6 address on the interface.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ipv6 route** *ipv6-prefix/prefix length* {*ipv6-address* | *interface-id* [*ipv6-address*]} [*administrative distance*]	Configure a static IPv6 route. - *ipv6-prefix*—The IPv6 network that is the destination of the static route. It can also be a hostname when static host routes are configured. - */prefix length*—The length of the IPv6 prefix. A decimal value that shows how many of the high-order contiguous bits comprise the prefix (the network portion of the address). A slash mark must precede the decimal value. - *ipv6-address*—The IPv6 address of the next hop that can be used to reach the specified network. The next hop does not need to be directly connected; recursion finds the IPv6 address of the directly connected next hop. The address must be specified in hexadecimal using 16-bit values between colons. - *interface-id*—Specify direct static routes from point-to-point and broadcast interfaces. On point-to-point interfaces, you do not need to specify the IPv6 address of the next hop. On broadcast interfaces, you should always specify the IPv6 address of the next hop, or ensure that the specified prefix is assigned to the link, specifying a link-local address as the next hop. You can optionally specify the IPv6 address of the next hop to which packets are sent. **Note:** You must specify an *interface-id* when using a link-local address as the next hop. The link-local next hop must be an adjacent router. - *administrative distance*—(Optional) An administrative distance. The range is 1 to 254; the default value is 1, which gives static routes precedence over all but connected routes. To configure a floating static route, use an administrative distance greater than that of the dynamic routing protocol.

	Command	Purpose
3.	**end**	Return to privileged EXEC mode.
4.	**show ipv6 static** [*ipv6-address* | *ipv6-prefix/prefix length*] [**interface** *interface-id*] [**recursive**] [**detail**] or **show ipv6 route static** [*updated*]	Verify your entries by displaying the IPv6 routing table. ■ **interface** *interface-id*–(Optional) Display only those static routes with the specified interface as an egress interface. ■ **recursive**–(Optional) Display only recursive static routes. The **recursive** keyword is mutually exclusive with the **interface** keyword, but it can be used with or without the IPv6 prefix in the command syntax. ■ **detail**–(Optional) Display this additional information: – For valid recursive routes, the output path set, and maximum resolution depth. – For invalid routes, the reason why the route is not valid.
5.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove a configured static route, use the **no ipv6 route** *ipv6-prefix/prefix length* {*ipv6-address* | *interface-id* [*ipv6-address*]} [*administrative distance*] global configuration command.

For more information about configuring static IPv6 routing, see the “Implementing Static Routes for IPv6” chapter in the *IPv6 Implementation Guide, Cisco IOS Release 15.2M&T*.

EXAMPLE

This example shows how to configure a floating static route to an interface. The route has an administrative distance of 130:

```
Switch(config)# ipv6 route 2001:0DB8::/32 gigabitethernet0/1 130
```

Configuring RIP for IPv6

BEFORE YOU BEGIN

Before configuring the switch to run IPv6 RIP, you must:

- Enable routing by using the **ip routing** global configuration command.
- Enable the forwarding of IPv6 packets by using the **ipv6 unicast-routing** global configuration command.
- Enable IPv6 on any Layer 3 interfaces on which IPv6 RIP is to be enabled.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ipv6 router rip** *name*	Configure an IPv6 RIP routing process, and enter router configuration mode for the process.
3.	**maximum-paths** *number-paths*	(Optional) Define the maximum number of equal-cost routes that IPv6 RIP can support. The range is from 1 to 64, and the default is 4 routes.
4.	**exit**	Return to global configuration mode.
5.	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
6.	**ipv6 rip** *name* **enable**	Enable the specified IPv6 RIP routing process on the interface.
7.	**ipv6 rip** *name* **default-information** {**only** | **originate**}	(Optional) Originate the IPv6 default route (::/0) into the RIP routing process updates sent from the specified interface. **Note:** To avoid routing loops after the IPv6 default route (::/0) is originated from any interface, the routing process ignores all default routes received on any interface. ■ **only**—Select to originate the default route, but suppress all other routes in the updates sent on this interface. ■ **originate**—Select to originate the default route in addition to all other routes in the updates sent on this interface.
8.	**end**	Return to privileged EXEC mode.
9.	**show ipv6 rip** [*name*] [**interface** *interface-id*] [**database**] [**next-hops**] or **show ipv6 route rip** [*updated*]	Display information about current IPv6 RIP processes. Display the current contents of the IPv6 routing table.
10.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable a RIP routing process, use the **no ipv6 router rip** *name* global configuration command. To disable the RIP routing process for an interface, use the **no ipv6 rip** *name* interface configuration command.

For more information about configuring RIP routing for IPv6, see the “Implementing RIP for IPv6” chapter in the *IPv6 Implementation Guide, Cisco IOS Release 15.2M&T*.

EXAMPLE

This example shows how to enable the RIP routing process *cisco* with a maximum of eight equal-cost routes and to enable it on an interface:

```
Switch(config)# ipv6 router rip cisco
Switch(config-router)# maximum-paths 8
Switch(config)# exit
Switch(config)# interface gigabitethernet0/3
Switch(config-if)# ipv6 rip cisco enable
```

Configuring OSPF for IPv6

You can customize OSPF for IPv6 for your network. However, the defaults are set to meet the requirements of most customers and features.

Be careful when changing the defaults for IPv6 commands. Doing so might adversely affect OSPF for the IPv6 network.

BEFORE YOU BEGIN

Before you enable IPv6 OSPF on an interface, you must:

- Enable routing by using the **ip routing** global configuration command.
- Enable the forwarding of IPv6 packets by using the **ipv6 unicast-routing** global configuration command.
- Enable IPv6 on Layer 3 interfaces on which you are enabling IPv6 OSPF.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ipv6 router ospf** *process-id*	Enable OSPF router configuration mode for the process. The process ID is the number assigned administratively when enabling the OSPF for IPv6 routing process. It is locally assigned and can be a positive integer from 1 to 65535.
3.	**area** *area-id* **range** {*ipv6-prefix/prefix length*} [**advertise** | **not-advertise**] [**cost** *cost*]	(Optional) Consolidate and summarize routes at an area boundary. ■ *area-id*–Identifier of the area about which routes are to be summarized. It can be specified as either a decimal value or as an IPv6 prefix. ■ *ipv6-prefix/prefix length*–The destination IPv6 network and a decimal value that shows how many of the high-order contiguous bits of the address comprise the prefix (the network portion of the address). A slash mark (/) must precede the decimal value. ■ **advertise**–(Optional) Set the address range status to advertise and to generate a Type 3 summary link-state advertisement (LSA). ■ **not-advertise**–(Optional) Set the address range status to DoNotAdvertise. The Type 3 summary LSA is suppressed, and component networks remain hidden from other networks. ■ **cost** *cost*–(Optional) Metric or cost for this summary route, which is used during OSPF SPF calculation to determine the shortest paths to the destination. The value can be 0 to 16777215.
4.	**maximum paths** *number-paths*	(Optional) Define the maximum number of equal-cost routes to the same destination that IPv6 OSPF should enter in the routing table. The range is from 1 to 64, and the default is 16 paths.
5.	**exit**	Return to global configuration mode.
6.	**interface** *interface-id*	Enter interface configuration mode, and specify the Layer 3 interface to configure.
7.	**ipv6 ospf** *process-id* **area** *area-id* [**instance** *instance-id*]	Enable OSPF for IPv6 on the interface. ■ **instance** *instance-id*–(Optional) Instance identifier.

	Command	Purpose
8.	**end**	Return to privileged EXEC mode.
9.	**show ipv6 ospf** [*process-id*] [*area-id*] **interface** [*interface-id*] or **show ipv6 ospf** [*process-id*] [*area-id*]	Display information about OSPF interfaces. Display general information about OSPF routing processes.
10.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable an OSPF routing process, use the no **ipv6 router ospf** *process-id* global configuration command. To disable the OSPF routing process for an interface, use the **no ipv6 ospf** *process-id* **area** *area-id* interface configuration command.

For more information about configuring OSPF routing for IPv6, see the "Implementing OSPF for IPv6" chapter in the *IPv6 Implementation Guide, Cisco IOS Release 15.2M&T*.

Configuring EIGRP for IPv6

By default, EIGRP for IPv6 is disabled. You can configure EIGRP for IPv6 on an interface. After configuring the router and the interface for EIGRP, enter the **no shutdown** privileged EXEC command to start EIGRP.

Note: If EIGRP for IPv6 is not in shutdown mode, EIGRP might start running before you enter the EIRGP router-mode commands to configure the router and the interface.

To set an explicit router ID, use the **show ipv6 eigrp** command to see the configured router IDs, and then use the **router-id** command.

As with EIGRP IPv4, you can use EIGRPv6 to specify your EIGRP IPv4 interfaces and to select a subset of those as passive interfaces. Use the **passive-interface default** command to make all interfaces passive, and then use the **no passive-interface** command on selected interfaces to make them active. EIGRP IPv6 does not need to be configured on a passive interface.

For more configuration procedures, see the "Implementing EIGRP for IPv6" chapter in the *IPv6 Implementation Guide, Cisco IOS Release 15.2M&T*.

Configuring BGP for IPv6

When configuring multiprotocol BGP extensions for IPv6, you must create the BGP routing process, configure peering relationships, and customize BGP for your particular network. Note that BGP functions the same in IPv6 as in IPv4.

BEFORE YOU BEGIN

Before configuring the router to run BGP for IPv6, you must use the **ipv6 unicast-routing** command to globally enable IPv6 routing.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**router bgp** *as-number*	Configure a BGP routing process, and enter BGP router configuration mode for the autonomous system number.
3.	**no bgp default ipv4-unicast**	Disable the IPv4 unicast address family for the BGP routing process specified in the previous step. Routing information for the IPv4 unicast address family is advertised by default for each BGP routing session unless you enter this command before configuring the **neighbor remote-as** command.
4.	**bgp router-id** *ip-address*	(Optional) Configure a fixed 32-bit router ID as the identifier of the local router running BGP. By default, the router ID is the IPv4 address of a router loopback interface. On a router enabled only for IPv6 (no IPv4 address), you must manually configure the BGP router ID. **Note:** Configuring a router ID by using this command resets all active BGP peering sessions.
5.	**neighbor** {*ip-address* | *ipv6-address[%] interface-type interface-number* | *peer-group-name*} **remote-as** *as-number*	Add the IPv6 address of the neighbor in the specified autonomous system to the IPv6 multiprotocol BGP neighbor table of the local router. **Note:** The ipv6-address must be in hexadecimal, using 16-bit values between colons.
6.	**address-family ipv6**	Specify the IPv6 address family and enter address family configuration mode
7.	**neighbor** {*ip-address* | *peer-group-name* | *ipv6-address*} **activate**	Enable the neighbor to exchange prefixes for the IPv6 address family with the local router.
8.	**end**	Return to privileged EXEC mode.
9.	**show bgp ipv6**	Display information about IPv6 BGP configuration.
10.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

For more configuration procedures, see the “Implementing Multiprotocol BGP for IPv6” chapter in the *IPv6 Implementation Guide, Cisco IOS Release 15.2M&T*.

The switch does not support multicast IPv6 BGP, nonstop forwarding (NSF) for IPv6 BGP, 6PE multipath (EoMPLS), or IPv6 VRF.

EXAMPLE

```
router bgp 1
  bgp router-id 1.1.1.1
  no bgp default ipv4-unicast
!--- Without configuring ""no bgp default ipv4-unicast"" only IPv4 will be
!--- advertised
  bgp log-neighbor-changes
  neighbor 2010:AB8:0:2:C601:10FF:FE58:0 remote-as 2
  !
  address-family ipv6
    neighbor 2010:AB8:0:2:C601:10FF:FE58:0 activate
    network 2010:AB8:2::/48
    network 2010:AB8:3::/48
  exit-address-family
!
```

Verifying Configuration

Command	Purpose
show bgp ipv6	Display BGP IPv6 configuration and routing tables.
show ipv6 access-list	Display IPv6 access lists.
show ipv6 cef	Display Cisco Express Forwarding for IPv6.
show ipv6 interface *interface-id*	Display IPv6 interface status and configuration.
show ipv6 mtu	Display IPv6 MTU per destination cache.
show ipv6 neighbors	Display IPv6 neighbor cache entries.
show ipv6 ospf	Display IPv6 OSPF information.
show ipv6 prefix-list	Display IPv6 prefix lists.
show ipv6 protocols	Display IPv6 routing protocols on the switch.
show ipv6 rip	Display IPv6 RIP routing protocol status.
show ipv6 route	Display IPv6 route table entries.
show ipv6 routers	Display local IPv6 routers.
show ipv6 static	Display IPv6 static routes.
show ipv6 traffic	Display IPv6 traffic statistics.

Command	Purpose
show ipv6 eigrp [*as-number*] *interface*	Display information about interfaces configured for EIGRP IPv6.
show ipv6 eigrp [*as-number*] *neighbor*	Display the neighbors discovered by EIGRP IPv6.
show ipv6 eigrp [*as-number*] *traffic*	Display the number of EIGRP IPv6 packets sent and received.
show ipv6 eigrp topology [*as-number* | *ipv6-address*] [**active** | **all-links** | **detail-links** | **pending** | **summary** | **zero-successors**]	Display EIGRP entries in the IPv6 topology table.

Command	Purpose
show ip http server history	Display the previous 20 connections to the HTTP server, including the IP address accessed and the time when the connection was closed.
show ip http server connection	Display the current connections to the HTTP server, including the local and remote IP addresses being accessed.
show ip http client connection	Display the configuration values for HTTP client connections to HTTP servers.
show ip http client history	Display a list of the last 20 requests made by the HTTP client to the server.

Configuration Example

This is an example of the output from the **show ipv6 interface** privileged EXEC command:

```
Switch# show ipv6 interface
Vlan1 is up, line protocol is up
```

```
  IPv6 is enabled, link-local address is FE80::20B:46FF:FE2F:D940
  Global unicast address(es):
    3FFE:C000:0:1:20B:46FF:FE2F:D940, subnet is 3FFE:C000:0:1::/64 [EUI]
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::1:FF2F:D940
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds
  ND advertised reachable time is 0 milliseconds
  ND advertised retransmit interval is 0 milliseconds
  ND router advertisements are sent every 200 seconds
  ND router advertisements live for 1800 seconds
<output truncated>
```

This is an example of the output from the **show ipv6 cef** privileged EXEC command:

```
Switch# show ipv6 cef
::/0
  nexthop 3FFE:C000:0:7::777 Vlan7
3FFE:C000:0:1::/64
  attached to Vlan1
3FFE:C000:0:1:20B:46FF:FE2F:D940/128
  receive
3FFE:C000:0:7::/64
  attached to Vlan7
3FFE:C000:0:7::777/128
  attached to Vlan7
3FFE:C000:0:7:20B:46FF:FE2F:D97F/128
  receive
3FFE:C000:111:1::/64
  attached to GigabitEthernet0/11
3FFE:C000:111:1:20B:46FF:FE2F:D945/128
  receive
3FFE:C000:168:1::/64
  attached to GigabitEthernet0/43
3FFE:C000:168:1:20B:46FF:FE2F:D94B/128
  receive
3FFE:C000:16A:1::/64
  attached to Loopback10
3FFE:C000:16A:1:20B:46FF:FE2F:D900/128
  receive

<output truncated>
```

This is an example of the output from the **show ipv6 protocols** privileged EXEC command:

```
Switch# show ipv6 protocols
IPv6 Routing Protocol is “connected”
IPv6 Routing Protocol is “static”
IPv6 Routing Protocol is “rip fer”
  Interfaces:
    Vlan6
GigabitEthernet0/4
GigabitEthernet0/11
GigabitEthernet0/12
  Redistribution:
    None
```

This is an example of the output from the **show ipv6 rip** privileged EXEC command:

```
Switch# show ipv6 rip
```

```
RIP process "fer", port 521, multicast-group FF02::9, pid 190
     Administrative distance is 120. Maximum paths is 16
     Updates every 30 seconds, expire after 180
     Holddown lasts 0 seconds, garbage collect after 120
     Split horizon is on; poison reverse is off
     Default routes are not generated
     Periodic updates 9040, trigger updates 60
  Interfaces:
    Vlan6
GigabitEthernet0/4
GigabitEthernet0/11
GigabitEthernet0/12
Redistribution:
    None
```

This is an example of the output from the **show ipv6 neighbor** privileged EXEC command:

```
Switch# show ipv6 neighbors
IPv6 Address                              Age Link-layer Addr State Interface
3FFE:C000:0:7::777                          - 0007.0007.0007  REACH Vl7
3FFE:C101:113:1::33                         - 0000.0000.0033  REACH Gi0/13
```

This is an example of the output from the **show ipv6 static** privileged EXEC command:

```
Switch# show ipv6 static
IPv6 Static routes
Code: * - installed in RIB
* ::/0 via nexthop 3FFE:C000:0:7::777, distance 1
```

This is an example of the output from the **show ipv6 route** privileged EXEC command:

```
Switch# show ipv6 route
IPv6 Routing Table - 21 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
S   ::/0 [1/0]
     via 3FFE:C000:0:7::777
C   3FFE:C000:0:1::/64 [0/0]
     via ::, Vlan1
L   3FFE:C000:0:1:20B:46FF:FE2F:D940/128 [0/0]
     via ::, Vlan1
C   3FFE:C000:0:7::/64 [0/0]
     via ::, Vlan7
L   3FFE:C000:0:7:20B:46FF:FE2F:D97F/128 [0/0]
     via ::, Vlan7
C   3FFE:C000:111:1::/64 [0/0]
     via ::, GigabitEthernet0/11
L   3FFE:C000:111:1:20B:46FF:FE2F:D945/128 [0/0]
C   3FFE:C000:168:1::/64 [0/0]
     via ::, GigabitEthernet0/4
L   3FFE:C000:168:1:20B:46FF:FE2F:D94B/128 [0/0]
     via ::, GigabitEthernet0/4
C   3FFE:C000:16A:1::/64 [0/0]
     via ::, Loopback10
L   3FFE:C000:16A:1:20B:46FF:FE2F:D900/128 [0/0]
     via ::, Loopback10

<output truncated>
```

This is an example of the output from the **show ipv6 traffic** privileged EXEC command.

```
Switch# show ipv6 traffic
IPv6 statistics:
  Rcvd:  1 total, 1 local destination
         0 source-routed, 0 truncated
         0 format errors, 0 hop count exceeded
         0 bad header, 0 unknown option, 0 bad source
         0 unknown protocol, 0 not a router
         0 fragments, 0 total reassembled
         0 reassembly timeouts, 0 reassembly failures
  Sent:  36861 generated, 0 forwarded
         0 fragmented into 0 fragments, 0 failed
         0 encapsulation failed, 0 no route, 0 too big
         0 RPF drops, 0 RPF suppressed drops
  Mcast: 1 received, 36861 sent

ICMP statistics:
  Rcvd: 1 input, 0 checksum errors, 0 too short
        0 unknown info type, 0 unknown error type
        unreach: 0 routing, 0 admin, 0 neighbor, 0 address, 0 port
        parameter: 0 error, 0 header, 0 option
        0 hopcount expired, 0 reassembly timeout,0 too big
        0 echo request, 0 echo reply
        0 group query, 0 group report, 0 group reduce
        1 router solicit, 0 router advert, 0 redirects
        0 neighbor solicit, 0 neighbor advert
  Sent: 10112 output, 0 rate-limited
        unreach: 0 routing, 0 admin, 0 neighbor, 0 address, 0 port
        parameter: 0 error, 0 header, 0 option
        0 hopcount expired, 0 reassembly timeout,0 too big
        0 echo request, 0 echo reply
        0 group query, 0 group report, 0 group reduce
        0 router solicit, 9944 router advert, 0 redirects
        84 neighbor solicit, 84 neighbor advert

UDP statistics:
  Rcvd: 0 input, 0 checksum errors, 0 length errors
        0 no port, 0 dropped
  Sent: 26749 output

TCP statistics:
  Rcvd: 0 input, 0 checksum errors
  Sent: 0 output, 0 retransmitted
```

Related Documents

For information about how Cisco Systems implements IPv6:

- http://www.cisco.com/en/US/products/ps6553/products_ios_technology_home.html

For information about IPv6 and other features in this chapter:

- IPv6 Configuration Library, Cisco IOS Release 15M&T
- IPv6 Implementation Guide, Cisco IOS Release 15.2M&T

Unicast Overview

This document describes how to configure unicast routing on the Cisco Industrial Ethernet Switches, hereafter referred to as switch. To use unicast routing, the switch must be running the IP services image.

This chapter provides an overview of the following unicast routing features:

IPv4 Unicast Routing

Routers and Layer 3 switches can route packets in the following ways:

- By using default routing—sending traffic with a destination unknown to the router to a default outlet or destination.
- By using preprogrammed static routes for the traffic

  Static unicast routing forwards packets from predetermined ports through a single path into and out of a network. Static routing does not automatically respond to changes in the network and therefore, might result in unreachable destinations.
- By dynamically calculating routes by using a routing protocol

  Dynamic routing protocols are used by routers to dynamically calculate the best route for forwarding traffic. Routing protocols supported by the switch are Routing Information Protocol (RIP), Border Gateway Protocol (BGP), Open Shortest Path First (OSPF) protocol, Enhanced IGRP (EIGRP), System-to-Intermediate System (IS-IS), and Bidirectional Forwarding Detection (BFD).

IPv6 Unicast Routing

IPv4 users can move to IPv6 and receive services such as end-to-end security, quality of service (QoS), and globally unique addresses. The IPv6 address space reduces the need for private addresses and Network Address Translation (NAT) processing by border routers at network edges.

IPv6 unicast routing support on the switch includes expanded address capability, header format simplification, improved support of extensions and options, and hardware parsing of the extension header. The switch supports hop-by-hop extension header packets, which are routed or bridged in software.

The switch provides IPv6 routing capability over 802.1Q trunk ports for static routes, Routing Information Protocol (RIP) for IPv6, and Open Shortest Path First (OSPF) Version 3 Protocol. It supports up to 16 equal-cost routes and can simultaneously forward IPv4 and IPv6 frames at line rate.

Enhanced Object Tracking

Enhanced object tracking on the switch provides a more complete alternative to the Hot Standby Routing Protocol (HSRP) tracking mechanism, which allows you to track the line-protocol state of an interface. If the line protocol state of an interface goes down, the HSRP priority of the interface is reduced and another HSRP device with a higher priority becomes active. The enhanced object tracking feature separates the tracking mechanism from HSRP and creates a separate, standalone tracking process that can be used by processes other than HSRP. This allows tracking other objects in addition to the interface line-protocol state.

A client process, such as HSRP or Gateway Local Balancing Protocol (GLBP), can register an interest in tracking objects and request notification when the tracked object changes state.This feature increases the availability and speed of recovery of a routing system and decreases outages and outage duration.

Configuring Cisco IOS IP SLAs Operations

Prerequisites for Configuring Cisco IOS IP SLAs Operations

- Before configuring any IP SLAs application, we recommend that you verify the operation type supported on your software image by using the **show ip sla application** privileged EXEC command.

Restrictions for Configuring Cisco IOS IP SLAs Operations

- The IP SLAs responder can be a Cisco IOS Layer 2, responder-configurable switch.
- The switch does not support Voice over IP (VoIP) service levels using the gatekeeper registration delay operations measurements. Before configuring any IP SLAs application, you can use the **show ip sla application** privileged EXEC command to verify that the operation type is supported on your software image.

Information About Configuring Cisco IOS IP SLAs Operations

This chapter describes how to use Cisco IOS IP Service Level Agreements (SLAs) on the switch. Cisco IP SLAs is a part of Cisco IOS software that allows Cisco customers to analyze IP service levels for IP applications and services by using active traffic monitoring—the generation of traffic in a continuous, reliable, and predictable manner—for measuring network performance. With Cisco IOS IP SLAs, service provider customers can measure and provide service level agreements, and enterprise customers can verify service levels, verify outsourced service level agreements, and understand network performance. Cisco IOS IP SLAs can perform network assessments, verify quality of service (QoS), ease the deployment of new services, and assist with network troubleshooting.

Cisco IOS IP SLAs

Cisco IOS IP SLAs sends data across the network to measure performance between multiple network locations or across multiple network paths. It simulates network data and IP services and collects network performance information in real time. Cisco IOS IP SLAs generates and analyzes traffic either between Cisco IOS devices or from a Cisco IOS device to a remote IP device such as a network application server. Measurements provided by the various Cisco IOS IP SLAs operations can be used for troubleshooting, for problem analysis, and for designing network topologies.

Depending on the specific Cisco IOS IP SLAs operation, various network performance statistics are monitored within the Cisco device and stored in both command-line interface (CLI) and Simple Network Management Protocol (SNMP) MIBs. IP SLAs packets have configurable IP and application layer options such as source and destination IP address, User Datagram Protocol (UDP)/TCP port numbers, a type of service (ToS) byte (including Differentiated Services Code Point [DSCP] and IP Prefix bits), Virtual Private Network (VPN) routing/forwarding instance (VRF), and URL web address.

Because Cisco IP SLAs is Layer 2 transport independent, you can configure end-to-end operations over disparate networks to best reflect the metrics that an end user is likely to experience. IP SLAs collects a unique subset of these performance metrics:

- Delay (both round-trip and one-way)
- Jitter (directional)

- Packet loss (directional)
- Packet sequencing (packet ordering)
- Path (per hop)
- Connectivity (directional)
- Server or website download time

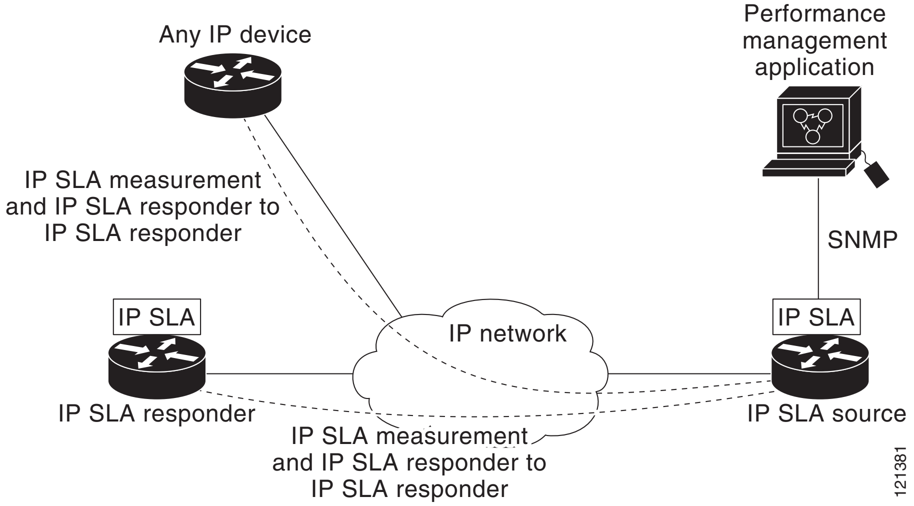
Because Cisco IOS IP SLAs is SNMP-accessible, it can also be used by performance-monitoring applications like CiscoWorks Internetwork Performance Monitor (IPM) and other third-party Cisco partner performance management products. Using IP SLAs can provide these benefits:

- Service-level agreement monitoring, measurement, and verification.
- Network performance monitoring
  - Measures the jitter, latency, or packet loss in the network.
  - Provides continuous, reliable, and predictable measurements.
- IP service network health assessment to verify that the existing QoS is sufficient for new IP services.
- Edge-to-edge network availability monitoring for proactive verification and connectivity testing of network resources (for example, shows the network availability of an NFS server used to store business critical data from a remote site).
- Troubleshooting of network operation by providing consistent, reliable measurement that immediately identifies problems and saves troubleshooting time.
- Multiprotocol Label Switching (MPLS) performance monitoring and network verification (if the switch supports MPLS)

Cisco IOS IP SLAs to Measure Network Performance

You can use IP SLAs to monitor the performance between any area in the network—core, distribution, and edge—without deploying a physical probe. It uses generated traffic to measure network performance between two networking devices. Figure 108 on page 1003 shows how IP SLAs begins when the source device sends a generated packet to the destination device. After the destination device receives the packet, depending on the type of IP SLAs operation, it responds with time-stamp information for the source to make the calculation on performance metrics. An IP SLAs operation performs a network measurement from the source device to a destination in the network using a specific protocol such as UDP.

Figure 108 Cisco IOS IP SLAs Operation


Any IP device
Performance management application
IP SLA measurement and IP SLA responder to IP SLA responder
SNMP
IP SLA
IP network
IP SLA
IP SLA responder
IP SLA source
IP SLA measurement and IP SLA responder to IP SLA responder
121381

To implement IP SLAs network performance measurement, you need to perform these tasks:

- Enable the IP SLAs responder, if required.
- Configure the required IP SLAs operation type.
- Configure any options available for the specified operation type.
- Configure threshold conditions, if required.
- Schedule the operation to run, then let the operation run for a period of time to gather statistics.
- Display and interpret the results of the operation using the Cisco IOS CLI or a network management system (NMS) system with SNMP.

IP SLAs Responder and IP SLAs Control Protocol

The IP SLAs responder is a component embedded in the destination Cisco device that allows the system to anticipate and respond to IP SLAs request packets. The responder provides accurate measurements without the need for dedicated probes. The responder uses the Cisco IOS IP SLAs Control Protocol to provide a mechanism through which it can be notified on which port it should listen and respond. Only a Cisco IOS device can be a source for a destination IP SLAs Responder.

Figure 108 on page 1003 shows where the Cisco IOS IP SLAs responder fits in the IP network. The responder listens on a specific port for control protocol messages sent by an IP SLAs operation. Upon receipt of the control message, it enables the specified UDP or TCP port for the specified duration. During this time, the responder accepts the requests and responds to them. It disables the port after it responds to the IP SLAs packet, or when the specified time expires. MD5 authentication for control messages is available for added security.

You do not need to enable the responder on the destination device for all IP SLAs operations. For example, a responder is not required for services that are already provided by the destination router (such as Telnet or HTTP). You cannot configure the IP SLAs responder on non-Cisco devices and Cisco IOS IP SLAs can send operational packets only to services native to those devices.

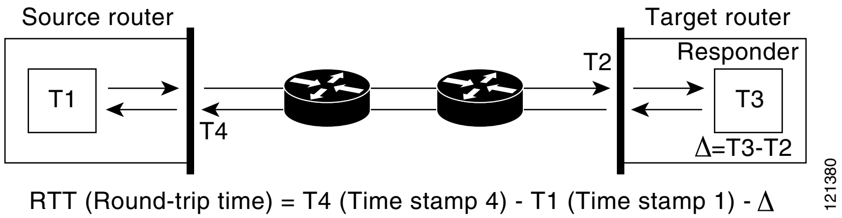
Response Time Computation for IP SLAs

Switches and routers can take tens of milliseconds to process incoming packets due to other high priority processes. This delay affects the response times because the test-packet reply might be in a queue while waiting to be processed. In this situation, the response times would not accurately represent true network delays. IP SLAs minimizes these processing delays on the source device as well as on the target device (if the responder is being used) to determine true round-trip times. IP SLAs test packets use time stamping to minimize the processing delays.

When the IP SLAs responder is enabled, it allows the target device to take time stamps when the packet arrives on the interface at interrupt level and again just as it is leaving, eliminating the processing time. This time stamping is made with a granularity of sub-milliseconds (ms).

Figure 109 on page 1004 demonstrates how the responder works. Four time stamps are taken to make the calculation for round-trip time. At the target router, with the responder functionality enabled, time stamp 2 (TS2) is subtracted from time stamp 3 (TS3) to produce the time spent processing the test packet as represented by delta. This delta value is then subtracted from the overall round-trip time. Notice that the same principle is applied by IP SLAs on the source router where the incoming time stamp 4 (TS4) is also taken at the interrupt level to allow for greater accuracy.

Figure 109 Cisco IOS IP SLAs Responder Time Stamping


Source router
Target router
Responder
T1
T2
T3
T4
Δ=T3-T2
RTT (Round-trip time) = T4 (Time stamp 4) - T1 (Time stamp 1) - Δ
121380

An additional benefit of the two time stamps at the target device is the ability to track one-way delay, jitter, and directional packet loss. Because much network behavior is asynchronous, it is critical to have these statistics. However, to capture one-way delay measurements, you must configure both the source router and target router with Network Time Protocol (NTP) so that the source and target are synchronized to the same clock source. One-way jitter measurements do not require clock synchronization.

IP SLAs Operation Scheduling

When you configure an IP SLAs operation, you must schedule the operation to begin capturing statistics and collecting error information. You can schedule an operation to start immediately or to start at a certain month, day, and hour. You can use the pending option to set the operation to start at a later time. The pending option is an internal state of the operation that is visible through SNMP. The pending state is also used when an operation is a reaction (threshold) operation waiting to be triggered. You can schedule a single IP SLAs operation or a group of operations at one time.

You can schedule several IP SLAs operations on a switch running the IP services image by using a single command through the Cisco IOS CLI or the CISCO RTTMON-MIB. Scheduling the operations to run at evenly distributed times allows you to control the amount of IP SLAs monitoring traffic. This distribution of IP SLAs operations helps minimize the CPU utilization and thus improves network scalability.

IP SLAs Operation Threshold Monitoring

To support successful service level agreement monitoring, you must have mechanisms that notify you immediately of any possible violation. IP SLAs can send SNMP traps that are triggered by events such as these:

- Connection loss
- Timeout
- Round-trip time threshold

- Average jitter threshold
- One-way packet loss
- One-way jitter
- One-way mean opinion score (MOS)
- One-way latency

An IP SLAs threshold violation can also trigger another IP SLAs operation for further analysis. For example, the frequency could be increased or an ICMP path echo or ICMP path jitter operation could be initiated for troubleshooting.

Determining the type of threshold and the level to set can be complex, and depends on the type of IP service being used in the network.

IP Service Levels by Using the UDP Jitter Operation

Jitter means interpacket delay variance. When multiple packets are sent consecutively 10 ms apart from source to destination, if the network is behaving correctly, the destination should receive them 10 ms apart. But if there are delays in the network (like queuing, arriving through alternate routes, and so on) the arrival delay between packets might be more than or less than 10 ms with a positive jitter value meaning that the packets arrived more than 10 ms apart. If the packets arrive 12 ms apart, positive jitter is 2 ms; if the packets arrive 8 ms apart, negative jitter is 2 ms. For delay-sensitive networks, positive jitter values are undesirable, and a jitter value of 0 is ideal.

In addition to monitoring jitter, the IP SLAs UDP jitter operation can be used as a multipurpose data gathering operation. The packets IP SLAs generates carry packet sending and receiving sequence information and sending and receiving time stamps from the source and the operational target. Based on these, UDP jitter operations measure this data:

- Per-direction jitter (source to destination and destination to source)
- Per-direction packet-loss
- Per-direction delay (one-way delay)
- Round-trip delay (average round-trip time)

Because the paths for the sending and receiving of data can be different (asymmetric), you can use the per-direction data to more readily identify where congestion or other problems are occurring in the network.

The UDP jitter operation generates synthetic (simulated) UDP traffic and sends a number of UDP packets, each of a specified size, sent a specified number of milliseconds apart, from a source router to a target router, at a given frequency. By default, ten packet-frames, each with a payload size of 10 bytes are generated every 10 ms, and the operation is repeated every 60 seconds. You can configure each of these parameters to best simulate the IP service you want to provide.

To provide accurate one-way delay (latency) measurements, time synchronization, such as that provided by NTP, is required between the source and the target device. Time synchronization is not required for the one-way jitter and packet loss measurements. If the time is not synchronized between the source and target devices, one-way jitter and packet loss data is returned, but values of *0* are returned for the one-way delay measurements provided by the UDP jitter operation

Note: Before you configure a UDP jitter operation on the source device, you must enable the IP SLAs responder on the target device (the operational target).

IP Service Levels by Using the ICMP Echo Operation

The ICMP echo operation measures end-to-end response time between a Cisco device and any devices using IP. Response time is computed by measuring the time taken between sending an ICMP echo request message to the destination and receiving an ICMP echo reply. Many customers use IP SLAs ICMP-based operations, in-house ping testing, or ping-based dedicated probes for response time measurements between the source IP SLAs device and the destination IP device. The IP SLAs ICMP echo operation conforms to the same specifications as ICMP ping testing, and the two methods result in the same response times.

Note: This operation does not require the IP SLAs responder to be enabled.

How to Configure Cisco IOS IP SLAs Operations

Note: Not all of the IP SLAs commands or operations described in this guide are supported on the switch. The switch supports IP service level analysis by using UDP jitter, UDP echo, HTTP, TCP connect, ICMP echo, ICMP path echo, ICMP path jitter, FTP, DNS, and DHCP, as well as multiple operation scheduling and proactive threshold monitoring. It does not support VoIP service levels using the gatekeeper registration delay operations measurements.

Configuring the IP SLAs Responder

Before You Begin

For the IP SLAs responder to function, you must also configure a source device, such as a Catalyst 3750 or Catalyst 3560 switch running the IP services image, that has full IP SLAs support. Refer to the documentation for the source device for configuration information.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**ip sla responder** {**tcp-connect** | **udp-echo**} **ipaddress** *ip-address* **port** *port-number*	Configures the switch as an IP SLAs responder. The optional keywords have these meanings: ■ **tcp-connect**–Enables the responder for TCP connect operations. ■ **udp-echo**–Enables the responder for User Datagram Protocol (UDP) echo or jitter operations. ■ **ipaddress** *ip-address*–Enters the destination IP address. ■ **port** *port-number*–Enters the destination port number. **Note:** The IP address and port number must match those configured on the source device for the IP SLAs operation.
3.	**end**	Returns to privileged EXEC mode.

Configuring UDP Jitter Operation

Before You Begin

Before you configure a UDP jitter operation on the source device, you must enable the IP SLAs responder on the target device (the operational target).

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**ip sla** *operation-number*	Creates an IP SLAs operation, and enters IP SLAs configuration mode.
3.	**udp-jitter** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*}] [**source-port** *port-number*] [**control** {**enable** | **disable**}] [**num-packets** *number-of-packets*] [**interval** *interpacket-interval*]	Configures the IP SLAs operation as a UDP jitter operation, and enters UDP jitter configuration mode. ■ *destination-ip-address* | *destination-hostname*–Specifies the destination IP address or hostname. ■ *destination-port*–Specifies the destination port number in the range from 1 to 65535. ■ (Optional) **source-ip** {*ip-address* | *hostname*}–Specifies the source IP address or hostname. When a source IP address or hostname is not specified, IP SLAs chooses the IP address nearest to the destination. ■ (Optional) **source-port** *port-number*–Specifies the source port number in the range from 1 to 65535. When a port number is not specified, IP SLAs chooses an available port. ■ (Optional) **control**–Enables or disables sending of IP SLAs control messages to the IP SLAs responder. By default, IP SLAs control messages are sent to the destination device to establish a connection with the IP SLAs responder. ■ (Optional) **num-packets** *number-of-packets*–Enters the number of packets to be generated. The range is 1 to 6000; the default is 10. ■ (Optional) **interval** *inter-packet-interval*–Enters the interval between sending packets in milliseconds. The range is 1 to 6000; the default value is 20 ms.
4.	**frequency** *seconds*	(Optional) Sets the rate at which a specified IP SLAs operation repeats. The range is from 1 to 604800 seconds; the default is 60 seconds.

	Command	Purpose
5.	**exit**	Exits UDP jitter configuration mode, and returns to global configuration mode.
6.	**ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm* [:*ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**]	Configures the scheduling parameters for an individual IP SLAs operation. ■ *operation-number*–Enters the RTR entry number. ■ (Optional) **life**–Sets the operation to run indefinitely (**forever**) or for a specific number of *seconds*. The range is from 0 to 2147483647. The default is 3600 seconds (1 hour). ■ (Optional) **start-time**–Enters the time for the operation to begin collecting information: – To start at a specific time, enter the hour, minute, second (in 24-hour notation), and day of the month. If no month is entered, the default is the current month. – Enter **pending** to select no information collection until a start time is selected. – Enter **now** to start the operation immediately. – Enter **after** *hh:mm:ss* to show that the operation should start after the entered time has elapsed. ■ (Optional) **ageout** *seconds*–Enters the number of seconds to keep the operation in memory when it is not actively collecting information. The range is 0 to 2073600 seconds, the default is 0 seconds (never ages out). ■ (Optional) **recurring**–Sets the operation to automatically run every day.
7.	**end**	Returns to privileged EXEC mode.

Analyzing IP Service Levels by Using the ICMP Echo Operation

Note: This operation does not require the IP SLAs responder to be enabled.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**ip sla** *operation-number*	Creates an IP SLAs operation and enters IP SLAs configuration mode.
3.	**icmp-echo** {*destination-ip-address* | *destination-hostname*} [**source-ip** {*ip-address* | *hostname*} | **source-interface** *interface-id*]	Configures the IP SLAs operation as an ICMP Echo operation and enters ICMP echo configuration mode. ■ *destination-ip-address* | *destination-hostname*–Specifies the destination IP address or hostname. ■ (Optional) **source-ip** {*ip-address* | *hostname*}–Specifies the source IP address or hostname. When a source IP address or hostname is not specified, IP SLAs chooses the IP address nearest to the destination. ■ (Optional) **source-interface** *interface-id*–Specifies the source interface for the operation.
4.	**frequency** *seconds*	(Optional) Sets the rate at which a specified IP SLAs operation repeats. The range is from 1 to 604800 seconds; the default is 60 seconds.

	Command	Purpose
5.	**exit**	Exits UDP jitter configuration mode, and returns to global configuration mode.
6.	**ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm* [:*ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**]	Configures the scheduling parameters for an individual IP SLAs operation. ■ *operation-number*–Enters the RTR entry number. ■ (Optional) **life**–Sets the operation to run indefinitely (**forever**) or for a specific number of *seconds*. The range is from 0 to 2147483647. The default is 3600 seconds (1 hour). ■ (Optional) **start-time**–Enters the time for the operation to begin collecting information: – To start at a specific time, enter the hour, minute, second (in 24-hour notation), and day of the month. If no month is entered, the default is the current month. – Enter **pending** to select no information collection until a start time is selected. – Enter **now** to start the operation immediately. – Enter **after** *hh:mm:ss* to indicate that the operation should start after the entered time has elapsed. ■ (Optional) **ageout** *seconds*–Enters the number of seconds to keep the operation in memory when it is not actively collecting information. The range is 0 to 2073600 seconds; the default is 0 seconds (never ages out). ■ (Optional) **recurring**–Sets the operation to automatically run every day.
7.	end	Returns to privileged EXEC mode.

Monitoring and Maintaining Cisco IP SLAs Operations

Command	Purpose
show ip sla application	Displays global information about Cisco IOS IP SLAs.
show ip sla authentication	Displays IP SLAs authentication information.
show ip sla configuration [*entry-number*]	Displays configuration values including all defaults for all IP SLAs operations or a specific operation.
show ip sla enhanced-history {**collection-statistics** | **distribution statistics**} [*entry-number*]	Displays enhanced history statistics for collected history buckets or distribution statistics for all IP SLAs operations or a specific operation.
show ip sla ethernet-monitor configuration [*entry-number*]	Displays IP SLAs automatic Ethernet configuration.
show ip sla event-publisher	Displays the list of client applications that are registered to receive IP SLAs notifications.
show ip sla group schedule [*schedule-entry-number*]	Displays IP SLAs group scheduling configuration and details.
show ip sla history [*entry-number* / **full** / **tabular**]	Displays history collected for all IP SLAs operations

Command	Purpose
show ip sla mpls-lsp-monitor {**collection-statistics** | **configuration** | **ldp operational-state** | **scan-queue** | **summary** [*entry-number*] | **neighbors**}	Displays MPLS label switched path (LSP) Health Monitor operations.
show ip sla reaction-configuration [*entry-number*]	Displays the configured proactive threshold monitoring settings for all IP SLAs operations or a specific operation.
show ip sla reaction-trigger [*entry-number*]	Displays the reaction trigger information for all IP SLAs operations or a specific operation.
show ip sla responder	Displays information about the IP SLAs responder.
show ip sla standards	Displays information about the IP SLAs standards.
show ip sla statistics [*entry-number* | **aggregated** | **details**]	Displays current or aggregated operational status and statistics.

Configuration Examples for Configuring Cisco IP SLAs Operations

Configuring an ICMP Echo IP SLAs Operation: Example

This example shows how to configure an ICMP echo IP SLAs operation:

```
Switch(config)# ip sla 12
Switch(config-ip-sla)# icmp-echo 172.29.139.134
Switch(config-ip-sla-echo)# frequency 30
Switch(config-ip-sla-echo)# exit
Switch(config)# ip sla schedule 5 start-time now life forever
Switch(config)# end
Switch# show ip sla configuration 22
IP SLAs, Infrastructure Engine-II.

Entry number: 12
Owner:
Tag:
Type of operation to perform: echo
Target address: 2.2.2.2
Source address: 0.0.0.0
Request size (ARR data portion): 28
Operation timeout (milliseconds): 5000
Type Of Service parameters: 0x0
Verify data: No
Vrf Name:
Schedule:
    Operation frequency (seconds): 60
    Next Scheduled Start Time: Pending trigger
    Group Scheduled : FALSE
    Randomly Scheduled : FALSE
    Life (seconds): 3600
    Entry Ageout (seconds): never
    Recurring (Starting Everyday): FALSE
    Status of entry (SNMP RowStatus): notInService
Threshold (milliseconds): 5000
Distribution Statistics:
    Number of statistic hours kept: 2
    Number of statistic distribution buckets kept: 1
    Statistic distribution interval (milliseconds): 20
History Statistics:
    Number of history Lives kept: 0
    Number of history Buckets kept: 15
    History Filter Type: None
Enhanced History:
```

Sample Output for Show IP SLA Command: Example

This is an example of the output from the command:

```
Switch# show ip sla application

        IP SLAs
Version: 2.2.0 Round Trip Time MIB, Infrastructure Engine-II
Time of last change in whole IP SLAs: 22:17:39.117 UTC Fri Jun
Estimated system max number of entries: 15801

Estimated number of configurable operations: 15801
Number of Entries configured  : 0
Number of active Entries      : 0
Number of pending Entries     : 0
Number of inactive Entries    : 0

        Supported Operation Types
Type of Operation to Perform: 802.1agEcho
Type of Operation to Perform: 802.1agJitter
Type of Operation to Perform: dhcp
Type of Operation to Perform: dns
Type of Operation to Perform: echo
Type of Operation to Perform: ftp
Type of Operation to Perform: http
Type of Operation to Perform: jitter
Type of Operation to Perform: pathEcho
Type of Operation to Perform: pathJitter
Type of Operation to Perform: tcpConnect
Type of Operation to Perform: udpEcho

IP SLAs low memory water mark: 21741224
```

Configuring a Responder UDP Jitter IP SLAs Operation: Example

This example shows how to configure the device as a responder for the UDP jitter IP SLAs operation in the next procedure:

```
Switch(config)# ip sla responder udp-echo 172.29.139.134 5000
```

Configuring a UDP Jitter IP SLAs Operation: Example

This example shows how to configure a UDP jitter IP SLAs operation:

```
Switch(config)# ip sla 10
Switch(config-ip-sla)# udp-jitter 172.29.139.134 5000
Switch(config-ip-sla-jitter)# frequency 30
Switch(config-ip-sla-jitter)# exit
Switch(config)# ip sla schedule 5 start-time now life forever
Switch(config)# end
Switch# show ip sla configuration 10
IP SLAs, Infrastructure Engine-II.
```

```
Entry number: 10
Owner:
Tag:
Type of operation to perform: udp-jitter
Target address/Source address: 1.1.1.1/0.0.0.0
Target port/Source port: 2/0
Request size (ARR data portion): 32
Operation timeout (milliseconds): 5000
Packet Interval (milliseconds)/Number of packets: 20/10
Type Of Service parameters: 0x0
Verify data: No
Vrf Name:
Control Packets: enabled
Schedule:
    Operation frequency (seconds): 30
    Next Scheduled Start Time: Pending trigger
    Group Scheduled : FALSE
    Randomly Scheduled : FALSE
    Life (seconds): 3600
    Entry Ageout (seconds): never
    Recurring (Starting Everyday): FALSE
    Status of entry (SNMP RowStatus): notInService
Threshold (milliseconds): 5000
Distribution Statistics:
    Number of statistic hours kept: 2
    Number of statistic distribution buckets kept: 1
    Statistic distribution interval (milliseconds): 20
Enhanced History:
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*
IP SLAs commands and configuration	*Cisco IOS IP SLAs Configuration Guide* on Cisco.com *Cisco IOS IP SLAs Command Reference* on Cisco.com

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Dying Gasp

This chapter describes the Dying-Gasp feature for the Cisco Industrial Ethernet series switches.

Dying Gasp resides on a hardware component on the High-performance WAN Interface Card (HWIC) and supports Gigabit Ethernet interfaces. The networking devices rely on a temporary back-up power supply on a capacitor, that allows for a graceful shutdown and the generation of the dying-gasp message. This temporary power supply is designed to last from 10 to 20 milliseconds to perform these tasks.

Dying-Gasp packets are created when you configure the host by using the **dying-gasp** configuration command. The **show dying-gasp packets** command displays the detailed information about the created packets.

The SNMP server for the SNMP Dying Gasp message is specified through the **snmp-server host** configuration command. The syslog server sending the syslog Dying Gasp message is specified through the **logging host hostname-or-ipaddress transport udp** command. The Ethernet-OAM Dying Gasp packets are created for interfaces where Ethernet-OAM is enabled.

Dying Gasp packets can be sent to a maximum number of 5 servers for each notification type.

For more information about configuring Dying Gasp, see the Configuring Dying Gasp chapter of the System Management guide at this URL:
http://www.cisco.com/c/en/us/td/docs/switches/connectedgrid/cg-switch-sw-master/software/configuration/guide/sysmgmt/CGS_1000_Sysmgmt/cgs_dying_gasp.html

Configuring Enhanced Object Tracking

This chapter describes how to configure enhanced object tracking. This feature provides a more complete alternative to the Hot Standby Routing Protocol (HSRP) tracking mechanism, which allows you to track the line-protocol state of an interface. If the line protocol state of an interface goes down, the HSRP priority of the interface is reduced and another HSRP device with a higher priority becomes active. The enhanced object tracking feature separates the tracking mechanism from HSRP and creates a separate, standalone tracking process that can be used by processes other than HSRP. This allows tracking other objects in addition to the interface line-protocol state.

A client process, such as HSRP or Gateway Local Balancing Protocol (GLBP), can register an interest in tracking objects and request notification when the tracked object changes state.This feature increases the availability and speed of recovery of a routing system and decreases outages and outage duration.

For more information about enhanced object tracking and the commands used to configure it, see this URL:

http://www.cisco.com/en/US/docs/ios/12_2t/12_2t15/feature/guide/fthsrptk.html

The chapter includes these sections:

Understanding Enhanced Object Tracking

Each tracked object has a unique number that is specified in the tracking command-line interface (CLI). Client processes use this number to track a specific object. The tracking process periodically polls the tracked object for value changes and sends any changes (as up or down values) to interested client processes, either immediately or after a specified delay. Several clients can track the same object, and can take different actions when the object changes state.

You can also track a combination of objects in a list by using either a weight threshold or a percentage threshold to measure the state of the list. You can combine objects using Boolean logic. A tracked list with a Boolean "AND" function requires that each object in the list be in an up state for the tracked object to be up. A tracked list with a Boolean "OR" function needs only one object in the list to be in the up state for the tracked object to be up.

Configuring Enhanced Object Tracking Features

- Configuring Static Routing Support, page 1025

Default Configuration

No type of object tracking is configured.

Tracking Interface Line-Protocol or IP Routing State

You can track either the interface line protocol state or the interface IP routing state. When you track the IP routing state, these three conditions are required for the object to be up:

- IP routing must be enabled and active on the interface.
- The interface line-protocol state must be up.
- The interface IP address must be known.

If all three of these conditions are not met, the IP routing state is down.

Beginning in privileged EXEC mode, follow these steps to track the line-protocol state or IP routing state of an interface:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**track** *object-number* **interface** *interface-id* **line-protocol**	(Optional) Create a tracking list to track the line-protocol state of an interface and enter tracking configuration mode. ■ The *object-number* identifies the tracked object and can be from 1 to 500. ■ The **interface** *interface-id* is the interface being tracked.
3.	**delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}	(Optional) Specify a period of time in seconds to delay communicating state changes of a tracked object. The range is from 1 to 180 seconds.
4.	**exit**	Return to global configuration mode.
5.	**track** *object-number* **interface** *interface-id* **ip routing**	(Optional) Create a tracking list to track the IP routing state of an interface, and enter tracking configuration mode. IP-route tracking tracks an IP route in the routing table and the ability of an interface to route IP packets. ■ The *object-number* identifies the tracked object and can be from 1 to 500. ■ The **interface** *interface-id* is the interface being tracked.
6.	**delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}	(Optional) Specify a period of time in seconds to delay communicating state changes of a tracked object. The range is from 1 to 180 seconds.
7.	**end**	Return to privileged EXEC mode.
8.	**show track** *object-number*	Verify that the specified objects are being tracked.
9.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

This example configures the tracking of an interface line-protocol state and verifies the configuration:

```
Switch(config)# track 33 interface GigabitEthernet1/17 line-protocol
Switch(config-track)# end
Switch# show track 33
Track 33
```

```
Interface GigabitEthernet1/17 line-protocol
Line protocol is Down (hw down)
  1 change, last change 00:18:28
```

Configuring a Tracked List

You can configure a tracked list of objects with a Boolean expression, a weight threshold, or a percentage threshold. A tracked list contains one or more objects. An object must exist before it can be added to the tracked list.

- You configure a Boolean expression to specify calculation by using either “AND” or “OR” operators.
- When you measure the tracked list state by a weight threshold, you assign a weight number to each object in the tracked list. The state of the tracked list is determined by whether or not the threshold was met. The state of each object is determined by comparing the total weight of all objects against a threshold weight for each object.
- When you measure the tracked list by a percentage threshold, you assign a percentage threshold to all objects in the tracked list. The state of each object is determined by comparing the assigned percentages of each object to the list.

Configuring a Tracked List with a Boolean Expression

Configuring a tracked list with a Boolean expression enables calculation by using either “AND” or “OR” operators. For example, when tracking two interfaces using the “AND” operator, *up* means that both interfaces are up, and *down* means that either interface is down.

Beginning in privileged EXEC mode, follow these steps to configure a tracked list of objects with a Boolean expression:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**track** *track-number* **list boolean** {**and** | **or**}	Configure a tracked list object, and enter tracking configuration mode. The *track-number* can be from 1 to 500. - **boolean**–Specify the state of the tracked list based on a Boolean calculation. - **and**–Specify that the list is up if all objects are up or down if one or more objects are down. - **or**–Specify that the list is up if one object is up or down if all objects are down.
3.	**object** *object-number* [**not**]	Specify the object to be tracked. The range is from 1 to 500. The keyword **not** negates the state of the object, which means that when the object is up, the tracked list detects the object as down. **Note:** An object must exist before you can add it to a tracked list.
4.	**delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}	(Optional) Specify a period of time in seconds to delay communicating state changes of a tracked object. The range is from 1 to 180 seconds.
5.	**end**	Return to privileged EXEC mode.
6.	**show track** *object-number*	Verify that the specified objects are being tracked.
7.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no track** *track-number* global configuration command to delete the tracked list.

This example configures track list 4 with a Boolean AND expression that contains two objects with one object state negated. If the list is up, the list detects that object 2 is down:

```
Switch(config)# track 4 list boolean and
Switch(config-track)# object 1
Switch(config-track)# object 2 not
Switch(config-track)# exit
```

Configuring a Tracked List with a Weight Threshold

To track by weight threshold, configure a tracked list of objects, specify that weight is used as the threshold, and configure a weight for each of its objects. The state of each object is determined by comparing the total weight of all objects that are up against a threshold weight for each object.

You cannot use the Boolean "NOT" operator in a weight threshold list.

Beginning in privileged EXEC mode, follow these steps to configure a tracked list of objects by using a weight threshold and to configure a weight for each object:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**track** *track-number* **list threshold weight**	Configure a tracked list object and enter tracking configuration mode. The *track-number* can be from 1 to 500. ■ **threshold**–Specify the state of the tracked list based on a threshold. ■ **weight**–Specify that the threshold is based on weight.
3.	**object** *object-number* [**weight** *weight-number*]	Specify the object to be tracked. The range is from 1 to 500. The optional **weight** *weight-number* specifies a threshold weight for the object. The range is from 1 to 255. **Note:** An object must exist before you can add it to a tracked list.
4.	**threshold weight** {**up** *number* | [**down** *number*]}	Specify the threshold weight. ■ **up** *number*–The valid range is from 1 to 255. ■ **down** *number*–(Optional) The range depends on the number selected for the **up** *number*. If you configure the **up** *number* as 25, the range shown for the down number is 0 to 24.
5.	**delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}	(Optional) Specify a period of time in seconds to delay communicating state changes of a tracked object. The range is from 1 to 180 seconds.
6.	**end**	Return to privileged EXEC mode.
7.	**show track** *object-number*	Verify that the specified objects are being tracked.
8.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no track** *track-number* global configuration command to delete the tracked list.

The example configures track list 4 to track by weight threshold. If object 1 and object 2 are down, then track list 4 is up because object 3 satisfies the up threshold value of up 30. But if object 3 is down, both objects 1 and 2 must be up in order to satisfy the threshold weight.

```
Switch(config)# track 4 list threshold weight
Switch(config-track)# object 1 weight 15
Switch(config-track)# object 2 weight 20
Switch(config-track)# object 3 weight 30
Switch(config-track)# threshold weight up 30 down 10
Switch(config-track)# exit
```

This configuration can be useful if object 1 and object 2 represent two small bandwidth connections and object 3 represents one large bandwidth connection. The configured **down 10** value means that once the tracked object is up, it will not go down until the threshold value is equal to or lower than 10, which in this example means that all connections are down.

Configuring a Tracked List with a Percentage Threshold

To track by percentage threshold, configure a tracked list of objects, specify that a percentage will be used as the threshold, and specify a percentage for all objects in the list. The state of the list is determined by comparing the assigned percentage of each object to the list.

You cannot use the Boolean "NOT" operator in a percentage threshold list.

Beginning in privileged EXEC mode, follow these steps to configure a tracked list of objects by using a percentage threshold:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**track** *track-number* **list threshold percentage**	Configure a tracked list object and enter tracking configuration mode. The *track-number* can be from 1 to 500. ■ **threshold**–Specify the state of the tracked list based on a threshold. ■ **percentage**–Specify that the threshold is based on percentage.
3.	**object** *object-number*	Specify the object to be tracked. The range is from 1 to 500. **Note:** An object must exist before you can add it to a tracked list.
4.	**threshold percentage** {**up** *number* | [**down** *number*]}	Specify the threshold percentage. ■ **up** *number*–The valid range is from 1 to 100. ■ **down** *number*]–(Optional) The range depends on the number selected for the **up** *number*. If you configure the **up** *number* as 25, the range shown for the down number is 0 to 24.
5.	**delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}	(Optional) Specify a period of time in seconds to delay communicating state changes of a tracked object. The range is from 1 to 180 seconds.
6.	**end**	Return to privileged EXEC mode.
7.	**show track** *object-number*	Verify that the specified objects are being tracked.
8.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Use the **no track** *track-number* global configuration command to delete the tracked list.

This example configures tracked list 4 with three objects and a specified percentages to measure the state of the list:

```
Switch(config)# track 4 list threshold percentage
Switch(config-track)# object 1
Switch(config-track)# object 2
Switch(config-track)# object 3
Switch(config-track)# threshold percentage up 51 down 10
Switch(config-track)# exit
```

Configuring HSRP Object Tracking

Beginning in privileged EXEC mode, follow these steps to configure a standby HSRP group to track an object and change the HSRP priority based on the object state:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**track** *object-number* {**interface** *interface-id* {**line-protocol** | **ip routing}** | **ip route** *ip-address/prefix-length* {**metric threshold** | **reachability**} | **list** {**boolean** {**and** | **or**}} | {**threshold** {**weight** | **percentage**}}}	(Optional) Create a tracking list to track the configured state and enter tracking configuration mode. ■ The *object-number* range is from 1 to 500. ■ Enter **interface** *interface-id* to select an interface to track. ■ Enter **line-protocol** to track the interface line protocol state or enter **ip routing to** track the interface IP routing state. ■ Enter **ip route** *ip-address/prefix-length* to track the state of an IP route. ■ Enter **metric threshold** to track the threshold metric or enter **reachability** to track if the route is reachable. The default up threshold is 254 and the default down threshold is 255. ■ Enter **list** to track objects grouped in a list. Configure the list as described on the previous pages. – For **boolean**, see Configuring a Tracked List with a Boolean Expression, page 1019 – For **threshold weight**, see Configuring a Tracked List with a Weight Threshold, page 1020 – For **threshold percentage**, see Configuring a Tracked List with a Percentage Threshold, page 1021 **Note:** Repeat this step for each interface to be tracked.
3.	**exit**	Return to global configuration mode.
4.	**interface** *interface-id*	Enter interface configuration mode.
5.	**standby** [*group-number*] **ip** [*ip-address* [**secondary**]]	Create (or enable) the HSRP group by using its number and virtual IP address. ■ (Optional) *group-number*–Enter a group number on the interface for which HSRP is being enabled. The range is 0 to 255; the default is 0. If there is only one HSRP group, you do not need to enter a group number. ■ (Optional on all but one interface) *ip-address*–Specify the virtual IP address of the hot standby router interface. You must enter the virtual IP address for at least one of the interfaces; it can be learned on the other interfaces. ■ (Optional) **secondary**–Specify that the IP address is a secondary hot standby router interface. If this keyword is omitted, the configured address is the primary IP address.

	Command	Purpose
6.	**standby** [*group-number*] **track** *object-number* [**decrement** [*priority-decrement*]]	Configure HSRP to track an object and change the hot standby priority based on the state of the object. ■ (Optional) *group-number*–Enter the group number to which the tracking applies. ■ *object-number*–Enter a number representing the object to be tracked. The range is from 1 to 500; the default is 1. ■ (Optional) **decrement** *priority-decrement*–Specify the amount by which the hot standby priority for the router is decremented (or incremented) when the tracked object goes down (or comes back up). The range is from 1 to 255; the default is 10.
7.	**end**	Return to privileged EXEC mode.
8.	**show standby**	Verify the standby router IP address and tracking states.
9.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Configuring Other Tracking Characteristics

You can also use the enhanced object tracking for tracking other characteristics.

- You can track the reachability of an IP route by using the **track ip route reachability** global configuration command**.**
- You can use the **track ip route metric threshold** global configuration command to determine if a route is above or below threshold.
- You can use the **track resolution** global configuration command to change the metric resolution default values for routing protocols.
- You can use the **track timer** tracking configuration command to configure the tracking process to periodically poll tracked objects.

Use the **show track** privileged EXEC command to verify enhanced object tracking configuration.

For more information about enhanced object tracking and the commands used to configure it, see this URL:

http://www.cisco.com/en/US/docs/ios/12_2t/12_2t15/feature/guide/fthsrptk.html

Configuring IP SLAs Object Tracking

Cisco IOS IP Service Level Agreements (IP SLAs) is a network performance measurement and diagnostics tool that uses active monitoring by generating traffic to measure network performance. Cisco IP SLAs operations collects real-time metrics that you can use for network troubleshooting, design, and analysis.

For IP SLAs command information see the *Cisco IOS IP SLAs Command Reference Guide, Release 12.4T* at this URL:

http://www.cisco.com/en/US/docs/ios/ipsla/configuration/guide/12_4t/sla_12_4t_book.html

Object tracking of IP SLAs operations allows clients to track the output from IP SLAs objects and use this information to trigger an action. Every IP SLAs operation maintains an SNMP operation return-code value, such as *OK* or *OverThreshold*, that can be interpreted by the tracking process. You can track two aspects of IP SLAs operation: state and reachability. For state, if the return code is OK, the track state is up; if the return code is not OK, the track state is down. For reachability, if the return code is OK or OverThreshold, reachability is up; if not OK, reachability is down.

Beginning in privileged EXEC mode, follow these steps to track the state of an IP SLAs operation or the reachability of an IP SLAs IP host:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**track** *object-number* **rtr** *operation-number* **state**	Enter tracking configuration mode to track the state of an IP SLAs operation. ■ The *object-number* range is from 1 to 500. ■ The *operation-number* range is from 1 to 2147483647.
3.	**delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}	(Optional) Specify a period of time in seconds to delay communicating state changes of a tracked object. The range is from 1 to 180 seconds.
4.	**exit**	Return to global configuration mode.
5.	**track** *object-number* **rtr** *operation-number* **reachability**	Enter tracking configuration mode to track the reachability of an IP SLAs IP host. ■ The *object-number* range is from 1 to 500. ■ The *operation-number* range is from 1 to 2147483647.
6.	**delay** {**up** *seconds* [**down** *seconds*] | [**up** *seconds*] **down** *seconds*}	(Optional) Specify a period of time in seconds to delay communicating state changes of a tracked object. The range is from 1 to 180 seconds.
7.	**end**	Return to privileged EXEC mode.
8.	**show track** *object-number*	Display tracking information to verify the configuration.
9.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

This example shows how to configure and display IP SLAs state tracking:

```
Switch(config)# track 2 200 state
Switch(config)# end
Switch# show track 2
Track 2
  Response Time Reporter 1 state
  State is Down
    1 change, last change 00:00:47
  Latest operation return code: over threshold
  Latest RTT (millisecs) 4
  Tracked by:
    HSRP Ethernet0/1 3
```

This example output shows whether a route is reachable:

```
Switch(config)# track 3 500 reachability
Switch(config)# end
Switch# show track 3
Track 3
  Response Time Reporter 1 reachability
  Reachability is Up
    1 change, last change 00:00:47
  Latest operation return code: over threshold
  Latest RTT (millisecs) 4
  Tracked by:
    HSRP Ethernet0/1 3
```

Configuring Static Routing Support

Static routing support using enhanced object tracking provides the ability for the switch to use ICMP pings to identify when a preconfigured static route or a DHCP route goes down. When tracking is enabled, the system tracks the state of the route and informs the client when that state changes. Static route object tracking uses Cisco IP SLAs to generate ICMP pings to monitor the state of the connection to the primary gateway.

- For more information about Cisco IP SLAs support on the switch, see Configuring Cisco IOS IP SLAs Operations, page 1001
- For more information about static route object tracking, see this URL:

  http://www.cisco.com/en/US/docs/ios/12_3/12_3x/12_3xe/feature/guide/dbackupx.html

You use this process to configure static route object tracking:

1. Configure a primary interface for static routing or for DHCP.
2. Configure an IP SLAs agent to ping an IP address using a primary interface and a track object to monitor the state of the agent.
3. Configure a default static default route using a secondary interface. This route is used only if the primary route is removed.

Configuring a Primary Interface

Beginning in privileged EXEC mode, follow these steps to configure a primary interface for static routing:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Select a primary or secondary interface and enter interface configuration mode.
3.	**description** *string*	Add a description to the interface.
4.	**ip address** *ip-address mask* [**secondary**]	Set the primary or secondary IP address for the interface.
5.	**exit**	Return to global configuration mode.

Beginning in privileged EXEC mode, follow these steps to configure a primary interface for DHCP:

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	Select a primary or secondary interface and enter interface configuration mode.
3.	**description** *string*	Add a description to the interface.
4.	**ip dhcp client route track** *number*	Configure the DCHP client to associate any added routes with the specified track number. Valid numbers are from 1 to 500.
5.	**ip address dhcp**	Acquire an IP address on an Ethernet interface from DHCP.
6.	**exit**	Return to global configuration mode.

Configuring a Cisco IP SLAs Monitoring Agent and Track Object

Beginning in privileged EXEC mode, follow these steps to configure network monitoring with Cisco IP SLAs:

1.	**configure terminal**	Enter global configuration mode.
2.	**ip sla** *operation-number*	Begin configuring a Cisco IP SLAs operation and enter IP SLA configuration mode.
3.	**icmp-echo** {*destination-ip-address* | *destination hostname* [**source- ipaddr** {*ip-address* | *hostname* **source-interface** *interface-id*]	Configure a Cisco IP SLAs end-to-end ICMP echo response time operation and enter IP SLAs ICMP echo configuration mode.
4.	**timeout** *milliseconds*	Set the amount of time for which the operation waits for a response from its request packet.
5.	**frequency** *seconds*	Set the rate at which the operation is sent into the network.
6.	**threshold** *milliseconds*	Set the rising threshold (hysteresis) that generates a reaction event and stores history information for the operation.
7.	**exit**	Exit IP SLAs ICMP echo configuration mode.
8.	**ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] **start-time** *time* | **pending** | **now** | **after** *time*] [**ageout** *seconds*] [**recurring**]	Configure the scheduling parameters for a single IP SLAs operation.
9. {	**track** *object-number* **rtr** *operation-number* {**state** | **reachability**}	Track the state of a Cisco IOS IP SLAs operation and enter tracking configuration mode.
10.	**end**	Return to privileged EXEC mode.
11.	**show track** *object-number*	Display tracking information to verify the configuration.
12.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Configuring a Routing Policy and Default Route

Beginning in privileged EXEC mode, follow these steps to configure a routing policy for backup static routing by using object tracking. For more details about the commands in the procedure, see this URL:

http://www.cisco.com/en/US/docs/ios/12_3/12_3x/12_3xe/feature/guide/dbackupx.html

1.	**configure terminal**	Enter global configuration mode.
2.	**access-list** *access-list-number*	Define an extended IP access list. Configure any optional characteristics.
3.	**route-map** *map-tag* [**permit | deny**] [*sequence-number*]	Enter route-map configuration mode and define conditions for redistributing routes from one routing protocol to another.
4.	**match ip address {***access-list number* | *access-list name*}	Distribute any routes that have a destination network number address that is permitted by a standard or extended access list or performs policy routing on packets. You can enter multiple numbers or names.
5.	**set ip next-hop dynamic dhcp**	For DHCP networks only. Set the next hop to the gateway that was most recently learned by the DHCP client.
6.	**set interface** *interface-id*	For static routing networks only. Indicate where to send output packets that pass a match clause of a route map for policy routing.
7.	**exit**	Exit route-map configuration mode.
8.	**ip local policy route-map** *map-tag*	Identify a route map to use for local policy routing.

9. {	**ip route** *prefix mask* {*ip-address* | *interface-id* [*ip-address*]} [*distance*] [*name*] [**permanent | track** *track-number*] [*tag tag*]	For static routing networks only. Establish static routes. Entering **track** *track-number* specifies that the static route is installed only if the configured track object is up.
10.	**end**	Return to privileged EXEC mode.
11.	**show ip route track table**	Display information about the IP route track table.
12.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

For configuration examples, see this URL:

http://www.cisco.com/en/US/docs/ios/12_3/12_3x/12_3xe/feature/guide/dbackupx.html

Monitoring Enhanced Object Tracking

Use the following privileged EXEC or User EXEC commands ito display enhanced object tracking information.

Command	Purpose
show ip route track table	Display information about the IP route track table.
show track [*object-number*]	Display information about the all tracking lists or the specified list.
show track brief	Display a single line of tracking information output.
show track interface [**brief**]	Display information about tracked interface objects.
show track ip [*object-number*] [**brief**] **route**	Display information about tracked IP-route objects.
show track resolution	Display the resolution of tracked parameters.
show track timers	Display tracked polling interval timers.

Configuring MODBUS TCP

Understanding MODBUS TCP

Use Modicon Communication Bus (MODBUS) TCP over an Ethernet network when connecting the switch to devices such as intelligent electronic devices (IEDs), distributed controllers, substation routers, Cisco IP Phones, Cisco Wireless Access Points, and other network devices such as redundant substation switches.

MODBUS is a serial communications protocol for client-server communication between a switch (server) and a device in the network running MODBUS client software (client). You can use MODBUS to connect a computer to a remote terminal unit (RTU) in supervisory control and data acquisition (SCADA) systems.

The client can be an IED or a human machine interface (HMI) application that remotely configure and manage devices running MODBUS TCP. The switch functions as the server.

The switch encapsulates a request or response message in a MODBUS TCP application data unit (ADU). A client sends a message to a TCP port on the switch. The default port number is 502.

MODBUS and Security

If a firewall or other security services are enabled, the switch TCP port might be blocked, and the switch and the client cannot communicate.

If a firewall and other security services are disabled, a denial-of-service attack might occur on the switch.

- To prevent a denial-of-service attack and to allow a specific client to send messages to the switch (server), you can use this standard access control list (ACL) that permits traffic only from the source IP address *10.1.1.n*:

```
interface Ethernet0/0
ip address 10.1.1.1 255.255.255.0
ip access-group 1 in
!
access-list 1 permit 10.1.1.0 0.0.0.255
```

- To configure quality of service (QoS) to set the rate-limit for MODBUS TCP traffic:

```
interface FastEthernet0/1
ip address 10.1.1.1 255.255.255.0
ip access-group 1 in
 rate-limit input access-group 101 8000 8000 8000 conform-action transmit exceed-action drop
```

```
!
access-list 101 permit tcp 10.1.1.0 0.0.0.255 any eq 502
```

Multiple Request Messages

The switch can receive multiple request messages from clients and respond to them simultaneously.

You can set the number of client connections from 1 to 5. The default is 1.

Configuring the Switch as the MODBUS TCP Server

Defaults

The switch is not configured as a MODBUS TCP server.

The TCP switch port number is 502.

The number of simultaneous connection requests is 1.

Enabling MODBUS TCP on the Switch

Beginning in privileged EXEC mode:

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**scada modbus tcp server**	Enables MODBUS TCP on the switch
3.	**scada modbus tcp server port** *tcp-port-number*	(Optional) Sets the TCP port to which clients send messages. The range for *tcp-port-number* is 1 to 65535. The default is 502.
4.	**scada modbus tcp server connection** *connection-requests*	(Optional) Sets the number of simultaneous connection requests sent to the switch. The range for *connection-requests* is 1 to 5. The default is 1.
5.	**end**	Returns to privileged EXEC mode.
6.	**show scada modbus tcp server**	Displays the server information and statistics.
7.	**copy running-config startup config**	(Optional) Saves your entries in the configuration file.

To disable MODBUS on the switch and return to the default settings, enter the **no scada modbus tcp server** global configuration command.

To clear the server and client statistics, enter the **clear scada modbus tcp server statistics** privileged EXEC command.

After you enable MODBUS TCP on the switch, this warning appears:

```
WARNING: Starting Modbus TCP server is a security risk.
Please understand the security issues involved before
proceeding further. Do you still want to start the
server? [yes/no]:
```

To add security when using MODBUS TCP, configure an ACL to permit traffic from specific clients or configure QoS to rate-limit traffic.

Displaying MODBUS TCP Information

Command	Purpose
show scada modbus tcp server	Displays the server information and statistics.
show scada modbus tcp server connections	Displays the client information and statistics.

Ethernet CFM

Cisco Industrial Ethernet switches supports Ethernet CFM. Ethernet CFM is an end-to-end per-service-instance (per VLAN) Ethernet layer OAM protocol that includes proactive connectivity monitoring, fault verification, and fault isolation. End-to-end can be provider-edge-to-provider-edge (PE-to-PE) device or customer-edge-to-customer-edge (CE-to-CE) device. Ethernet CFM, as specified by 802.1ag, is the standard for Layer 2 ping, Layer 2 traceroute, and end-to-end connectivity check of the Ethernet network.

For complete command and configuration information for Ethernet CFM, see the Configuring Ethernet OAM, CFM, and E-LMI chapter of the System Management guide at this URL:
http://www.cisco.com/c/en/us/td/docs/switches/connectedgrid/cg-switch-sw-master/software/configuration/guide/sysmgmt/CGS_1000_Sysmgmt/sm_oam.html

Working with the Cisco IOS File System, Configuration Files, and Software Images

This document describes how to manipulate the switch flash file system, how to copy configuration files, and how to archive (upload and download) software images to a switch.

Note: For complete syntax and usage information for the commands used in this chapter, see the *Cisco IOS Configuration Fundamentals Command Reference, Release 15.0* from the Cisco.com page.

Working with the Flash File System

The flash file system is a single flash device on which you can store files. It also provides several commands to help you manage software image and configuration files. The default flash file system on the switch is named *flash:*.

The switch has a removable compact flash card that stores the Cisco IOS software image and configuration files. Removing the compact flash card does not interrupt switch operation unless you need to reload the Cisco IOS software. However, if you remove the compact flash card, you do not have access to the flash file system, and any attempt to access it generates an error message.

Use the **show flash:** privileged EXEC command to display the compact flash file settings. For more information about the command, go to this URL:

https://www.cisco.com/c/en/us/td/docs/ios/fundamentals/command/reference/cf_book.html

For information about how to remove or replace the compact flash memory card on the switch, see the *Hardware Installation Guide Hardware Technical Guide*.

Displaying Available File Systems

To display the available file systems on your switch, use the **show file systems** privileged EXEC command as shown in this example.

```
Switch# show file systems

File Systems:

     Size(b)     Free(b)      Type  Flags  Prefixes
           -           -    opaque     ro   bs:
*  134086656   117346304     flash     rw   flash:
           -           -    opaque     rw   system:
           -           -    opaque     rw   tmpsys:
      524288      518334     nvram     rw   nvram:
           -           -    opaque     ro   xmodem:
           -           -    opaque     ro   ymodem:
           -           -    opaque     rw   null:
           -           -    opaque     ro   tar:
           -           -   network     rw   tftp:
           -           -   network     rw   rcp:
           -           -   network     rw   http:
           -           -   network     rw   ftp:
```

```
          -           -   network     rw   scp:
          -           -   network     rw   https:
          -           -    opaque     ro   cns:

Switch#
```

Detecting an Unsupported SD Flash Memory Card

When the switch starts and detects an unsupported Secure Digital (SD) flash memory card, or when you insert an unsupported SD flash memory card while the switch is running, the following warning message is displayed:

```
WARNING: Non-IT SD flash detected. Use of this card during normal
         operation can impact and severely degrade performance of the system.
          Please use supported SD flash cards only.
```

To display information about the SD flash memory card on the screen, use the **show platform sdflash** privileged EXEC command.

This example shows an unsupported SD flash memory card:

```
Switch# show platform sdflash

SD Flash Manufacturer      : SMART MODULAR (ID=27h) - Non IT

         Size              : 485MB

         Serial number     : B01000A5

         Revision          : 2.0

         Manufacturing date: 12/2009
```

This example shows a supported SD flash memory card:

```
Switch# show platform sdflash

SD Flash Manufacturer      : SMART MODULAR (ID=27h)

         Size                           : 972MB

         Serial number         : 07000019

         Revision                   : 2.0

         Manufacturing date: 3/2010
```

Note: When you enter the **show platform sdflash** privileged EXEC command, the name, date, and other fields that are displayed depend on the manufacturer of the SD flash memory card. However, if the SD flash memory card is unsupported, “Non IT” is displayed after the manufacturer’s name.

Note: The output of the **show platform sdflash** privileged EXEC command is also included in the **show tech-support** privileged EXEC command output.

SD Flash Memory Card LED

Color	System Status
Off / blinking green	SD flash memory card transfer in progress.
Slow blinking amber	SD flash memory card is unsupported.
Fast blinking amber	SD flash memory card is not present.
Amber	Error accessing the SD flash memory card. Cisco IOS boot image cannot be found.
Green	SD flash memory card is functioning.

Setting the Default File System

Table 67 show file systems Field Descriptions

Field	Value
Size(b)	Amount of memory in the file system in bytes.
Free(b)	Amount of free memory in the file system in bytes.
Type	Type of file system. **flash**–The file system is for a flash memory device. **nvram**–The file system is for a NVRAM device. **opaque**–The file system is a locally generated *pseudo* file system (for example, the *system*) or a download interface, such as brimux. **unknown**–The file system is an unknown type.
Flags	Permission for file system. **ro**–read-only. **rw**–read/write.\ **wo**–write-only.
Prefixes	Alias for file system. **flash:**–Flash file system. **nvram:**–NVRAM. **null:**–Null destination for copies. You can copy a remote file to null to find its size. **rcp:**–Remote Copy Protocol (RCP) network server. **system:**–Contains the system memory, including the running configuration. **tftp:**–TFTP network server. **xmodem:**–Obtain the file from a network machine by using the Xmodem protocol. **ymodem:**–Obtain the file from a network machine by using the Ymodem protocol.

You can specify the file system or directory that the system uses as the default file system by using the **cd** *filesystem:*

privileged EXEC command. You can set the default file system to omit the *filesystem:* argument from related commands. For example, for all privileged EXEC commands that have the optional *filesystem:* argument, the system uses the file system specified by the **cd** command.

By default, the default file system is *flash:*.

You can display the current default file system as specified by the **cd** command by using the **pwd** privileged EXEC command.

Displaying Information About Files on a File System

You can view a list of the contents of a file system before manipulating its contents. For example, before copying a new configuration file to flash memory, you might want to verify that the file system does not already contain a configuration file with the same name. Similarly, before copying a flash configuration file to another location, you might want to verify its filename for use in another command.

To display information about files on a file system, use one of the privileged EXEC commands in Table 68 on page 1038.

Table 68 Commands for Displaying Information About Files

Command	Description
dir [**/all**] [*filesystem***:**][*filename*]	Display a list of files on a file system.
show file systems	Display more information about each of the files on a file system.
show file information *file-url*	Display information about a specific file.
show file descriptors	Display a list of open file descriptors. File descriptors are the internal representations of open files. You can use this command to see if another user has a file open.

Changing Directories and Displaying the Working Directory

Beginning in privileged EXEC mode, follow these steps to change directories and display the working directory:

	Command	Purpose
1.	**dir** *filesystem***:**	Displays the directories on the specified file system. For *filesystem***:**, use **flash:** for the system board flash device.
2.	**cd new_configs**	Changes to the directory of interest. The command example shows how to change to the directory named *new_configs*.
3.	**pwd**	Displays the working directory.

Creating and Removing Directories

Beginning in privileged EXEC mode, follow these steps to create and remove a directory:

	Command	Purpose
1.	**dir** *filesystem***:**	Displays the directories on the specified file system. For *filesystem***:**, use **flash:** for the system board flash device.
2.	**mkdir old_configs**	Creates a new directory. The command example shows how to create the directory named *old_configs*. Directory names are case sensitive. Directory names are limited to 45 characters between the slashes (/); the name cannot contain control characters, spaces, deletes, slashes, quotes, semicolons, or colons.
3.	**dir** *filesystem***:**	Verifies your entry.

To delete a directory with all its files and subdirectories, use the **delete /force /recursive** *filesystem***:/***file-url* privileged EXEC command.

Use the **/recursive** keyword to delete the named directory and all subdirectories and the files contained in it. Use the **/force** keyword to suppress the prompting that confirms a deletion of each file in the directory. You are prompted only once at the beginning of this deletion process. Use the **/force** and **/recursive** keywords for deleting old software images that were installed by using the **archive download-sw** command but are no longer needed.

For *filesystem*, use **flash:** for the system board flash device. For *file-url*, enter the name of the directory to be deleted. All the files in the directory and the directory are removed.

Caution: When files and directories are deleted, their contents cannot be recovered.

Copying Files

To copy a file from a source to a destination, use the **copy** *source-url destination-url* privileged EXEC command. For the source and destination URLs, you can use **running-config** and **startup-config** keyword shortcuts. For example, the **copy running-config startup-config** command saves the currently running configuration file to the NVRAM section of flash memory to be used as the configuration during system initialization.

You can also copy from special file systems (**xmodem:**, **ymodem:**) as the source for the file from a network machine that uses the Xmodem or Ymodem protocol.

Network file system URLs include **ftp:**, **rcp:**, and **tftp:** and have these syntaxes:

- FTP—**ftp:**[[**//***username* [**:***password*]**@***location*]**/***directory*]**/***filename*
- RCP—**rcp:**[[**//***username***@***location*]**/***directory*]**/***filename*
- TFTP—**tftp:**[[**//***location*]**/***directory*]**/***filename*

Local writable file systems include flash:.

Some invalid combinations of source and destination exist. Specifically, you cannot copy these combinations:

- From a running configuration to a running configuration
- From a startup configuration to a startup configuration
- From a device to the same device (for example, the **copy flash: flash:** command is invalid)

For specific examples of using the **copy** command with configuration files, see Working with Configuration Files, page 1042.

To copy software images either by downloading a new version or by uploading the existing one, use the **archive download-sw** or the **archive upload-sw** privileged EXEC command. For more information, see Working with Software Images, page 1055.

Deleting Files

When you no longer need a file on a flash memory device, you can permanently delete it. To delete a file or directory from a specified flash device, use the **delete** [**/force**] [**/recursive**] [*filesystem***:**]**/***file-url* privileged EXEC command.

Use the **/recursive** keyword for deleting a directory and all subdirectories and the files contained in it. Use the **/force** keyword to suppress the prompting that confirms a deletion of each file in the directory. You are prompted only once at the beginning of this deletion process. Use the **/force** and **/recursive** keywords for deleting old software images that were installed by using the **archive download-sw** command but are no longer needed.

If you omit the *filesystem***:** option, the switch uses the default device specified by the **cd** command. For *file-url*, you specify the path (directory) and the name of the file to be deleted.

When you attempt to delete any files, the system prompts you to confirm the deletion.

Caution: When files are deleted, their contents cannot be recovered.

This example shows how to delete the file *myconfig* from the default flash memory device:

```
Switch# delete myconfig
```

Creating, Displaying, and Extracting tar Files

You can create a tar file and write files into it, list the files in a tar file, and extract the files from a tar file as described in the next sections.

Note: Instead of using the **copy** privileged EXEC command or the **archive tar** privileged EXEC command, we recommend using the **archive download-sw** and **archive upload-sw** privileged EXEC commands to download and upload software image files.

Creating a tar File

To create a tar file and write files into it, use this privileged EXEC command:

archive tar /create *destination-url* **flash:/***file-url*

For *destination-url*, specify the destination URL alias for the local or network file system and the name of the tar file to create. These options are supported:

- For the local flash file system, the syntax is
  flash:
- For the FTP, the syntax is
  ftp:[[**//***username*[**:***password*]**@***location*]**/***directory*]**/***tar-filename***.tar**
- For the RCP, the syntax is
  rcp:[[**//***username***@***location*]**/***directory*]**/***tar-filename***.tar**
- For the TFTP, the syntax is
  tftp:[[**//***location*]**/***directory*]**/***tar-filename***.tar**

The *tar-filename***.tar** is the tar file to be created.

For **flash:/***file-url*, specify the location on the local flash file system from which the new tar file is created. You can also specify an optional list of files or directories within the source directory to write to the new tar file. If none are specified, all files and directories at this level are written to the newly created tar file.

This example shows how to create a tar file. This command writes the contents of the *new-configs* directory on the local flash device to a file named *saved.tar* on the TFTP server at 172.20.10.30:

```
Switch# archive tar /create tftp:172.20.10.30/saved.tar flash:/new-configs
```

Displaying the Contents of a tar File

To display the contents of a tar file on the screen, use this privileged EXEC command:

archive tar /table *source-url*

For *source-url*, specify the source URL alias for the local or network file system. These options are supported:

- For the local flash file system, the syntax is
  flash:
- For the FTP, the syntax is
  ftp:[[**//***username*[**:***password*]**@***location*]**/***directory*]**/***tar-filename***.tar**
- For the RCP, the syntax is
  rcp:[[**//***username***@***location*]**/***directory*]**/***tar-filename***.tar**
- For the TFTP, the syntax is
  tftp:[[**//***location*]**/***directory*]**/***tar-filename***.tar**

The *tar-filename***.tar** is the tar file to display.

You can also limit the display of the files by specifying an optional list of files or directories after the tar file; then only those files appear. If none are specified, all files and directories appear.

This example shows how to display the contents of a switch tar file that is in flash memory:

```
Switch# archive tar /table flash:image-name.tar
image-name/ (directory)
image-name/html/ (directory)
image-name/html/file.html (0 bytes)
image-name/image-name.bin (610856 bytes)
image-name/info (219 bytes)
```

This example shows how to display only the */html* directory and its contents:

```
Switch# archive tar /table flash: image-name/html
cimage-name/html
cimage-name/html/ (directory)
cimage-name/html/const.htm (556 bytes)
cimage-name/html/xhome.htm (9373 bytes)
cimage-name/html/menu.css (1654 bytes)
<output truncated>
```

Extracting a tar File

To extract a tar file into a directory on the flash file system, use this privileged EXEC command:

archive tar /xtract *source-url* **flash:/***file-url* [*dir/file*...]

For *source-url*, specify the source URL alias for the local file system. These options are supported:

- For the local flash file system, the syntax is
  flash:

- For the FTP, the syntax is
  ftp:[[**//***username*[**:***password*]**@***location*]**/***directory*]**/***tar-filename***.tar**

- For the RCP, the syntax is
  rcp:[[**//***username***@***location*]**/***directory*]**/***tar-filename***.tar**

- For the TFTP, the syntax is
  tftp:[[**//***location*]**/***directory*]**/***tar-filename***.tar**

The *tar-filename***.tar** is the tar file from which to extract files.

For **flash:/***file-url* [*dir/file*...], specify the location on the local flash file system into which the tar file is extracted. Use the *dir/file*... option to specify an optional list of files or directories within the tar file to be extracted. If none are specified, all files and directories are extracted.

This example shows how to extract the contents of a tar file located on the TFTP server at 172.20.10.30. This command extracts just the *new-configs* directory into the root directory on the local flash file system. The remaining files in the *saved.tar* file are ignored.

```
Switch# archive tar /xtract tftp://172.20.10.30/saved.tar flash:/new-configs
```

Displaying the Contents of a File

To display the contents of any readable file, including a file on a remote file system, use the **more** [**/ascii** | **/binary** | **/ebcdic**] *file-url* privileged EXEC command:.

This example shows how to display the contents of a configuration file on a TFTP server:

```
Switch# more tftp://serverA/hampton/savedconfig
!
! Saved configuration on server
!
version 11.3
service timestamps log datetime localtime
service linenumber
service udp-small-servers
service pt-vty-logging
!
<output truncated>
```

Working with Configuration Files

This section describes how to create, load, and maintain configuration files.

Configuration files contain commands entered to customize the function of the Cisco IOS software. A way to create a basic configuration file is to use the **setup** program or to enter the **setup** privileged EXEC command. For more information, see Performing Switch Setup Configuration, page 59

You can copy (*download*) configuration files from a TFTP, FTP, or RCP server to the running configuration or startup configuration of the switch. You might want to perform this for one of these reasons:

- To restore a backed-up configuration file.

- To use the configuration file for another switch. For example, you might add another switch to your network and want it to have a configuration similar to the original switch. By copying the file to the new switch, you can change the relevant parts rather than recreating the whole file.

- To load the same configuration commands on all the switches in your network so that all the switches have similar configurations.

You can copy (*upload*) configuration files from the switch to a file server by using TFTP, FTP, or RCP. You might perform this task to back up a current configuration file to a server before changing its contents so that you can later restore the original configuration file from the server.

The protocol you use depends on which type of server you are using. The FTP and RCP transport mechanisms provide faster performance and more reliable delivery of data than TFTP. These improvements are possible because FTP and RCP are built on and use the TCP/IP stack, which is connection-oriented.

Guidelines for Creating and Using Configuration Files

Creating configuration files can aid in your switch configuration. Configuration files can contain some or all of the commands 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.

Use these guidelines when creating a configuration file:

- We recommend that you connect through the console port for the initial configuration of the switch. If you are accessing the switch through a network connection instead of through a direct connection to the console port, keep in mind that some configuration changes (such as changing the switch IP address or disabling ports) can cause a loss of connectivity to the switch.
- If no password has been set on the switch, we recommend that you set one by using the **enable secret** *secret-password* global configuration command.

Note: The **copy** {**ftp: | rcp: | tftp:**} **system:running-config** privileged EXEC command loads the configuration files on the switch as if you were entering the commands at the command line. The switch does not erase the existing running configuration before adding the commands. If a command in the copied configuration file replaces a command in the existing configuration file, the existing command is erased. For example, if the copied configuration file contains a different IP address in a particular command than the existing configuration, the IP address in the copied configuration is used. However, some commands in the existing configuration might not be replaced or negated. In this case, the resulting configuration file is a mixture of the existing configuration file and the copied configuration file, with the copied configuration file having precedence.

To restore a configuration file to an exact copy of a file stored on a server, copy the configuration file directly to the startup configuration (by using the **copy** {**ftp: | rcp: | tftp:**} **nvram:startup-config** privileged EXEC command), and reload the switch.

Configuration File Types and Location

Startup configuration files are used during system startup to configure the software. Running configuration files contain the current configuration of the software. The two configuration files can be different. For example, you might want to change the configuration for a short time period rather than permanently. In this case, you would change the running configuration but not save the configuration by using the **copy running-config startup-config** privileged EXEC command.

The running configuration is saved in DRAM; the startup configuration is stored in the NVRAM section of flash memory.

Creating a Configuration File By Using a Text Editor

When creating a configuration file, you must list commands logically so that the system can respond appropriately. This is one method of creating a configuration file:

1. Copy an existing configuration from a switch to a server.

For more information, see Downloading the Configuration File By Using TFTP, page 1044, the Downloading a Configuration File By Using FTP, page 1046, or Downloading a Configuration File By Using RCP, page 1049.

2. Open the configuration file in a text editor, such as vi or emacs on UNIX or Notepad on a PC.
3. Extract the portion of the configuration file with the desired commands, and save it in a new file.
4. Copy the configuration file to the appropriate server location. For example, copy the file to the TFTP directory on the workstation (usually /tftpboot on a UNIX workstation).
5. Make sure the permissions on the file are set to world-read.

Copying Configuration Files By Using TFTP

You can configure the switch by using configuration files you create, download from another switch, or download from a TFTP server. You can copy (upload) configuration files to a TFTP server for storage.

Preparing to Download or Upload a Configuration File By Using TFTP

Before you begin downloading or uploading a configuration file by using TFTP, do these tasks:

- Ensure that the workstation acting as the TFTP server is properly configured. 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
  ```

  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). For more information on the TFTP daemon, see the documentation for your workstation.

- 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 traffic between subnets. Check connectivity to the TFTP server by using the **ping** command.
- Ensure that the configuration file to be downloaded is in the correct directory on the TFTP server (usually /tftpboot on a UNIX workstation).
- For download operations, ensure that the permissions on the file are set correctly. The permission on the file should be world-read.
- Before uploading the configuration file, you might need to create an empty file on the TFTP server. To create an empty file, enter the **touch** *filename* command, where *filename* is the name of the file you will use when uploading it to the server.
- During upload operations, if you are overwriting an existing file (including an empty file, if you had to create one) on the server, ensure that the permissions on the file are set correctly. Permissions on the file should be world-write.

Downloading the Configuration File By Using TFTP

To configure the switch by using a configuration file downloaded from a TFTP server, follow these steps:

1. Copy the configuration file to the appropriate TFTP directory on the workstation.

2. Verify that the TFTP server is properly configured by referring to the Preparing to Download or Upload a Configuration File By Using TFTP, page 1044.
3. Log into the switch through the console port or a Telnet session.
4. Download the configuration file from the TFTP server to configure the switch.

Specify the IP address or hostname of the TFTP server and the name of the file to download.

Use one of these privileged EXEC commands:

- **copy tftp:**[[[**//***location*]**/***directory*]**/***filename*] **system:running-config**
- **copy tftp:**[[[**//***location*]**/***directory*]**/***filename*] **nvram:startup-config**

The configuration file downloads, and the commands are executed as the file is parsed line-by-line.

This example shows how to configure the software from the file *tokyo-confg* at IP address 172.16.2.155:

```
Switch# copy tftp://172.16.2.155/tokyo-confg system:running-config
Configure using tokyo-confg from 172.16.2.155? [confirm] y
Booting tokyo-confg from 172.16.2.155:!!! [OK - 874/16000 bytes]
```

Uploading the Configuration File By Using TFTP

To upload a configuration file from a switch to a TFTP server for storage, follow these steps:

1. Verify that the TFTP server is properly configured by referring to the Preparing to Download or Upload a Configuration File By Using TFTP, page 1044.
2. Log into the switch through the console port or a Telnet session.
3. Upload the switch configuration to the TFTP server. Specify the IP address or hostname of the TFTP server and the destination filename.

Use one of these privileged EXEC commands:

- **copy system:running-config tftp:**[[[**//***location*]**/***directory*]**/***filename*]
- **copy nvram:startup-config tftp:**[[[**//***location*]**/***directory*]**/***filename*]

The file is uploaded to the TFTP server.

This example shows how to upload a configuration file from a switch to a TFTP server:

```
Switch# copy system:running-config tftp://172.16.2.155/tokyo-confg
Write file tokyo-confg on host 172.16.2.155? [confirm] y
#
Writing tokyo-confg!!! [OK]
```

Copying Configuration Files By Using FTP

You can copy configuration files to or from an FTP server.

The FTP protocol requires a client to send a remote username and password on each FTP request to a server. When you copy a configuration file from the switch to a server by using FTP, the Cisco IOS software sends the first valid username in this list:

- The username specified in the **copy** command if a username is specified.

- The username set by the **ip ftp username** *username* global configuration command if the command is configured.
- Anonymous.

The switch sends the first valid password in this list:

- The password specified in the **copy** command if a password is specified.
- The password set by the **ip ftp password** *password* global configuration command if the command is configured.
- The switch forms a password named *username@switchname.domain*. The variable *username* is the username associated with the current session, *switchname* is the configured hostname, and *domain* is the domain of the switch.

The username and password must be associated with an account on the FTP server. If you are writing to the server, the FTP server must be properly configured to accept your FTP write request.

Use the **ip ftp username** and **ip ftp password** commands to specify a username and password for all copies. Include the username in the **copy** command if you want to specify only a username for that copy operation.

If the server has a directory structure, the configuration file is written to or copied from the directory associated with the username on the server. For example, if the configuration file resides in the home directory of a user on the server, specify that user's name as the remote username.

For more information, see the documentation for your FTP server.

Preparing to Download or Upload a Configuration File By Using FTP

Before you begin downloading or uploading a configuration file by using FTP, do these tasks:

- Ensure that the switch has a route to the FTP server. The switch and the FTP server must be in the same subnetwork if you do not have a router to route traffic between subnets. Check connectivity to the FTP server by using the **ping** command.
- If you are accessing the switch through the console or a Telnet session and you do not have a valid username, make sure that the current FTP username is the one that you want to use for the FTP download. You can enter the **show users** privileged EXEC command to view the valid username. If you do not want to use this username, create a new FTP username by using the **ip ftp username** *username* global configuration command during all copy operations. The new username is stored in NVRAM. If you are accessing the switch through a Telnet session and you have a valid username, this username is used, and you do not need to set the FTP username. Include the username in the **copy** command if you want to specify a username for only that copy operation.
- When you upload a configuration file to the FTP server, it must be properly configured to accept the write request from the user on the switch.

For more information, see the documentation for your FTP server.

Downloading a Configuration File By Using FTP

Beginning in privileged EXEC mode, follow these steps to download a configuration file by using FTP:

	Command	Purpose
1.	Verify that the FTP server is properly configured by referring to the Preparing to Download or Upload a Configuration File By Using FTP, page 1046.	
2.	Log into the switch through the console port or a Telnet session.	
3.	**configure terminal**	Enters global configuration mode on the switch. This step is required only if you override the default remote username or password (see Steps 4, 5, and 6).
4.	**ip ftp username** *username*	(Optional) Changes the default remote username.
5.	**ip ftp password** *password*	(Optional) Changes the default password.
6.	**end**	Returns to privileged EXEC mode.
7.	**copy ftp:**[[[**//**[*username*[**:***password*]**@**]*location*]**/***directory*]**/***filename*] **system:running-config** or **copy ftp:**[[[**//**[*username*[**:***password*]**@**]*location*]**/***directory*]**/***filename*] **nvram:startup-config**	Using FTP, copies the configuration file from a network server to the running configuration or to the startup configuration file.

This example shows how to copy a configuration file named *host1-confg* from the *netadmin1* directory on the remote server with an IP address of 172.16.101.101 and to load and run those commands on the switch:

```
Switch# copy ftp://netadmin1:mypass@172.16.101.101/host1-confg system:running-config
Configure using host1-confg from 172.16.101.101? [confirm]
Connected to 172.16.101.101
Loading 1112 byte file host1-confg:![OK]
Switch#
%SYS-5-CONFIG: Configured from host1-config by ftp from 172.16.101.101
```

This example shows how to specify a remote username of *netadmin1*. The software copies the configuration file *host2-confg* from the *netadmin1* directory on the remote server with an IP address of 172.16.101.101 to the switch startup configuration.

```
Switch# configure terminal
Switch(config)# ip ftp username netadmin1
Switch(config)# ip ftp password mypass
Switch(config)# end
Switch# copy ftp: nvram:startup-config
Address of remote host [255.255.255.255]? 172.16.101.101
Name of configuration file[rtr2-confg]? host2-confg
Configure using host2-confg from 172.16.101.101?[confirm]
Connected to 172.16.101.101
Loading 1112 byte file host2-confg:![OK]
[OK]
Switch#
%SYS-5-CONFIG_NV:Non-volatile store configured from host2-config by ftp from 172.16.101.101
```

Uploading a Configuration File By Using FTP

Beginning in privileged EXEC mode, follow these steps to upload a configuration file by using FTP:

	Command	Purpose
1.	Verify that the FTP server is properly configured by referring to the Preparing to Download or Upload a Configuration File By Using FTP, page 1046.	
2.	Log into the switch through the console port or a Telnet session.	
3.	**configure terminal**	Enters global configuration mode. This step is required only if you override the default remote username or password (see Steps 4, 5, and 6).
4.	**ip ftp username** *username*	(Optional) Changes the default remote username.
5.	**ip ftp password** *password*	(Optional) Changes the default password.
6.	**end**	Returns to privileged EXEC mode.
7.	**copy system:running-config ftp:**[[[**//**[*username*[**:**_password_]**@**]*location*]**/**_directory_]**/**_filename_] or **copy nvram:startup-config ftp:**[[[**//**[*username*[**:**_password_]**@**]*location*]**/**_directory_]**/**_filename_]	Using FTP, copies the switch running or startup configuration file to the specified location.

This example shows how to copy the running configuration file named *switch2-confg* to the *netadmin1* directory on the remote host with an IP address of 172.16.101.101:

```
Switch# copy system:running-config ftp://netadmin1:mypass@172.16.101.101/switch2-confg
Write file switch2-confg on host 172.16.101.101?[confirm]
Building configuration...[OK]
Connected to 172.16.101.101
Switch#
```

This example shows how to store a startup configuration file on a server by using FTP to copy the file:

```
Switch# configure terminal
Switch(config)# ip ftp username netadmin2
Switch(config)# ip ftp password mypass
Switch(config)# end
Switch# copy nvram:startup-config ftp:
Remote host[]? 172.16.101.101
Name of configuration file to write [switch2-confg]?
Write file switch2-confg on host 172.16.101.101?[confirm]
![OK]
```

Copying Configuration Files By Using RCP

The RCP provides another method of downloading, uploading, and copying configuration files between remote hosts and the switch. Unlike TFTP, which uses User Datagram Protocol (UDP), a connectionless protocol, RCP uses TCP, which is connection-oriented.

To use RCP to copy files, the server from or to which you will be copying files must support RCP. The RCP copy commands rely on the rsh server (or daemon) on the remote system. To copy files by using RCP, you do not need to create a server for file distribution as you do with TFTP. You only need 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.

The RCP requires a client to send a remote username with each RCP request to a server. When you copy a configuration file from the switch to a server, the Cisco IOS software sends the first valid username in this list:

- The username specified in the **copy** command if a username is specified.
- The username set by the **ip rcmd remote-username** *username* global configuration command if the command is configured.
- The remote username associated with the current TTY (terminal) process. For example, if the user is connected to the router through Telnet and was authenticated through the **username** command, the switch software sends the Telnet username as the remote username.
- The switch hostname.

For a successful RCP copy request, you must define an account on the network server for the remote username. If the server has a directory structure, the configuration file is written to or copied from the directory associated with the remote username on the server. For example, if the configuration file is in the home directory of a user on the server, specify that user's name as the remote username.

Preparing to Download or Upload a Configuration File By Using RCP

Before you begin downloading or uploading a configuration file by using RCP, do these tasks:

- Ensure that the workstation acting as the RCP server supports the remote shell (rsh).
- Ensure that the switch has a route to the RCP server. The switch and the server must be in the same subnetwork if you do not have a router to route traffic between subnets. Check connectivity to the RCP server by using the **ping** command.
- If you are accessing the switch through the console or a Telnet session and you do not have 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** privileged EXEC command to view the valid username. If you do not want to use this username, create a new RCP username by using the **ip rcmd remote-username** *username* global configuration command to be used during all copy operations. The new username is stored in NVRAM. If you are accessing the switch through a Telnet session and you have a valid username, this username is used, and you do not need to set the RCP username. Include the username in the **copy** command if you want to specify a username for only that copy operation.
- When you upload a file to the RCP server, it must be properly configured to accept the RCP write request from the user on the switch. For UNIX systems, you must add an entry to the .rhosts file for the remote user on the RCP server. For example, suppose that the switch contains these configuration lines:

  ```
  hostname Switch1
  ip rcmd remote-username User0
  ```

  If the switch IP address translates to *Switch1.company.com*, the .rhosts file for User0 on the RCP server should contain this line:

  ```
  Switch1.company.com Switch1
  ```

For more information, see the documentation for your RCP server.

Downloading a Configuration File By Using RCP

Beginning in privileged EXEC mode, follow these steps to download a configuration file by using RCP:

	Command	Purpose
1.	Verify that the RCP server is properly configured by referring to the Preparing to Download or Upload a Configuration File By Using RCP, page 1049.	
2.	Log into the switch through the console port or a Telnet session.	
3.	**configure terminal**	Enters global configuration mode. This step is required only if you override the default remote username (see Steps 4 and 5).
4.	**ip rcmd remote-username** *username*	(Optional) Specifesthe remote username.
5.	**end**	Returns to privileged EXEC mode.
6.	**copy rcp:**[[[**//**[*username***@**]*location*]**/***directory*]**/***filename*] **system:running-config** or **copy rcp:**[[[**//**[*username***@**]*location*]**/***directory*]**/***filename*] **nvram:startup-config**	Using RCP, copies the configuration file from a network server to the running configuration or to the startup configuration file.

This example shows how to copy a configuration file named *host1-confg* from the *netadmin1* directory on the remote server with an IP address of 172.16.101.101 and load and run those commands on the switch:

```
Switch# copy rcp://netadmin1@172.16.101.101/host1-confg system:running-config
Configure using host1-confg from 172.16.101.101? [confirm]
Connected to 172.16.101.101
Loading 1112 byte file host1-confg:![OK]
Switch#
%SYS-5-CONFIG: Configured from host1-config by rcp from 172.16.101.101
```

This example shows how to specify a remote username of *netadmin1*. Then it copies the configuration file *host2-confg* from the *netadmin1* directory on the remote server with an IP address of 172.16.101.101 to the startup configuration:

```
Switch# configure terminal
Switch(config)# ip rcmd remote-username netadmin1
Switch(config)# end
Switch# copy rcp: nvram:startup-config
Address of remote host [255.255.255.255]? 172.16.101.101
Name of configuration file[rtr2-confg]? host2-confg
Configure using host2-confg from 172.16.101.101?[confirm]
Connected to 172.16.101.101
Loading 1112 byte file host2-confg:![OK]
[OK]
Switch#
%SYS-5-CONFIG_NV:Non-volatile store configured from host2-config by rcp from 172.16.101.101
```

Uploading a Configuration File By Using RCP

Beginning in privileged EXEC mode, follow these steps to upload a configuration file by using RCP:

	Command	Purpose
1.	Verify that the RCP server is properly configured by referring to the Preparing to Download or Upload a Configuration File By Using RCP, page 1049.	
2.	Log into the switch through the console port or a Telnet session.	
3.	**configure terminal**	Enters global configuration mode. This step is required only if you override the default remote username (see Steps 4 and 5).
4.	**ip rcmd remote-username** *username*	(Optional) Specifies the remote username.
5.	**end**	Returns to privileged EXEC mode.
6.	**copy system:running-config rcp:**[[[**//**[*username***@**]*location*]**/***directory*]**/***filename*] or **copy nvram:startup-config rcp:**[[[**//**[*username***@**]*location*]**/***directory*]**/***filename*]	Using RCP, copies the configuration file from a switch running or startup configuration file to a network server.

This example shows how to copy the running configuration file named *switch2-confg* to the *netadmin1* directory on the remote host with an IP address of 172.16.101.101:

```
Switch# copy system:running-config rcp://netadmin1@172.16.101.101/switch2-confg
Write file switch-confg on host 172.16.101.101?[confirm]
Building configuration...[OK]
Connected to 172.16.101.101
Switch#
```

This example shows how to store a startup configuration file on a server:

```
Switch# configure terminal
Switch(config)# ip rcmd remote-username netadmin2
Switch(config)# end
Switch# copy nvram:startup-config rcp:
Remote host[]? 172.16.101.101
Name of configuration file to write [switch2-confg]?
Write file switch2-confg on host 172.16.101.101?[confirm]
![OK]
```

Clearing Configuration Information

You can clear the configuration information from the startup configuration. If you reboot the switch with no startup configuration, the switch enters the setup program so that you can reconfigure the switch with all new settings.

Clearing the Startup Configuration File

To clear the contents of your startup configuration, use the **erase nvram:** or the **erase startup-config** privileged EXEC command.

Caution: You cannot restore the startup configuration file after it has been deleted.

Deleting a Stored Configuration File

To delete a saved configuration from flash memory, use the **delete flash:***filename* privileged EXEC command. Depending on the setting of the **file prompt** global configuration command, you might be prompted for confirmation before you delete a file. By default, the switch prompts for confirmation on destructive file operations. For more information about the **file prompt** command, see the *Cisco IOS Command Reference for Release 12.2*.

Caution: You cannot restore a file after it has been deleted.

Replacing and Rolling Back Configurations

The configuration replacement and rollback feature replaces the running configuration with any saved Cisco IOS configuration file. You can use the rollback function to roll back to a previous configuration.

Understanding Configuration Replacement and Rollback

Archiving a Configuration

The configuration archive provides a mechanism to store, organize, and manage an archive of configuration files. The **configure replace** privileged EXEC command increases the configuration rollback capability. As an alternative, you can save copies of the running configuration by using the **copy running-config** *destination-url* privileged EXEC command, storing the replacement file either locally or remotely. However, this method lacks any automated file management. The configuration replacement and rollback feature can automatically save copies of the running configuration to the configuration archive.

You use the **archive config** privileged EXEC command to save configurations in the configuration archive by using a standard location and filename prefix that is automatically appended with an incremental version number (and optional timestamp) as each consecutive file is saved. You can specify how many versions of the running configuration are kept in the archive. After the maximum number of files are saved, the oldest file is automatically deleted when the next, most recent file is saved. The **show archive** privileged EXEC command displays information for all the configuration files saved in the configuration archive.

The Cisco IOS configuration archive, in which the configuration files are stored and available for use with the **configure replace** command, is in any of these file systems: FTP, HTTP, RCP, TFTP.

Replacing a Configuration

The **configure replace** privileged EXEC command replaces the running configuration with any saved configuration file. When you enter the **configure replace** command, the running configuration is compared with the specified replacement configuration, and a set of configuration differences is generated. The resulting differences are used to replace the configuration. The configuration replacement operation is usually completed in no more than three passes. To prevent looping behavior no more than five passes are performed.

You can use the **copy** *source-url* **running-config** privileged EXEC command to copy a stored configuration file to the running configuration. When using this command as an alternative to the **configure replace** *target-url* privileged EXEC command, note these major differences:

- The **copy** *source-url* **running-config** command is a merge operation and preserves all the commands from both the source file and the running configuration. This command does not remove commands from the running configuration that are not present in the source file. In contrast, the **configure replace** *target-url* command removes commands from the running configuration that are not present in the replacement file and adds commands to the running configuration that are not present.

- You can use a partial configuration file as the source file for the **copy** *source-url* **running-config** command. You must use a complete configuration file as the replacement file for the **configure replace** *target-url* command.

Rolling Back a Configuration

You can also use the **configure replace** command to roll back changes that were made since the previous configuration was saved. Instead of basing the rollback operation on a specific set of changes that were applied, the configuration rollback capability reverts to a specific configuration based on a saved configuration file.

If you want the configuration rollback capability, you must first save the running configuration before making any configuration changes. Then, after entering configuration changes, you can use that saved configuration file to roll back the changes by using the **configure replace** *target-url* command.

You can specify any saved configuration file as the rollback configuration. You are not limited to a fixed number of rollbacks, as is the case in some rollback models.

Configuration Guidelines

Follow these guidelines when configuring and performing configuration replacement and rollback:

- Make sure that the switch has free memory larger than the combined size of the two configuration files (the running configuration and the saved replacement configuration). Otherwise, the configuration replacement operation fails.
- Make sure that the switch also has sufficient free memory to execute the configuration replacement or rollback configuration commands.
- Certain configuration commands, such as those pertaining to physical components of a networking device (for example, physical interfaces), cannot be added or removed from the running configuration.
  - A configuration replacement operation cannot remove the **interface** *interface-id* command line from the running configuration if that interface is physically present on the device.
  - The **interface** *interface-id* command line cannot be added to the running configuration if no such interface is physically present on the device.
- When using the **configure replace** command, you must specify a saved configuration as the replacement configuration file for the running configuration. The replacement file must be a complete configuration generated by a Cisco IOS device (for example, a configuration generated by the **copy running-config** *destination-url* command).

Note: If you generate the replacement configuration file externally, it must comply with the format of files generated by Cisco IOS devices.

Configuring the Configuration Archive

Using the **configure replace** command with the configuration archive and with the **archive config** command is optional but offers significant benefit for configuration rollback scenarios. Before using the **archive config command**, you must first configure the configuration archive. Starting in privileged EXEC mode, follow these steps to configure the configuration archive:

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**archive**	Enters archive configuration mode.
3.	**path** *url*	Specifies the location and filename prefix for the files in the configuration archive.

	Command	Purpose
4.	**maximum** *number*	(Optional) Sets the maximum number of archive files of the running configuration to be saved in the configuration archive. *number*—Maximum files of the running configuration file in the configuration archive. Valid values are from 1 to 14. The default is 10. **Note:** Before using this command, you must first enter the **path** archive configuration command to specify the location and filename prefix for the files in the configuration archive.
5.	**time-period** *minutes*	(Optional) Sets the time increment for automatically saving an archive file of the running configuration in the configuration archive. *minutes*—Specifies how often, in minutes, to automatically save an archive file of the running configuration in the configuration archive.
6.	**end**	Returns to privileged EXEC mode.
7.	**show running-config**	Verifies the configuration.
8.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Performing a Configuration Replacement or Rollback Operation

Starting in privileged EXEC mode, follow these steps to replace the running configuration file with a saved configuration file:

	Command	Purpose
1.	**archive config**	(Optional) Saves the running configuration file to the configuration archive. **Note:** Enter the **path** archive configuration command before using this command.
2.	**configure terminal**	Enters global configuration mode.
3.		Makes necessary changes to the running configuration.
4.	**exit**	Returns to privileged EXEC mode.

	Command	Purpose
5.	**configure replace** *target-url* [**list**] [**force**] [**time** *seconds*] [**nolock**]	Replaces the running configuration file with a saved configuration file. *target-url*–URL (accessible by the file system) of the saved configuration file that is to replace the running configuration, such as the configuration file created in Step 2 by using the **archive config** privileged EXEC command. **list**–Displays a list of the command entries applied by the software parser during each pass of the configuration replacement operation. The total number of passes also appears. **force**– Replaces the running configuration file with the specified saved configuration file without prompting you for confirmation. **time** *seconds*–Specifies the time (in seconds) within which you must enter the **configure confirm** command to confirm replacement of the running configuration file. If you do not enter the **configure confirm** command within the specified time limit, the configuration replacement operation is automatically stopped. (In other words, the running configuration file is restored to the configuration that existed before you entered the **configure replace** command). **Note:** You must first enable the configuration archive before you can use the **time** *seconds* command line option. **nolock**–Disables the locking of the running configuration file that prevents other users from changing the running configuration during a configuration replacement operation.
6.	**configure confirm**	(Optional) Confirms replacement of the running configuration with a saved configuration file. **Note:** Use this command only if the **time** *seconds* keyword and argument of the **configure replace** command are specified.
7.	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Working with Software Images

This section describes how to archive (download and upload) software image files, which contain the system software, the Cisco IOS code, and the embedded Device Manager software.

Note: Instead of using the **copy** privileged EXEC command or the **archive tar** privileged EXEC command, we recommend using the **archive download-sw** and **archive upload-sw** privileged EXEC commands to download and upload software image files.

You can download a switch image file from a TFTP, FTP, or RCP server to upgrade the switch software. If you do not have access to a TFTP server, you can download a software image file directly to your PC or workstation by using a web browser (HTTP) and then by using Device Manager or Cisco Network Assistant to upgrade your switch. For information about upgrading your switch by using a TFTP server or a web browser (HTTP), see the release notes.

You can replace the current image with the new one or keep the current image in flash memory after a download.

You upload a switch image file to a TFTP, FTP, or RCP server for backup purposes. You can use this uploaded image for future downloads to the same switch or to another of the same type.

The protocol that you use depends on which type of server you are using. The FTP and RCP transport mechanisms provide faster performance and more reliable delivery of data than TFTP. These improvements are possible because FTP and RCP are built on and use the TCP/IP stack, which is connection-oriented.

Note: For a list of software images and the supported upgrade paths, see the release notes.

Image Location on the Switch

The Cisco IOS image is stored as a *.bin* file in a directory that shows the version number. A subdirectory contains the files needed for web management. The image is stored on the system board flash memory (flash:).

You can use the **show version** privileged EXEC command to see the software version that is currently running on your switch. In the display, check the line that begins with `System image file is...` . It shows the directory name in flash memory where the image is stored.

You can also use the **dir** *filesystem***:** privileged EXEC command to see the directory names of other software images that might be stored in flash memory.The **archive download-sw /directory** privileged EXEC command allows you to specify a directory one time followed by a tar file or list of tar files to be downloaded instead of specifying complete paths with each tar file.

tar File Format of Images on a Server or Cisco.com

Software images located on a server or downloaded from Cisco.com are provided in a tar file format, which contains these files:

- An *info* file, which serves as a table of contents for the tar file
- One or more subdirectories containing other images and files, such as Cisco IOS images and web management files

This example shows some of the information contained in the info file. Table 69 on page 1056 provides additional details about this information:

```
system_type:0x00000000:image-name
    image_family:xxxx
    stacking_number:x
    info_end:
version_suffix:xxxx
    version_directory:image-name
    image_system_type_id:0x00000000
    image_name:image-nameB.bin
    ios_image_file_size:6398464
    total_image_file_size:8133632
    image_feature:IP|LAYER_3|PLUS|MIN_DRAM_MEG=128
    image_family:xxxx
    stacking_number:x
    board_ids:0x401100c4 0x00000000 0x00000001 0x00000003 0x00000002 0x00008000 0x00008002 0x40110000
    info_end:
```

Note: Disregard the stacking_number field. It does not apply to the switch.

Table 69 info File Description

Field	Description
version_suffix	Specifies the Cisco IOS image version string suffix.
version_directory	Specifies the directory where the Cisco IOS image and the HTML subdirectory are installed.
image_name	Specifies the name of the Cisco IOS image within the tar file.
ios_image_file_size	Specifies the Cisco IOS image size in the tar file, which is an approximate measure of how much flash memory is required to hold just the Cisco IOS image.

Table 69 info File Description (continued)

Field	Description
total_image_file_size	Specifies the size of all the images (the Cisco IOS image and the web management files) in the tar file, which is an approximate measure of how much flash memory is required to hold them.
image_feature	Describes the core functionality of the image.
image_min_dram	Specifies the minimum amount of DRAM needed to run this image.
image_family	Describes the family of products on which the software can be installed.

Copying Image Files By Using TFTP

You can download a switch image from a TFTP server or upload the image from the switch to a TFTP server.

You download a switch image file from a server to upgrade the switch software. You can overwrite the current image with the new one or keep the current image after a download.

You upload a switch image file to a server for backup purposes; this uploaded image can be used for future downloads to the same or another switch of the same type.

Note: Instead of using the **copy** privileged EXEC command or the **archive tar** privileged EXEC command, we recommend using the **archive download-sw** and **archive upload-sw** privileged EXEC commands to download and upload software image files.

Preparing to Download or Upload an Image File By Using TFTP

Before you begin downloading or uploading an image file by using TFTP, do these tasks:

- Ensure that the workstation acting as the TFTP server is properly configured. 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
  ```

  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). For more information on the TFTP daemon, see the documentation for your workstation.

- 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 traffic between subnets. Check connectivity to the TFTP server by using the **ping** command.
- Ensure that the image to be downloaded is in the correct directory on the TFTP server (usually /tftpboot on a UNIX workstation).
- For download operations, ensure that the permissions on the file are set correctly. The permission on the file should be world-read.
- Before uploading the image file, you might need to create an empty file on the TFTP server. To create an empty file, enter the **touch** *filename* command, where *filename* is the name of the file you will use when uploading the image to the server.
- During upload operations, if you are overwriting an existing file (including an empty file, if you had to create one) on the server, ensure that the permissions on the file are set correctly. Permissions on the file should be world-write.

Downloading an Image File By Using TFTP

You can download a new image file and replace the current image or keep the current image.

Beginning in privileged EXEC mode, follow Steps 1 through 3 to download a new image from a TFTP server and overwrite the existing image. To keep the current image, go to Step 3.

	Command	Purpose
1.	Copy the image to the appropriate TFTP directory on the workstation. Make sure that the TFTP server is properly configured; see the Preparing to Download or Upload an Image File By Using TFTP, page 1057.	
2.	Log into the switch through the console port or a Telnet session.	
3.	**archive download-sw /overwrite /reload tftp:**[[**//***location*]**/***directory*]**/***image-name***.tar**	Downloads the image file from the TFTP server to the switch, and overwrite the current image. ■ The **/overwrite** option overwrites the software image in flash memory with the downloaded image. ■ The **/reload** option reloads the system after downloading the image unless the configuration has been changed and not been saved. ■ For **//***location*, specify the IP address of the TFTP server. ■ For /*directory***/***image-name***.tar**, specify the directory (optional) and the image to download. Directory and image names are case sensitive.
4.	**archive download-sw /leave-old-sw /reload tftp:**[[**//***location*]**/***directory*]**/***image-name***.tar**	Downloads the image file from the TFTP server to the switch, and keep the current image. ■ The **/leave-old-sw** option keeps the old software version after a download. ■ The **/reload** option reloads the system after downloading the image unless the configuration has been changed and not been saved. ■ For **//***location*, specify the IP address of the TFTP server. ■ For /*directory***/***image-name***.tar**, specify the directory (optional) and the image to download. Directory and image names are case sensitive.

The download algorithm verifies that the image is appropriate for the switch model and that enough DRAM is present, or it aborts the process and reports an error. If you specify the **/overwrite** option, the download algorithm removes the existing image on the flash device whether or not it is the same as the new one, downloads the new image, and then reloads the software.

Note: If the flash device has sufficient space to hold two images and you want to overwrite one of these images with the same version, you must specify the **/overwrite** option.

If you specify the **/leave-old-sw**, the existing files are not removed. If there is not enough space to install the new image and keep the running image, the download process stops, and an error message is displayed.

The algorithm installs the downloaded image on the system board flash device (flash:). The image is placed into a new directory named with the software version string, and the BOOT environment variable is updated to point to the newly installed image.

If you kept the old image during the download process (you specified the **/leave-old-sw** keyword), you can remove it by entering the **delete /force /recursive** *filesystem***:/***file-url* privileged EXEC command. For *filesystem*, use **flash:** for the system board flash device. For *file-url*, enter the directory name of the old image. All the files in the directory and the directory are removed.

Caution: For the download and upload algorithms to operate properly, do *not* rename image names.

Uploading an Image File By Using TFTP

You can upload an image from the switch to a TFTP server. You can later download this image to the switch or to another switch of the same type.

Use the upload feature only if the web management pages associated with the embedded Device Manager have been installed with the existing image.

Beginning in privileged EXEC mode, follow these steps to upload an image to a TFTP server:

	Command	Purpose
1.	Make sure the TFTP server is properly configured; see the Preparing to Download or Upload an Image File By Using TFTP, page 1057.	
2.	Log into the switch through the console port or a Telnet session.	
3.	**archive upload-sw tftp:**[[**//***location*]**/***directory*]**/***image-name***.tar**	Uploads the currently running switch image to the TFTP server. ■ For **//***location*, specify the IP address of the TFTP server. ■ For /*directory***/***image-name***.tar**, specify the directory (optional) and the name of the software image to be uploaded. Directory and image names are case sensitive. The *image-name***.tar** is the name of the software image to be stored on the server.

The **archive upload-sw** privileged EXEC command builds an image file on the server by uploading these files in order: info, the Cisco IOS image, and the web management files. After these files are uploaded, the upload algorithm creates the tar file format.

Caution: For the download and upload algorithms to operate properly, do *not* rename image names.

Copying Image Files By Using FTP

You can download a switch image from an FTP server or upload the image from the switch to an FTP server.

You download a switch image file from a server to upgrade the switch software. You can overwrite the current image with the new one or keep the current image after a download.

You upload a switch image file to a server for backup purposes. You can use this uploaded image for future downloads to the switch or another switch of the same type.

Note: Instead of using the **copy** privileged EXEC command or the **archive tar** privileged EXEC command, we recommend using the **archive download-sw** and **archive upload-sw** privileged EXEC commands to download and upload software image files.

Preparing to Download or Upload an Image File By Using FTP

You can copy images files to or from an FTP server.

The FTP protocol requires a client to send a remote username and password on each FTP request to a server. When you copy an image file from the switch to a server by using FTP, the Cisco IOS software sends the first valid username in this list:

- The username specified in the **archive download-sw** or **archive upload-sw** privileged EXEC command if a username is specified.
- The username set by the **ip ftp username** *username* global configuration command if the command is configured.
- Anonymous.

The switch sends the first valid password in this list:

- The password specified in the **archive download-sw** or **archive upload-sw** privileged EXEC command if a password is specified.
- The password set by the **ip ftp password** *password* global configuration command if the command is configured.
- The switch forms a password named *username@switchname.domain*. The variable *username* is the username associated with the current session, *switchname* is the configured hostname, and *domain* is the domain of the switch.

The username and password must be associated with an account on the FTP server. If you are writing to the server, the FTP server must be properly configured to accept the FTP write request from you.

Use the **ip ftp username** and **ip ftp password** commands to specify a username and password for all copies. Include the username in the **archive download-sw** or **archive upload-sw** privileged EXEC command if you want to specify a username only for that operation.

If the server has a directory structure, the image file is written to or copied from the directory associated with the username on the server. For example, if the image file resides in the home directory of a user on the server, specify that user's name as the remote username.

Before you begin downloading or uploading an image file by using FTP, do these tasks:

- Ensure that the switch has a route to the FTP server. The switch and the FTP server must be in the same subnetwork if you do not have a router to route traffic between subnets. Check connectivity to the FTP server by using the **ping** command.
- If you are accessing the switch through the console or a Telnet session and you do not have a valid username, make sure that the current FTP username is the one that you want to use for the FTP download. You can enter the **show users** privileged EXEC command to view the valid username. If you do not want to use this username, create a new FTP username by using the **ip ftp username** *username* global configuration command. This new name will be used during all archive operations. The new username is stored in NVRAM. If you are accessing the switch through a Telnet session and you have a valid username, this username is used, and you do not need to set the FTP username. Include the username in the **archive download-sw** or **archive upload-sw** privileged EXEC command if you want to specify a username for that operation only.
- When you upload an image file to the FTP server, it must be properly configured to accept the write request from the user on the switch.

For more information, see the documentation for your FTP server.

Downloading an Image File By Using FTP

You can download a new image file and overwrite the current image or keep the current image.

Beginning in privileged EXEC mode, follow Steps 1 through 7 to download a new image from an FTP server and overwrite the existing image. To keep the current image, go to Step 7.

	Command	Purpose
1.	Verify that the FTP server is properly configured by referring to the Preparing to Download or Upload a Configuration File By Using FTP, page 1046.	
2.	Log into the switch through the console port or a Telnet session.	
3.	**configure terminal**	Enters global configuration mode. This step is required only if you override the default remote username or password (see Steps 4, 5, and 6).
4.	**ip ftp username** *username*	(Optional) Changes the default remote username.
5.	**ip ftp password** *password*	(Optional) Changes the default password.

	Command	Purpose
6.	**end**	Returns to privileged EXEC mode.
7.	**archive download-sw /overwrite /reload ftp:**[[**//***username*[**:***password*]**@***location*]**/***directory*]**/***image-name***.tar**	Downloads the image file from the FTP server to the switch, and overwrite the current image. ■ The **/overwrite** option overwrites the software image in flash memory with the downloaded image. ■ The **/reload** option reloads the system after downloading the image unless the configuration has been changed and not been saved. ■ For **//***username*[**:***password*], specify the username and password; these must be associated with an account on the FTP server. ■ For **@***location*, specify the IP address of the FTP server. ■ For *directory***/***image-name***.tar**, specify the directory (optional) and the image to download. Directory and image names are case sensitive.
8.	**archive download-sw /leave-old-sw /reload ftp:**[[**//***username*[**:***password*]**@***location*]**/***directory*]**/***image-name***.tar**	Downloads the image file from the FTP server to the switch, and keep the current image. ■ The **/leave-old-sw** option keeps the old software version after a download. ■ The **/reload** option reloads the system after downloading the image unless the configuration has been changed and not been saved. ■ For **//***username*[**:***password*], specify the username and password. These must be associated with an account on the FTP server. ■ For **@***location*, specify the IP address of the FTP server. ■ For *directory***/***image-name***.tar**, specify the directory (optional) and the image to download. Directory and image names are case sensitive.

The download algorithm verifies that the image is appropriate for the switch model and that enough DRAM is present, or it aborts the process and reports an error. If you specify the **/overwrite** option, the download algorithm removes the existing image on the flash device, whether or not it is the same as the new one, downloads the new image, and then reloads the software.

Note: If the flash device has sufficient space to hold two images and you want to overwrite one of these images with the same version, you must specify the **/overwrite** option.

If you specify the **/leave-old-sw**, the existing files are not removed. If there is not enough space to install the new image and keep the running image, the download process stops, and an error message is displayed.

The algorithm installs the downloaded image onto the system board flash device (flash:). The image is placed into a new directory named with the software version string, and the BOOT environment variable is updated to point to the newly installed image.

If you kept the old image during the download process (you specified the **/leave-old-sw** keyword), you can remove it by entering the **delete /force /recursive** *filesystem***:/***file-url* privileged EXEC command. For *filesystem*, use **flash:** for the system board flash device. For *file-url*, enter the directory name of the old software image. All the files in the directory and the directory are removed.

Caution: For the download and upload algorithms to operate properly, do *not* rename image names.

Uploading an Image File By Using FTP

You can upload an image from the switch to an FTP server. You can later download this image to the same switch or to another switch of the same type.

Use the upload feature only if the web management pages associated with the embedded Device Manager have been installed with the existing image.

Beginning in privileged EXEC mode, follow these steps to upload an image to an FTP server:

	Command	Purpose
1.	Verify that the FTP server is properly configured by referring to the Preparing to Download or Upload a Configuration File By Using FTP, page 1046.	
2.	Log into the switch through the console port or a Telnet session.	
3.	**configure terminal**	Enters global configuration mode. This step is required only if you override the default remote username or password (see Steps 4, 5, and 6).
4.	**ip ftp username** *username*	(Optional) Changes the default remote username.
5.	**ip ftp password** *password*	(Optional) Changes the default password.
6.	**end**	Returns to privileged EXEC mode.
7.	**archive upload-sw ftp:**[[**//**[*username*[**:***password*]**@**]*location*]**/***directory*]**/***image-name***.tar**	Uploads the currently running switch image to the FTP server. ■ For **//***username***:***password*, specify the username and password. These must be associated with an account on the FTP server. ■ For **@***location*, specify the IP address of the FTP server. ■ For **/***directory***/***image-name***.tar**, specify the directory (optional) and the name of the software image to be uploaded. Directory and image names are case sensitive. The *image-name***.tar** is the name of the software image to be stored on the server.

The **archive upload-sw** command builds an image file on the server by uploading these files in order: info, the Cisco IOS image, and the web management files. After these files are uploaded, the upload algorithm creates the tar file format.

Caution: For the download and upload algorithms to operate properly, do *not* rename image names.

Copying Image Files By Using RCP

You can download a switch image from an RCP server or upload the image from the switch to an RCP server.

You download a switch image file from a server to upgrade the switch software. You can overwrite the current image with the new one or keep the current image after a download.

You upload a switch image file to a server for backup purposes. You can use this uploaded image for future downloads to the same switch or another of the same type.

Note: Instead of using the **copy** privileged EXEC command or the **archive tar** privileged EXEC command, we recommend using the **archive download-sw** and **archive upload-sw** privileged EXEC commands to download and upload software image files.

Preparing to Download or Upload an Image File By Using RCP

RCP provides another method of downloading and uploading image files between remote hosts and the switch. Unlike TFTP, which uses User Datagram Protocol (UDP), a connectionless protocol, RCP uses TCP, which is connection-oriented.

To use RCP to copy files, the server from or to which you will be copying files must support RCP. The RCP copy commands rely on the rsh server (or daemon) on the remote system. To copy files by using RCP, you do not need to create a server for file distribution as you do with TFTP. You only need 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.

RCP requires a client to send a remote username on each RCP request to a server. When you copy an image from the switch to a server by using RCP, the Cisco IOS software sends the first valid username in this list:

- The username specified in the **archive download-sw** or **archive upload-sw** privileged EXEC command if a username is specified.
- The username set by the **ip rcmd remote-username** *username* global configuration command if the command is entered.
- The remote username associated with the current TTY (terminal) process. For example, if the user is connected to the router through Telnet and was authenticated through the **username** command, the switch software sends the Telnet username as the remote username.
- The switch hostname.

For the RCP copy request to execute successfully, an account must be defined on the network server for the remote username. If the server has a directory structure, the image file is written to or copied from the directory associated with the remote username on the server. For example, if the image file resides in the home directory of a user on the server, specify that user’s name as the remote username.

Before you begin downloading or uploading an image file by using RCP, do these tasks:

- Ensure that the workstation acting as the RCP server supports the remote shell (rsh).
- Ensure that the switch has a route to the RCP server. The switch and the server must be in the same subnetwork if you do not have a router to route traffic between subnets. Check connectivity to the RCP server by using the **ping** command.
- If you are accessing the switch through the console or a Telnet session and you do not have 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** privileged EXEC command to view the valid username. If you do not want to use this username, create a new RCP username by using the **ip rcmd remote-username** *username* global configuration command to be used during all archive operations. The new username is stored in NVRAM. If you are accessing the switch through a Telnet session and you have a valid username, this username is used, and there is no need to set the RCP username. Include the username in the **archive download-sw** or **archive upload-sw** privileged EXEC command if you want to specify a username only for that operation.

- When you upload an image to the RCP to the server, it must be properly configured to accept the RCP write request from the user on the switch. For UNIX systems, you must add an entry to the .rhosts file for the remote user on the RCP server.

  For example, suppose the switch contains these configuration lines:

  ```
  hostname Switch1
  ip rcmd remote-username User0
  ```

  If the switch IP address translates to *Switch1.company.com*, the .rhosts file for User0 on the RCP server should contain this line:

  ```
  Switch1.company.com Switch1
  ```

For more information, see the documentation for your RCP server.

Downloading an Image File By Using RCP

You can download a new image file and replace or keep the current image.

Beginning in privileged EXEC mode, follow Steps 1 through 6 to download a new image from an RCP server and overwrite the existing image. To keep the current image, go to Step 6.

	Command	Purpose
1.	Verify that the RCP server is properly configured by referring to the Preparing to Download or Upload a Configuration File By Using RCP, page 1049.	
2.	Log into the switch through the console port or a Telnet session.	
3.	**configure terminal**	Enters global configuration mode. This step is required only if you override the default remote username (see Steps 4 and 5).
4.	**ip rcmd remote-username** *username*	(Optional) Specifies the remote username.

	Command	Purpose
5.	**end**	Returns to privileged EXEC mode.
6.	**archive download-sw /overwrite /reload rcp:**[[[**//**[*username***@**]*location*]**/***directory*]**/***image-name***.tar**]	Downloads the image file from the RCP server to the switch, and overwrite the current image. ■ The **/overwrite** option overwrites the software image in flash memory with the downloaded image. ■ The **/reload** option reloads the system after downloading the image unless the configuration has been changed and not been saved. ■ For **//***username,* specify the username. For the RCP copy request to execute successfully, an account must be defined on the network server for the remote username. ■ For **@***location*, specify the IP address of the RCP server. ■ For **/***directory***/***image-name***.tar**, specify the directory (optional) and the image to download. Directory and image names are case sensitive.
7.	**archive download-sw /leave-old-sw /reload rcp:**[[[**//**[*username***@**]*location*]**/***directory*]**/***image-name***.tar**]	Downloads the image file from the RCP server to the switch, and keep the current image. ■ The **/leave-old-sw** option keeps the old software version after a download. ■ The **/reload** option reloads the system after downloading the image unless the configuration has been changed and not been saved. ■ For **//***username,* specify the username. For the RCP copy request to execute, an account must be defined on the network server for the remote username. ■ For **@***location*, specify the IP address of the RCP server. ■ **For /***directory*]**/***image-name***.tar**, specify the directory (optional) and the image to download. Directory and image names are case sensitive.

The download algorithm verifies that the image is appropriate for the switch model and that enough DRAM is present, or it aborts the process and reports an error. If you specify the **/overwrite** option, the download algorithm removes the existing image on the flash device whether or not it is the same as the new one, downloads the new image, and then reloads the software.

Note: If the flash device has sufficient space to hold two images and you want to overwrite one of these images with the same version, you must specify the **/overwrite** option.

If you specify the **/leave-old-sw**, the existing files are not removed. If there is not enough room to install the new image an keep the running image, the download process stops, and an error message is displayed.

The algorithm installs the downloaded image onto the system board flash device (flash:). The image is placed into a new directory named with the software version string, and the BOOT environment variable is updated to point to the newly installed image.

If you kept the old software during the download process (you specified the **/leave-old-sw** keyword), you can remove it by entering the **delete /force /recursive** *filesystem***:/***file-url* privileged EXEC command. For *filesystem*, use **flash:** for the system board flash device. For *file-url*, enter the directory name of the old software image. All the files in the directory and the directory are removed.

Caution: For the download and upload algorithms to operate properly, do *not* rename image names.

Uploading an Image File By Using RCP

You can upload an image from the switch to an RCP server. You can later download this image to the same switch or to another switch of the same type.

The upload feature should be used only if the web management pages associated with the embedded device manager have been installed with the existing image.

Beginning in privileged EXEC mode, follow these steps to upload an image to an RCP server:

	Command	Purpose
1.	Verify that the RCP server is properly configured by referring to the Preparing to Download or Upload a Configuration File By Using RCP, page 1049.	
2.	Log into the switch through the console port or a Telnet session.	
3.	**configure terminal**	Enters global configuration mode. This step is required only if you override the default remote username (see Steps 4 and 5).
4.	**ip rcmd remote-username** *username*	(Optional) Specifies the remote username.
5.	**end**	Returns to privileged EXEC mode.
6.	**archive upload-sw** **rcp:**[[[**//**[*username***@**]*location*]**/***directory*]**/***image-name***.tar**]	Uploads the currently running switch image to the RCP server. ■ For **//***username,* specify the username; for the RCP copy request to execute, an account must be defined on the network server for the remote username. ■ For **@***location*, specify the IP address of the RCP server. ■ For **/***directory*]**/***image-name***.tar**, specify the directory (optional) and the name of the software image to be uploaded. Directory and image names are case sensitive. ■ The *image-name***.tar** is the name of software image to be stored on the server.

The **archive upload-sw** privileged EXEC command builds an image file on the server by uploading these files in order: info, the Cisco IOS image, and the web management files. After these files are uploaded, the upload algorithm creates the tar file format.

Caution: For the download and upload algorithms to operate properly, do *not* rename image names.

Displaying Image Upgrade and Downgrade History

The **show archive sw-upgrade history** command displays the history of all software image upgrades and downgrades performed on the device. This command displays the image name, version, upgrade method, and timeline for each upgrade that is done through Auto Install, PnP, **archive sw-download** CLI, or HTTP methods. Manual upgrades done through TFTP of tar files or binary files are not displayed.

Note: Only the first 100 upgrade or downgrade records are displayed.

To display the software image upgrade and downgrade history on a device, enter **show archive sw- upgrade history** in privileged EXEC mode as shown in the following examples.

Archive download Example

```
SW1-2#sh arc sw-upgrade history all

SWITCH: 1
File_name                                    Version            Install Mode/Date
---------------------------------- ------- ------------------
ie5000-universalk9-mz.152-7.1.85k.E3.bin     152-7.1.85k.E3     download-sw/UTC Sun Apr 24 2011
ie5000-universalk9-mz.152-7.1.85k.E3.bin     152-7.1.85k.E3     download-sw/UTC Sun Apr 24 2011
ie5000-universalk9-mz.152-7.1.85k.E3.bin     152-7.1.85k.E3     download-sw/UTC Sun Apr 24 2011
```

DNAC (PnP) Example

```
Switch#sh archive sw-upgrade history
File_name                                    Version            Install Mode/Date
---------------------------------- ------- ------------------
ie5000-universalk9-mz.152-7.68i.E3.bin       152-7.68i.E3       pnp/UTC Mon Jan 2 2006
ie5000-universalk9-mz.152-7.1.76i.E3.bin     152-7.1.76i.E3     pnp/UTC Mon Jan 2 2006
```

HTTP Example

```
Switch#sh archive sw-upgrade history
File_name                                    Version            Install Mode/Date
---------------------------------- ------- ------------------
ie5000-universalk9-mz.152-7.68i.E3.bin       152-7.68i.E3       http/UTC Mon Jan 2 2006
ie5000-universalk9-mz.152-7.1.76i.E3.bin     152-7.1.76i.E3     http/UTC Mon Jan 2 2006
```

Configuring EtherChannels

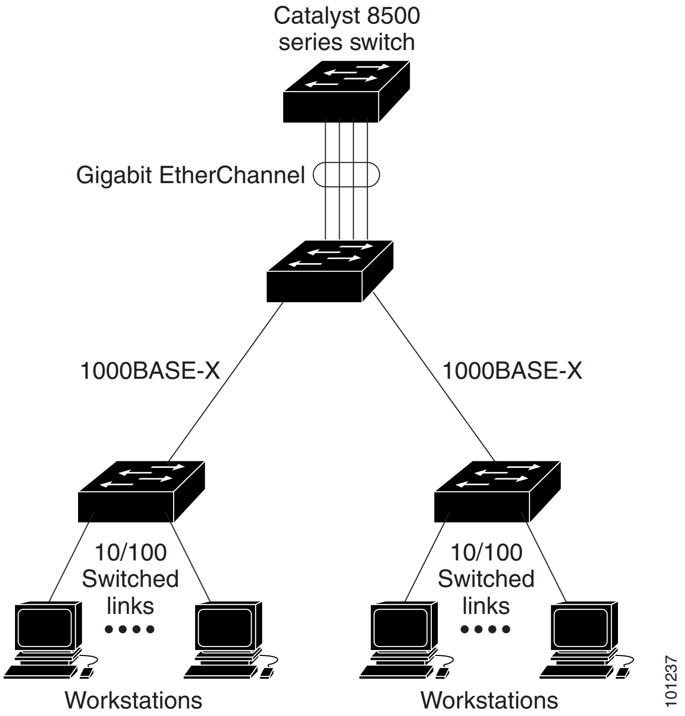
Information About Configuring EtherChannels

This chapter describes how to configure EtherChannels on the switch. EtherChannel provides fault-tolerant high-speed links between switches, routers, and servers. You can use it to increase the bandwidth between the wiring closets and the data center, and you can deploy it anywhere in the network where bottlenecks are likely to occur. EtherChannel provides automatic recovery for the loss of a link by redistributing the load across the remaining links. If a link fails, EtherChannel redirects traffic from the failed link to the remaining links in the channel without intervention. This chapter also describes how to configure link-state tracking.

EtherChannels

An EtherChannel consists of individual Ethernet links bundled into a single logical link as shown in Figure 110 on page 1069.

Figure 110 Typical EtherChannel Configuration



The EtherChannel provides full-duplex bandwidth up 2 Gb/s (Gigabit EtherChannel) between your switch and another switch or host. Each EtherChannel can consist of up to eight compatibly configured Ethernet ports.

The number of EtherChannels is limited to 10. For more information, see EtherChannel Configuration Guidelines, page 1077.

You can configure an EtherChannel in one of these modes: Port Aggregation Protocol (PAgP), Link Aggregation Control Protocol (LACP), or On. Configure both ends of the EtherChannel in the same mode:

- When you configure one end of an EtherChannel in either PAgP or LACP mode, the system negotiates with the other end of the channel to determine which ports should become active. Incompatible ports are put into an independent state and continue to carry data traffic as would any other single link. The port configuration does not change, but the port does not participate in the EtherChannel.

- When you configure an EtherChannel in the **on** mode, no negotiations take place. The switch forces all compatible ports to become active in the EtherChannel. The other end of the channel (on the other switch) must also be configured in the **on** mode; otherwise, packet loss can occur.

If a link within an EtherChannel fails, traffic previously carried over that failed link moves to the remaining links within the EtherChannel. If traps are enabled on the switch, a trap is sent for a failure that identifies the switch, the EtherChannel, and the failed link. Inbound broadcast and multicast packets on one link in an EtherChannel are blocked from returning on any other link of the EtherChannel.

Port-Channel Interfaces

When you create an EtherChannel, a port-channel logical interface is involved:

- With Layer 2 ports, use the **channel-group** interface configuration command to dynamically create the port-channel logical interface.

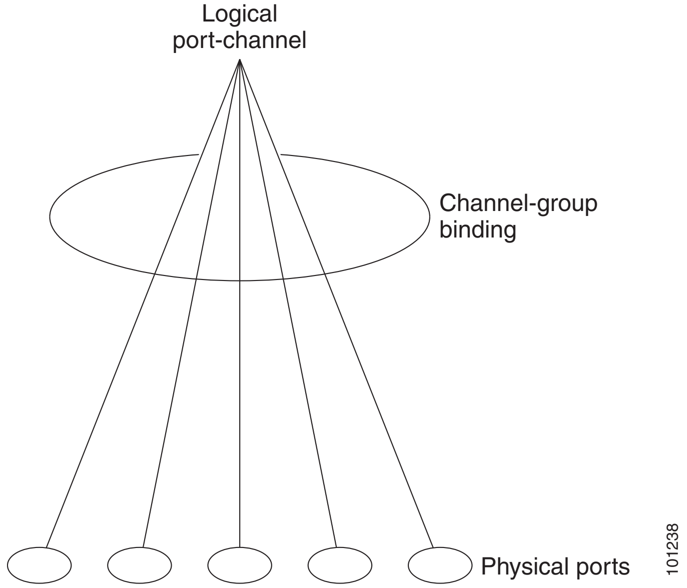
  You also can use the **interface port-channel** *port-channel-number* global configuration command to manually create the port-channel logical interface, but then you must use the **channel-group** *channel-group-number* command to bind the logical interface to a physical port. The *channel-group-number* can be the same as the *port-channel-number,* or you can use a new number. If you use a new number, the **channel-group** command dynamically creates a new port channel.

- With Layer 3 ports, you should manually create the logical interface by using the **interface port-channel** global configuration command followed by the **no switchport** interface configuration command. Then you manually assign an interface to the EtherChannel by using the **channel-group** interface configuration command.

For both Layer 2 and Layer 3 ports, the **channel-group** command binds the physical port and the logical interface together as shown in Figure 111 on page 1071.

Each EtherChannel has a port-channel logical interface numbered from 1 to 10. This port-channel interface number corresponds to the one specified with the **channel-group** interface configuration command.

Figure 111 Relationship of Physical Ports, Logical Port Channels, and Channel Groups



After you configure an EtherChannel, configuration changes applied to the port-channel interface apply to all the physical ports assigned to the port-channel interface. Configuration changes applied to the physical port affect only the port where you apply the configuration. To change the parameters of all ports in an EtherChannel, apply configuration commands to the port-channel interface, for example, spanning-tree commands or commands to configure a Layer 2 EtherChannel as a trunk.

Port Aggregation Protocol

The Port Aggregation Protocol (PAgP) is a Cisco-proprietary protocol that can be run only on Cisco switches and on those switches licensed by vendors to support PAgP. PAgP facilitates the automatic creation of EtherChannels by exchanging PAgP packets between Ethernet ports.

By using PAgP, the switch learns the identity of partners capable of supporting PAgP and the capabilities of each port. It then dynamically groups similarly configured ports into a single logical link (channel or aggregate port). Similarly configured ports are grouped based on hardware, administrative, and port parameter constraints. For example, PAgP groups the ports with the same speed, duplex mode, native VLAN, VLAN range, and trunking status and type. After grouping the links into an EtherChannel, PAgP adds the group to the spanning tree as a single switch port.

PAgP Modes

Table 70User-Configurable EtherChannel PAgP Modes, page 1072 shows the user-configurable EtherChannel PAgP modes for the **channel-group** interface configuration command.

Table 70 User-Configurable EtherChannel PAgP Modes

Mode	Description
auto	Places a port into a passive negotiating state, in which the port responds to PAgP packets it receives but does not start PAgP packet negotiation. This setting minimizes the transmission of PAgP packets.
desirable	Places a port into an active negotiating state, in which the port starts negotiations with other ports by sending PAgP packets.

Switch ports exchange PAgP packets only with partner ports configured in the **auto** or **desirable** modes. Ports configured in the **on** mode do not exchange PAgP packets.

Both the **auto** and **desirable** modes enable ports to negotiate with partner ports to form an EtherChannel based on criteria such as port speed and, for Layer 2 EtherChannels, trunking state and VLAN numbers.

Ports can form an EtherChannel when they are in different PAgP modes as long as the modes are compatible. For example:

- A port in the **desirable** mode can form an EtherChannel with another port that is in the **desirable** or **auto** mode.
- A port in the **auto** mode can form an EtherChannel with another port in the **desirable** mode.

A port in the **auto** mode cannot form an EtherChannel with another port that is also in the **auto** mode because neither port starts PAgP negotiation.

If your switch is connected to a partner that is PAgP-capable, you can configure the switch port for nonsilent operation by using the **non-silent** keyword. If you do not Specifies **non-silent** with the **auto** or **desirable** mode, silent mode is assumed.

Use the silent mode when the switch is connected to a device that is not PAgP-capable and seldom, if ever, sends packets. An example of a silent partner is a file server or a packet analyzer that is not generating traffic. In this case, running PAgP on a physical port connected to a silent partner prevents that switch port from ever becoming operational. However, the silent setting allows PAgP to operate, to attach the port to a channel group, and to use the port for transmission.

PAgP Learn Method and Priority

Network devices are classified as PAgP physical learners or aggregate-port learners. A device is a physical learner if it learns addresses by physical ports and directs transmissions based on that knowledge. A device is an aggregate-port learner if it learns addresses by aggregate (logical) ports. The learn method must be configured the same at both ends of the link.

When a device and its partner are both aggregate-port learners, they learn the address on the logical port-channel. The device sends packets to the source by using any of the ports in the EtherChannel. With aggregate-port learning, it is not important on which physical port the packet arrives.

PAgP cannot automatically detect when the partner device is a physical learner and when the local device is an aggregate-port learner. Therefore, you must manually set the learning method on the local device to learn addresses by physical ports. You also must set the load-distribution method to source-based distribution, so that any given source MAC address is always sent on the same physical port.

You also can configure a single port within the group for all transmissions and use other ports for hot standby. The unused ports in the group can be swapped into operation in just a few seconds if the selected single port loses hardware-signal detection. You can configure which port is always selected for packet transmission by changing its priority with the **pagp port-priority** interface configuration command. The higher the priority, the more likely that the port will be selected.

Note: The switch supports address learning only on aggregate ports even though the **physical-port** keyword is provided in the CLI. The **pagp learn-method** command and the **pagp port-priority** command have no effect on the switch hardware, but they are required for PAgP interoperability with devices that only support address learning by physical ports.

When the link partner of the switch is a physical learner (such as a Catalyst 1900 series switch), we recommend that you configure the switch as a physical-port learner by using the **pagp learn-method physical-port** interface configuration command. Set the load-distribution method based on the source MAC address by using the **port-channel load-balance src-mac** global configuration command. The switch then sends packets to the Catalyst 1900 switch using the same port in the EtherChannel from which it learned the source address. Only use the **pagp learn-method** command in this situation.

PAgP Interaction with Virtual Switches and Dual-Active Detection

A virtual switch can be two or more core switches connected by virtual switch links (VSLs) that carry control and data traffic between them. One of the switches is in active mode. The others are in standby mode. For redundancy, remote switches, are connected to the virtual switch by remote satellite links (RSLs).

If the VSL between two switches fails, one switch does not know the status of the other. Both switches could change to the active mode, causing a *dual-active situation* in the network with duplicate configurations (including duplicate IP addresses and bridge identifiers). The network might go down.

To prevent a dual-active situation, the core switches send PAgP protocol data units (PDUs) through the RSLs to the remote switches. The PAgP PDUs identify the active switch, and the remote switches forward the PDUs to core switches so that the core switches are in sync. If the active switch fails or resets, the standby switch takes over as the active switch. If the VSL goes down, one core switch knows the status of the other and does not change state.

PAgP Interaction with Other Features

The Dynamic Trunking Protocol (DTP) and the Cisco Discovery Protocol (CDP) send and receive packets over the physical ports in the EtherChannel. Trunk ports send and receive PAgP protocol data units (PDUs) on the lowest numbered VLAN.

In Layer 2 EtherChannels, the first port in the channel that comes up provides its MAC address to the EtherChannel. If this port is removed from the bundle, one of the remaining ports in the bundle provides its MAC address to the EtherChannel.

PAgP sends and receives PAgP PDUs only from ports that are up and have PAgP enabled for the auto or desirable mode.

Link Aggregation Control Protocol

The LACP is defined in IEEE 802.3ad and enables Cisco switches to manage Ethernet channels between switches that conform to the IEEE 802.3ad protocol. LACP facilitates the automatic creation of EtherChannels by exchanging LACP packets between Ethernet ports.

By using LACP, the switch learns the identity of partners capable of supporting LACP and the capabilities of each port. It then dynamically groups similarly configured ports into a single logical link (channel or aggregate port). Similarly configured ports are grouped based on hardware, administrative, and port parameter constraints. For example, LACP groups the ports with the same speed, duplex mode, native VLAN, VLAN range, and trunking status and type. After grouping the links into an EtherChannel, LACP adds the group to the spanning tree as a single switch port.

LACP Modes

Table 2 shows the user-configurable EtherChannel LACP modes for the **channel-group** interface configuration command.

Table 71 User-Configurable EtherChannel LACP Modes

Mode	Description
active	Places a port into an active negotiating state in which the port starts negotiations with other ports by sending LACP packets.
passive	Places a port into a passive negotiating state in which the port responds to LACP packets that it receives, but does not start LACP packet negotiation. This setting minimizes the transmission of LACP packets.

Both the **active** and **passive LACP** modes enable ports to negotiate with partner ports to an EtherChannel based on criteria such as port speed and, for Layer 2 EtherChannels, trunking state and VLAN numbers.

Ports can form an EtherChannel when they are in different LACP modes as long as the modes are compatible. For example:

- A port in the **active** mode can form an EtherChannel with another port that is in the **active** or **passive** mode.
- A port in the **passive** mode cannot form an EtherChannel with another port that is also in the **passive** mode because neither port starts LACP negotiation.

LACP Hot-Standby Ports

When enabled, LACP tries to configure the maximum number of LACP-compatible ports in a channel, up to a maximum of 16 ports. Only eight LACP links can be active at one time. The software places any additional links in a hot-standby mode. If one of the active links becomes inactive, a link that is in the hot-standby mode becomes active in its place.

If you configure more than eight links for an EtherChannel group, the software automatically decides which of the hot-standby ports to make active based on the LACP priority. To every link between systems that operate LACP, the software assigns a unique priority made up of these elements (in priority order):

- LACP system priority
- System ID (the switch MAC address)
- LACP port priority
- Port number

In priority comparisons, numerically lower values have higher priority. The priority decides which ports should be put in standby mode when there is a hardware limitation that prevents all compatible ports from aggregating.

Determining which ports are active and which are hot standby is a two-step procedure. First the system with a numerically lower system priority and system-id is placed in charge of the decision. Next, that system decides which ports are active and which are hot standby, based on its values for port priority and port number. The port-priority and port-number values for the other system are not used.

You can change the default values of the LACP system priority and the LACP port priority to affect how the software selects active and standby links.

By default, all ports use the same port priority. If the local system has a lower value for the system priority and the system ID than the remote system, you can affect which of the hot-standby links become active first by changing the port priority of LACP EtherChannel ports to a lower value than the default**.** The hot-standby ports that have lower port numbers become active in the channel first. You can use the **show etherchannel summary** privileged EXEC command to see which ports are in the hot-standby mode (denoted with an *H* port-state flag).

If LACP is not able to aggregate all the ports that are compatible (for example, the remote system might have more restrictive hardware limitations), all the ports that cannot be actively included in the EtherChannel are put in the hot-standby state and are used only if one of the channeled ports fails.

LACP Interaction with Other Features

The DTP and the CDP send and receive packets over the physical ports in the EtherChannel. Trunk ports send and receive LACP PDUs on the lowest numbered VLAN.

In Layer 2 EtherChannels, the first port in the channel that comes up provides its MAC address to the EtherChannel. If this port is removed from the bundle, one of the remaining ports in the bundle provides its MAC address to the EtherChannel.

LACP sends and receives LACP PDUs only from ports that are up and have LACP enabled for the active or passive mode.

EtherChannel On Mode

EtherChannel **on** mode can be used to manually configure an EtherChannel. The **on** mode forces a port to join an EtherChannel without negotiations. The **on** mode can be useful if the remote device does not support PAgP or LACP. In the **on** mode, a usable EtherChannel exists only when the switches at both ends of the link are configured in the **on** mode.

Ports that are configured in the **on** mode in the same channel group must have compatible port characteristics, such as speed and duplex. Ports that are not compatible are suspended, even though they are configured in the **on** mode.

Caution: You should use care when using the on mode. This is a manual configuration, and ports on both ends of the EtherChannel must have the same configuration. If the group is misconfigured, packet loss or spanning-tree loops can occur.

Load Balancing and Forwarding Methods

EtherChannel balances the traffic load 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 load balancing can use MAC addresses or IP addresses, source or destination addresses, or both source and destination addresses. The selected mode applies to all EtherChannels configured on the switch. You configure the load balancing and forwarding method by using the **port-channel load-balance** global configuration command.

With source-MAC address forwarding, when packets are forwarded to an EtherChannel, they are distributed across the ports in the channel based on the source-MAC address of the incoming packet. Therefore, to provide load balancing, packets from different hosts use different ports in the channel, but packets from the same host use the same port in the channel.

With destination-MAC address forwarding, when packets are forwarded to an EtherChannel, they are distributed across the ports in the channel based on the destination host’s MAC address of the incoming packet. Therefore, packets to the same destination are forwarded over the same port, and packets to a different destination are sent on a different port in the channel.

With source-and-destination MAC address forwarding, when packets are forwarded to an EtherChannel, they are distributed across the ports in the channel based on both the source and destination MAC addresses. This forwarding method, a combination source-MAC and destination-MAC address forwarding methods of load distribution, can be used if it is not clear whether source-MAC or destination-MAC address forwarding is better suited on a particular switch. With source-and-destination MAC-address forwarding, packets sent from host A to host B, host A to host C, and host C to host B could all use different ports in the channel.

With source-IP address-based forwarding, when packets are forwarded to an EtherChannel, they are distributed across the ports in the EtherChannel based on the source-IP address of the incoming packet. Therefore, to provide load-balancing, packets from different IP addresses use different ports in the channel, but packets from the same IP address use the same port in the channel.

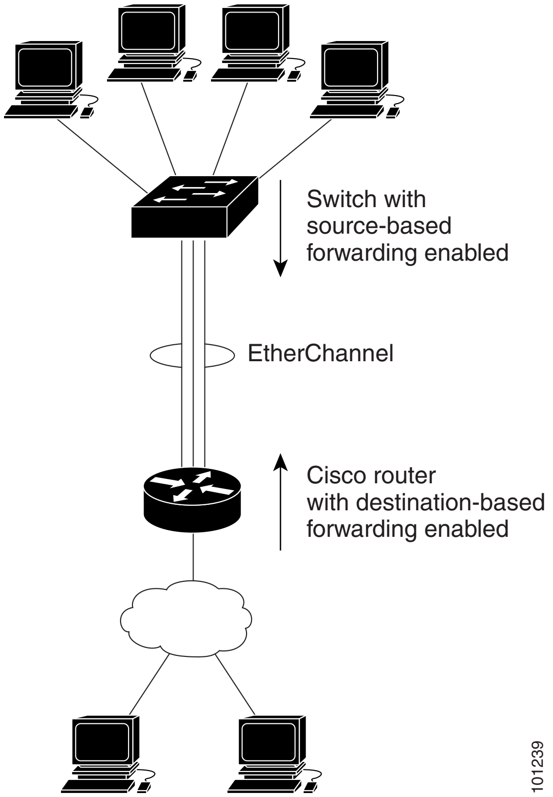
With destination-IP address-based forwarding, when packets are forwarded to an EtherChannel, they are distributed across the ports in the EtherChannel based on the destination-IP address of the incoming packet. Therefore, to provide load-balancing, packets from the same IP source address sent to different IP destination addresses could be sent on different ports in the channel. But packets sent from different source IP addresses to the same destination IP address are always sent on the same port in the channel.

With source-and-destination IP address-based forwarding, packets are sent to an EtherChannel and distributed across the EtherChannel ports, based on both the source and destination IP addresses of the incoming packet. This forwarding method, a combination of source-IP and destination-IP address-based forwarding, can be used if it is not clear whether source-IP or destination-IP address-based forwarding is better suited on a particular switch. In this method, packets sent from the IP address A to IP address B, from IP address A to IP address C, and from IP address C to IP address B could all use different ports in the channel.

Different load-balancing methods have different advantages, and the choice of a particular load-balancing method should be based on the position of the switch in the network and the kind of traffic that needs to be load-distributed. In Figure 112 on page 1077, an EtherChannel from a switch that is aggregating data from four workstations communicates with a router. Because the router is a single-MAC-address device, source-based forwarding on the switch EtherChannel ensures that the switch uses all available bandwidth to the router. The router is configured for destination-based forwarding because the large number of workstations ensures that the traffic is evenly distributed from the router EtherChannel.

Use the option that provides the greatest variety in your configuration. For example, if the traffic on a channel is only going to a single MAC address, using the destination-MAC address always chooses the same link in the channel. Using source addresses or IP addresses might result in better load balancing.

Figure 112 Load Distribution and Forwarding Methods



Default EtherChannel Settings

Feature	Default Setting
Channel groups	None assigned.
Port-channel logical interface	None defined.
PAgP mode	No default.
PAgP learn method	Aggregate-port learning on all ports.
PAgP priority	128 on all ports.
LACP mode	No default.
LACP learn method	Aggregate-port learning on all ports.
LACP port priority	32768 on all ports.
LACP system priority	32768.
LACP system ID	LACP system priority and the switch MAC address.
Load balancing	Load distribution on the switch is based on the source-MAC address of the incoming packet.

EtherChannel Configuration Guidelines

If improperly configured, some EtherChannel ports are automatically disabled to avoid network loops and other problems. Follow these guidelines to avoid configuration problems:

- Do not try to configure more than 6 EtherChannels on the switch.
- Configure a PAgP EtherChannel with up to ten Ethernet ports of the same type.
- Configure a LACP EtherChannel with up to16 Ethernet ports of the same type. Up to eight ports can be active, and up to eight ports can be in standby mode.
- Configure all ports in an EtherChannel to operate at the same speeds and duplex modes.
- Enable all ports in an EtherChannel. A port in an EtherChannel that is disabled by using the **shutdown** interface configuration command is treated as a link failure, and its traffic is transferred to one of the remaining ports in the EtherChannel.
- When a group is first created, all ports follow the parameters set for the first port to be added to the group. If you change the configuration of one of these parameters, you must also make the changes to all ports in the group:
  - Allowed-VLAN list
  - Spanning-tree path cost for each VLAN
  - Spanning-tree port priority for each VLAN
  - Spanning-tree Port Fast setting
- Do not configure a port to be a member of more than one EtherChannel group.
- Do not configure an EtherChannel in both the PAgP and LACP modes. EtherChannel groups running PAgP and LACP can coexist on the same switch. Individual EtherChannel groups can run either PAgP or LACP, but they cannot interoperate.
- Do not configure a Switched Port Analyzer (SPAN) destination port as part of an EtherChannel.
- Do not configure a secure port as part of an EtherChannel or the reverse.
- Do not configure a private-VLAN port as part of an EtherChannel.
- Do not configure a port that is an active or a not-yet-active member of an EtherChannel as an IEEE 802.1x port. If you try to enable IEEE 802.1x on an EtherChannel port, an error message appears, and IEEE 802.1x is not enabled.
- If EtherChannels are configured on switch interfaces, remove the EtherChannel configuration from the interfaces before globally enabling IEEE 802.1x on a switch by using the **dot1x system-auth-control** global configuration command.
- For Layer 2 EtherChannels:
  - Assign all ports in the EtherChannel to the same VLAN, or configure them as trunks. Ports with different native VLANs cannot form an EtherChannel.
  - If you configure an EtherChannel from trunk ports, verify that the trunking mode (ISL or IEEE 802.1Q) is the same on all the trunks. Inconsistent trunk modes on EtherChannel ports can have unexpected results.
  - An EtherChannel supports the same allowed range of VLANs on all the ports in a trunking Layer 2 EtherChannel. If the allowed range of VLANs is not the same, the ports do not form an EtherChannel even when PAgP is set to the **auto** or **desirable** mode.
  - Ports with different spanning-tree path costs can form an EtherChannel if they are otherwise compatibly configured. Setting different spanning-tree path costs does not, by itself, make ports incompatible for the formation of an EtherChannel.

How to Configure EtherChannels

Note: After you configure an EtherChannel, configuration changes applied to the port-channel interface apply to all the physical ports assigned to the port-channel interface, and configuration changes applied to the physical port affect only the port where you apply the configuration.

Configuring Layer 2 EtherChannels

You configure Layer 2 EtherChannels by assigning ports to a channel group with the **channel-group** interface configuration command. This command automatically creates the port-channel logical interface.

This required task explains how to configure a Layer 2 Ethernet port to a Layer 2 EtherChannel.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies a physical port, and enter interface configuration mode. Valid interfaces include physical ports. For a PAgP EtherChannel, you can configure up to eight ports of the same type and speed for the same group. For a LACP EtherChannel, you can configure up to 16 Ethernet ports of the same type. Up to eight ports can be active, and up to eight ports can be in standby mode.

	Command	Purpose
3.	**switchport mode** {**access** | **trunk**} **switchport access vlan** *vlan-id*	Assigns all ports as static-access ports in the same VLAN, or configures them as trunks. If you configure the port as a static-access port, assign it to only one VLAN. The range is 1 to 4096.
4.	**channel-group** *channel-group-number* **mode** {**auto** [**non-silent**] | **desirable** [**non-silent**] | **on**} | {**active** | **passive**}	Assigns the port to a channel group, and specifies the PAgP or the LACP mode. For *channel-group-number*, the range is 1 to 10. For **mode**, select one of these keywords: ■ **auto**–Enables PAgP only if a PAgP device is detected. It places the port into a passive negotiating state, in which the port responds to PAgP packets it receives but does not start PAgP packet negotiation. ■ **desirable**–Unconditionally enables PAgP. It places the port into an active negotiating state, in which the port starts negotiations with other ports by sending PAgP packets. ■ **on**–Forces the port to channel without PAgP or LACP. In the **on** mode, an EtherChannel exists only when a port group in the **on** mode is connected to another port group in the **on** mode. ■ **non-silent**–(Optional) If your switch is connected to a partner that is PAgP-capable, configure the switch port for nonsilent operation when the port is in the **auto** or **desirable** mode. If you do not Specifies **non-silent**, silent is assumed. The silent setting is for connections to file servers or packet analyzers. This setting allows PAgP to operate, to attach the port to a channel group, and to use the port for transmission. ■ **active**–Enables LACP only if a LACP device is detected. It places the port into an active negotiating state in which the port starts negotiations with other ports by sending LACP packets. ■ **passive**–Enables LACP on the port and places it into a passive negotiating state in which the port responds to LACP packets that it receives, but does not start LACP packet negotiation. For information on compatible modes for the switch and its partner, see PAgP Modes, page 1071 and the LACP Modes, page 1073.
5.	**end**	Returns to privileged EXEC mode.

Configuring EtherChannel Load Balancing

This task is optional.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**port-channel load-balance** {**dst-ip** | **dst-mac** | **src-dst-ip** | **src-dst-mac** | **src-ip** | **src-mac**}	Configures an EtherChannel load-balancing method. The default is **src-mac**. Select one of these load-distribution methods: ■ **dst-ip**–Specifies the destination-host IP address. ■ **dst-mac**–Specifies the destination-host MAC address of the incoming packet. ■ **src-dst-ip**– Specifies the source-and-destination host-IP address. ■ **src-dst-mac**–Specifies the source-and-destination host-MAC address. ■ **src-ip**– Specifies the source-host IP address. ■ **src-mac**–Specifies the source-MAC address of the incoming packet.
3.	**end**	Returns to privileged EXEC mode.

Configuring the PAgP Learn Method and Priority

This task is optional.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**interface** *interface-id*	Specifies the port for transmission, and enter interface configuration mode.

	Command	Purpose
3.	**pagp learn-method physical-port**	Selects the PAgP learning method. By default, **aggregation-port learning** is selected, which means the switch sends packets to the source by using any of the ports in the EtherChannel. With aggregate-port learning, it is not important on which physical port the packet arrives. Select **physical-port** to connect with another switch that is a physical learner. Make sure to configure the **port-channel load-balance** global configuration command to **src-mac** as described in the Configuring EtherChannel Load Balancing, page 1080. The learning method must be configured the same at both ends of the link.
4.	**pagp port-priority** *priority*	Assigns a priority so that the selected port is chosen for packet transmission. For *priority*, the range is 0 to 255. The default is 128. The higher the priority, the more likely that the port will be used for PAgP transmission.
5.	**end**	Returns to privileged EXEC mode.

Configuring the LACP Hot-Standby Ports

This task is optional.

	Command	Purpose
1.	**configure terminal**	Enters global configuration mode.
2.	**lacp system-priority** *priority*	Configures the LACP system priority. For *priority*, the range is 1 to 65535. The default is 32768. The lower the value, the higher the system priority.
3.	**interface** *interface-id*	Specifies the port to be configured, and enters interface configuration mode.
4.	**lacp port-priority** ***priority***	Configures the LACP port priority. For *priority*, the range is 1 to 65535. The default is 32768. The lower the value, the more likely that the port will be used for LACP transmission.
5.	**end**	Returns to privileged EXEC mode.

Monitoring and Maintaining EtherChannels

Command	Purpose
show etherchannel [*channel-group-number* {**detail** | **port** | **port-channel** | **protocol** | **summary**}] {**detail** | **load-balance** | **port** | **port-channel** | **protocol** | **summary**}	Displays EtherChannel information in a brief, detailed, and one-line summary form. Also displays the load-balance or frame-distribution scheme, port, port-channel, and protocol information.
show pagp [*channel-group-number*] {**counters** | **internal** | **neighbor**}	Displays PAgP information such as traffic information, the internal PAgP configuration, and neighbor information.
show pagp [*channel-group-number*] **dual-active**	Displays the dual-active detection status.
show lacp [*channel-group-number*] {**counters** | **internal** | **neighbor**}	Displays LACP information such as traffic information, the internal LACP configuration, and neighbor information.

Configuration Examples for Configuring EtherChannels

Configuring EtherChannels: Examples

This example shows how to configure an EtherChannel and assign two ports as static-access ports in VLAN 10 to channel 5 with the PAgP mode **desirable**:

```
Switch# configure terminal
Switch(config)# interface range GigabitEthernet1/17 -2
Switch(config-if-range)# switchport mode access
Switch(config-if-range)# switchport access vlan 10
Switch(config-if-range)# channel-group 5 mode desirable non-silent
Switch(config-if-range)# end
```

This example shows how to configure an EtherChannel and assign two ports as static-access ports in VLAN 10 to channel 5 with the LACP mode **active**:

```
Switch# configure terminal
Switch(config)# interface range GigabitEthernet1/17 -2
Switch(config-if-range)# switchport mode access
Switch(config-if-range)# switchport access vlan 10
Switch(config-if-range)# channel-group 5 mode active
Switch(config-if-range)# end
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Troubleshooting

This chapter describes how to identify and resolve software problems related to the Cisco IOS software on the switch. Depending on the nature of the problem, you can use the command-line interface (CLI), Network Assistant or Device Manager to identify and solve problems.

For additional troubleshooting information, such as LED descriptions, see the *Hardware Installation Guide*.

Information for Troubleshooting

Autonegotiation Mismatches Prevention

The IEEE 802.3ab autonegotiation protocol manages the switch settings for speed (10 Mb/s, 100 Mb/s, and 1000 Mb/s, excluding SFP module ports) and duplex (half or full). There are situations when this protocol can incorrectly align these settings, reducing performance. A mismatch occurs under these circumstances:

- A manually set speed or duplex parameter is different from the manually set speed or duplex parameter on the connected port.
- A port is set to autonegotiate, and the connected port is set to full duplex with no autonegotiation.

To maximize switch performance and ensure a link, follow one of these guidelines when changing the settings for duplex and speed:

- Let both ports autonegotiate both speed and duplex.
- Manually set the speed and duplex parameters for the ports on both ends of the connection.

Note: If a remote device does not autonegotiate, configure the duplex settings on the two ports to match. The speed parameter can adjust itself even if the connected port does not autonegotiate.

SFP Module Security and Identification

Cisco small form-factor pluggable (SFP) modules have a serial EEPROM that contains the module serial number, the vendor name and ID, a unique security code, and cyclic redundancy check (CRC). When an SFP module is inserted in the switch, the switch software reads the EEPROM to verify the serial number, vendor name and vendor ID, and recompute the security code and CRC. If the serial number, the vendor name or vendor ID, the security code, or CRC is invalid, the software generates a security error message and places the interface in an error-disabled state.

Note: The security error message references the GBIC_SECURITY facility. The switch supports SFP modules and does not support GBIC modules. Although the error message text refers to GBIC interfaces and modules, the security messages actually refer to the SFP modules and module interfaces.

If you are using a non-Cisco SFP module, remove the SFP module from the switch, and replace it with a Cisco module. After inserting a Cisco SFP module, use the **errdisable recovery cause gbic-invalid** global configuration command to verify the port status, and enter a time interval for recovering from the error-disabled state. After the elapsed interval, the switch brings the interface out of the error-disabled state and retries the operation.

If the module is identified as a Cisco SFP module, but the system is unable to read vendor-data information to verify its accuracy, an SFP module error message is generated. In this case, you should remove and reinsert the SFP module. If it continues to fail, the SFP module might be defective.

Ping

The switch supports IP ping, which you can use to test connectivity to remote hosts. Ping sends an echo request packet to an address and waits for a reply. Ping returns one of these responses:

- Normal response—The normal response (*hostname* is alive) occurs in 1 to 10 seconds, depending on 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.

Layer 2 Traceroute

The Layer 2 traceroute feature allows the switch to identify the physical path that a packet takes from a source device to a destination device. Layer 2 traceroute supports only unicast source and destination MAC addresses. It finds the path by using the MAC address tables of the switches in the path. When the switch detects a device in the path that does not support Layer 2 traceroute, the switch continues to send Layer 2 trace queries and lets them time out.

The switch can only identify the path from the source device to the destination device. It cannot identify the path that a packet takes from source host to the source device or from the destination device to the destination host.

Layer 2 Traceroute Usage Guidelines

- Cisco Discovery Protocol (CDP) must be enabled on all the devices in the network. For Layer 2 traceroute to function properly, do not disable CDP.

  If any devices in the physical path are transparent to CDP, the switch cannot identify the path through these devices. For more information about enabling CDP, see Configuring CDP, page 529
- A switch is reachable from another switch when you can test connectivity by using the **ping** privileged EXEC command. All switches in the physical path must be reachable from each other.
- The maximum number of hops identified in the path is ten.
- You can enter the **traceroute mac** or the **traceroute mac ip** privileged EXEC command on a switch that is not in the physical path from the source device to the destination device. All switches in the path must be reachable from this switch.
- The **traceroute mac** command output shows the Layer 2 path only when the specified source and destination MAC addresses belong to the same VLAN. If you specify source and destination MAC addresses that belong to different VLANs, the Layer 2 path is not identified, and an error message appears.
- If you specify a multicast source or destination MAC address, the path is not identified, and an error message appears.
- If the source or destination MAC address belongs to multiple VLANs, you must specify the VLAN to which both the source and destination MAC addresses belong. If the VLAN is not specified, the path is not identified, and an error message appears.

- The **traceroute mac ip** command output shows the Layer 2 path when the specified source and destination IP addresses belong to the same subnet. When you specify the IP addresses, the switch uses the Address Resolution Protocol (ARP) to associate the IP addresses with the corresponding MAC addresses and the VLAN IDs.
  - If an ARP entry exists for the specified IP address, the switch uses the associated MAC address and identifies the physical path.
  - If an ARP entry does not exist, the switch sends an ARP query and tries to resolve the IP address. If the IP address is not resolved, the path is not identified, and an error message appears.
- When multiple devices are attached to one port through hubs (for example, multiple CDP neighbors are detected on a port), the Layer 2 traceroute feature is not supported. When more than one CDP neighbor is detected on a port, the Layer 2 path is not identified, and an error message appears.

IP Traceroute

You can use IP traceroute to identify the path that packets take through the network on a hop-by-hop basis. The command output displays all network layer (Layer 3) devices, such as routers, that the traffic passes through on the way to the destination.

Your switches can participate as the source or destination of the **traceroute** privileged EXEC command and might or might not appear as a hop in the **traceroute** command output. If the switch is the destination of the traceroute, it is displayed as the final destination in the traceroute output. Intermediate switches do not show up in the traceroute output if they are only bridging the packet from one port to another within the same VLAN. However, if the intermediate switch is a multilayer switch that is routing a particular packet, this switch shows up as a hop in the traceroute output.

The **traceroute** privileged EXEC command uses the Time To Live (TTL) field in the IP header to cause routers and 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 an Internet Control Message Protocol (ICMP) time-to-live-exceeded message to the sender. Traceroute finds the address of the first hop by examining the source address field of the ICMP time-to-live-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-to-live-exceeded message to the source. This process continues until the TTL is incremented to a value large enough for the datagram to reach the destination host (or until the maximum TTL is reached).

To learn when a datagram reaches its destination, traceroute sets the UDP destination port number in the datagram to a very large value that the destination host is unlikely to be using. When a host receives a datagram destined to itself containing a destination port number that is unused locally, it sends an ICMP *port-unreachable* error to the source. Because all errors except port-unreachable errors come from intermediate hops, the receipt of a port-unreachable error means that this message was sent by the destination port.

TDR

You can use the Time Domain Reflector (TDR) feature to diagnose and resolve cabling problems. When running TDR, a local device sends a signal through a cable and compares the reflected signal to the initial signal.

TDR is supported only on 10/100 and 10/100/1000 copper Ethernet ports. It is not supported on SFP module ports.

TDR can detect these cabling problems:

- Open, broken, or cut twisted-pair wires—The wires are not connected to the wires from the remote device.
- Shorted twisted-pair wires—The wires are touching each other or the wires from the remote device. For example, a shorted twisted pair can occur if one wire of the twisted pair is soldered to the other wire.

If one of the twisted-pair wires is open, TDR can find the length at which the wire is open.

Use TDR to diagnose and resolve cabling problems in these situations:

- Replacing a switch
- Setting up a wiring closet
- Troubleshooting a connection between two devices when a link cannot be established or when it is not operating properly

Crashinfo Files

The crashinfo files save information that helps Cisco technical support representatives to debug problems that caused the Cisco IOS image to fail (crash). The switch writes the crash information to the console at the time of the failure. The switch creates two types of crashinfo files:

- Basic crashinfo file—The switch automatically creates this file the next time you boot up the Cisco IOS image after the failure.
- Extended crashinfo file—The switch automatically creates this file when the system is failing.

Basic crashinfo Files

The information in the basic file includes the Cisco IOS image name and version that failed, a list of the processor registers, and other switch-specific information. You can provide this information to the Cisco technical support representative by using the **show tech-support** privileged EXEC command.

Basic crashinfo files are kept in this directory on the flash file system:

flash:/crashinfo/.

The filenames are crashinfo_*n* where *n* is a sequence number.

Each new crashinfo file that is created uses a sequence number that is larger than any previously existing sequence number, so the file with the largest sequence number describes the most recent failure. Version numbers are used instead of a timestamp because the switches do not include a real-time clock. You cannot change the name of the file that the system will use when it creates the file. However, after the file is created, you can use the **rename** privileged EXEC command to rename it, but the contents of the renamed file will not be displayed by the **show tech-support** privileged EXEC command. You can delete crashinfo files by using the **delete** privileged EXEC command.

You can display the most recent basic crashinfo file (that is, the file with the highest sequence number at the end of its filename) by entering the **show tech-support** privileged EXEC command. You also can access the file by using any command that can copy or display files, such as the **more** or the **copy** privileged EXEC command.

Extended crashinfo Files

The switch creates the extended crashinfo file when the system is failing. The information in the extended file includes additional information that can help determine the cause of the switch failure. You provide this information to the Cisco technical support representative by manually accessing the file and using the **more** or the **copy** privileged EXEC command.

Extended crashinfo files are kept in this directory on the flash file system:

flash:/crashinfo_ext/.

The filenames are crashinfo_ext_*n* where *n* is a sequence number.

You can configure the switch to not create the extended creashinfo file by using the **no exception crashinfo** global configuration command.

CPU Utilization

This section lists some possible symptoms that could be caused by the CPU being too busy and shows how to verify a CPU utilization problem. Table 72 on page 1089 lists the primary types of CPU utilization problems that you can identify. It gives possible causes and corrective action with links to the *Troubleshooting High CPU Utilization* document on Cisco.com.

Excessive CPU utilization might result in these symptoms, but the symptoms could also result from other causes.

- Spanning tree topology changes
- EtherChannel links brought down due to loss of communication
- Failure to respond to management requests (ICMP ping, SNMP timeouts, slow Telnet or SSH sessions)
- UDLD flapping
- IP SLAs failures because of SLAs responses beyond an acceptable threshold
- DHCP or IEEE 802.1x failures if the switch does not forward or respond to requests

Problem and Cause for High CPU Utilization

To determine if high CPU utilization is a problem, enter the **show processes cpu sorted** privileged EXEC command. Note the underlined information in the first line of the output example.

```
Switch# show processes cpu sorted
CPU utilization for five seconds: 8%/0%; one minute: 7%; five minutes: 8%
PID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min TTY Process
140 8820183 4942081 1784 0.63% 0.37% 0.30% 0 HRPC qos request
100 3427318 16150534 212 0.47% 0.14% 0.11% 0 HRPC pm-counters
192 3093252 14081112 219 0.31% 0.14% 0.11% 0 Spanning Tree
143 8 37 216 0.15% 0.01% 0.00% 0 Exec
...
<output truncated>
```

This example shows normal CPU utilization. The output shows that utilization for the last 5 seconds is *8%/0%*, which has this meaning:

- The total CPU utilization is 8 percent, including both time running Cisco IOS processes and time spent handling interrupts.
- The time spent handling interrupts is zero percent.

Table 72 Troubleshooting CPU Utilization Problems

Type of Problem	Cause	Corrective Action
Interrupt percentage value is almost as high as total CPU utilization value.	The CPU is receiving too many packets from the network.	Determine the source of the network packet. Stop the flow, or change the switch configuration. See the section on "Analyzing Network Traffic."
Total CPU utilization is greater than 50% with minimal time spent on interrupts.	One or more Cisco IOS process is consuming too much CPU time. This is usually triggered by an event that activated the process.	Identify the unusual event, and troubleshoot the root cause. See the section on "Debugging Active Processes."

- For complete information about CPU utilization and how to troubleshoot utilization problems, see the *Troubleshooting High CPU Utilization* document on Cisco.com.

How to Troubleshoot

Recovering from Software Failures

Switch software can be corrupted during an upgrade, by downloading the wrong file to the switch, and by deleting the image file. In all of these cases, the switch does not pass the power-on self-test (POST), and there is no connectivity.

This procedure uses the Xmodem Protocol to recover from a corrupt or wrong image file. There are many software packages that support the Xmodem Protocol, and this procedure is largely dependent on the emulation software that you are using.

This recovery procedure requires that you have physical access to the switch.

1. From your PC, download the software image tar file (*image_filename.tar*) from Cisco.com.

The Cisco IOS image is stored as a bin file in a directory in the tar file. For information about locating the software image files on Cisco.com, see the release notes.

2. Extract the bin file from the tar file.

- If you are using Windows, use a zip program that can read a tar file. Use the zip program to navigate to and extract the bin file.

- If you are using UNIX, follow these steps:

  – Display the contents of the tar file by using the **tar -tvf** <*image_filename.tar*> UNIX command.

```
switch% tar -tvf image_filename.tar
```

  – Locate the bin file, and extract it by using the **tar -xvf** <*image_filename.tar*> <*image_filename.bin*> UNIX command.

```
switch% tar -xvf image_filename.tar image_filename.binx

x image_name.bin, 3970586 bytes, 7756 tape blocks
```

  – Verify that the bin file was extracted by using the **ls -l** <*image_filename.bin*> UNIX command.

```
switch% ls -l image_filename.bin-rwxr-xr-x   1 bschuett eng      6365325 May 19 13:03
<insert path for lan base image>

-rw-r--r--   1 boba      3970586 Apr 21 12:00 image_name.bin
```

3. Connect your PC with terminal-emulation software supporting the Xmodem Protocol to the switch console port.

4. Set the line speed on the emulation software to 9600 baud.

5. Unplug the switch power cord.

6. Press the **Express Setup** buttonfactory default button and at the same time, reconnect the power cord to the switch.

You can release the button a second or two after the LED above port 1 goes offwhen the *password-recovery mechanism is enabled.*message appears. Several lines of information about the software appear along with instructions:

```
The system has been interrupted prior to initializing the flash file system. The following commands
will initialize the flash file system, and finish loading the operating system software#

flash_init
load_helper
boot
```

7. Initialize the flash file system:

```
switch: flash_init
```

8. If you had set the console port speed to anything other than 9600, it has been reset to that particular speed. Change the emulation software line speed to match that of the switch console port.

9. Load any helper files:

```
switch: load_helper
```

10. Start the file transfer by using the Xmodem Protocol.

```
switch: copy xmodem: flash:image_filename.bin
```

11. After the Xmodem request appears, use the appropriate command on the terminal-emulation software to start the transfer and to copy the software image into flash memory.

12. Boot the newly downloaded Cisco IOS image.

```
switch:boot flash:image_filename.bin
```

13. Use the **archive download-sw** privileged EXEC command to download the software image to the switch.

14. Use the **reload** privileged EXEC command to restart the switch and to verify that the new software image is operating properly.

15. Delete the flash:*image_filename.bin* file from the switch.

Recovering from a Lost or Forgotten Password

If you lose or forget your password, you can delete the switch password and set a new one.

Before you begin, make sure that:

- You have physical access to the switch.
- At least one switch port is enabled and is not connected to a device.

To delete the switch password and set a new one, follow these steps:

1. Press the **Express Setup** button until the SETUP LED blinks green and the LED of an available switch downlink port blinks green.

If no switch downlink port is available for your PC or laptop connection, disconnect a device from one of the switch downlink ports. Press the **Express Setup** button again until the SETUP LED and the port LED blink green.

2. Connect your PC or laptop to the port with the blinking green LED.

The SETUP LED and the switch downlink port LED stop blinking and stay solid green.

3. Press and hold the **Express Setup** button. Notice that the SETUP LED starts blinking green again. Continue holding the button until the SETUP LED turns solid green (approximately 5 seconds). Release the **Express Setup** button immediately.

This procedure deletes the password without affecting any other configuration settings. You can now access the switch without a password through the console port or by using Device Manager.

4. Enter a new password through the device manager by using the Express Setup window or through the command line interface by using the **enable secret** global configuration command.

Recovering from Lost Cluster Member Connectivity

Some configurations can prevent the command switch from maintaining contact with member switches. If you are unable to maintain management contact with a member, and the member switch is forwarding packets normally, check for these conflicts:

- A member switch (Catalyst 3750, Catalyst 3560, Catalyst 3550, Catalyst 3500 XL, Catalyst 2970, Catalyst 2960, Catalyst 2950, Catalyst 2900 XL, Catalyst 2820, and Catalyst 1900 switch) cannot connect to the command switch through a port that is defined as a network port.
- Catalyst 3500 XL, Catalyst 2900 XL, Catalyst 2820, and Catalyst 1900 member switches must connect to the command switch through a port that belongs to the same management VLAN.
- A member switch (Catalyst 3750, Catalyst 3560, Catalyst 3550, Catalyst 2970, Catalyst 2960, Catalyst 2950, Catalyst 3500 XL, Catalyst 2900 XL, Catalyst 2820, and Catalyst 1900 switch) connected to the command switch through a secured port can lose connectivity if the port is disabled because of a security violation.

Executing Ping

If you attempt to ping a host in a different IP subnetwork, you must define a static route to the network or have IP routing configured to route between those subnets.

IP routing is disabled by default on all switches. If you need to enable or configure IP routing, see Configuring Static IP Unicast Routing, page 695

Beginning in privileged EXEC mode, use this command to ping another device on the network from the switch:

Command	Purpose
ping ip *host* | *address*	Pings a remote host through IP or by supplying the hostname or network address.

Note: Other protocol keywords are available with the **ping** command, but they are not supported in this release.

This example shows how to ping an IP host:

```
Switch# ping 172.20.52.3

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echoes to 172.20.52.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
Switch#
```

Table 73 on page 1093 describes the possible ping character output.

Table 73 Ping Text Characters

Character	Description
!	Each exclamation point means receipt of a reply.
.	Each period means the network server timed out while waiting for a reply.
U	A destination unreachable error PDU was received.
C	A congestion experienced packet was received.
I	User interrupted test.
?	Unknown packet type.
&	Packet lifetime exceeded.

To end a ping session, enter the escape sequence (**Ctrl-^ X** by default). Simultaneously press and release the **Ctrl**, **Shift**, and **6** keys and then press the **X** key.

Executing IP Traceroute

Beginning in privileged EXEC mode, enter the following command to trace that the path packets take through the network:

Command	Purpose
traceroute ip *host*	Traces the path that packets take through the network.

Note: Other protocol keywords are available with the **traceroute** privileged EXEC command, but they are not supported in this release.

This example shows how to perform a **traceroute** to an IP host:

```
Switch# traceroute ip 171.9.15.10

Type escape sequence to abort.
Tracing the route to 171.69.115.10

  1 172.2.52.1 0 msec 0 msec 4 msec
  2 172.2.1.203 12 msec 8 msec 0 msec
  3 171.9.16.6 4 msec 0 msec 0 msec
  4 171.9.4.5 0 msec 4 msec 0 msec
  5 171.9.121.34 0 msec 4 msec 4 msec
  6 171.9.15.9 120 msec 132 msec 128 msec
  7 171.9.15.10 132 msec 128 msec 128 msec
Switch#
```

The display shows the hop count, the IP address of the router, and the round-trip time in milliseconds for each of the three probes that are sent.

Table 74 on page 1094 lists the characters that can appear in the traceroute command output.

Table 74 Traceroute Text Characters

Character	Description
*	The probe timed out.
?	Unknown packet type.
A	Administratively unreachable. Usually, this output means that an access list is blocking traffic.
H	Host unreachable.
N	Network unreachable.
P	Protocol unreachable.
Q	Source quench.
U	Port unreachable.

To end a trace in progress, enter the escape sequence (**Ctrl-^ X** by default). Simultaneously press and release the **Ctrl**, **Shift**, and **6** keys and then press the **X** key.

Running TDR and Displaying the Results

To run TDR, enter the **test cable-diagnostics tdr interface** *interface-id* privileged EXEC command:

To display the results, enter the **show cable-diagnostics tdr interface** *interface-id* privileged EXEC command.

Enabling Debugging on a Specific Feature

Caution: Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use debug commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. It is best to use debug commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased debug command processing overhead will affect system use.

All **debug** commands are entered in privileged EXEC mode, and most **debug** commands take no arguments. For example, beginning in privileged EXEC mode, enter this command to enable the debugging for Switched Port Analyzer (SPAN):

```
Switch# debug span-session
```

The switch continues to generate output until you enter the **no** form of the command.

If you enable a **debug** command and no output appears, consider these possibilities:

- The switch might not be properly configured to generate the type of traffic you want to monitor. Use the **show running-config** command to check its configuration.
- Even if the switch is properly configured, it might not generate the type of traffic you want to monitor during the particular period that debugging is enabled. Depending on the feature you are debugging, you can use commands such as the TCP/IP **ping** command to generate network traffic.

To disable debugging of SPAN, enter this command in privileged EXEC mode:

```
Switch# no debug span-session
```

Alternately, in privileged EXEC mode, you can enter the **undebug** form of the command:

```
Switch# undebug span-session
```

To display the state of each debugging option, enter this command in privileged EXEC mode:

```
Switch# show debugging
```

Enabling All-System Diagnostics

Beginning in privileged EXEC mode, enter this command to enable all-system diagnostics:

```
Switch# debug all
```

Caution: Because debugging output takes priority over other network traffic, and because the debug all privileged EXEC command generates more output than any other debug command, it can severely diminish switch performance or even render it unusable. In virtually all cases, it is best to use more specific debug commands.

The **no debug all** privileged EXEC command disables all diagnostic output. Using the **no debug all** command is a convenient way to ensure that you have not accidentally left any **debug** commands enabled.

Redirecting Debug and Error Message Output

By default, the network server sends the output from **debug** commands and system error messages to the console. If you use this default, you can use a virtual terminal connection to monitor debug output instead of connecting to the console port.

Possible destinations include the console, virtual terminals, internal buffer, and UNIX hosts running a syslog server. The syslog format is compatible with 4.3 Berkeley Standard Distribution (BSD) UNIX and its derivatives.

Note: Be aware that the debugging destination you use affects system overhead. Logging messages to the console produces very high overhead, whereas logging messages to a virtual terminal produces less overhead. Logging messages to a syslog server produces even less, and logging to an internal buffer produces the least overhead of any method.

For more information about system message logging, see Configuring System Message Logging, page 545

Monitoring Information

Physical Path

You can display the physical path that a packet takes from a source device to a destination device by using one of these privileged EXEC commands:

- **tracetroute mac** [**interface** *interface-id*] {*source-mac-address*} [**interface** *interface-id*] {*destination-mac-address*} [**vlan** *vlan-id*] [**detail**]
- **tracetroute mac ip** {*source-ip-address* | *source-hostname*}{*destination-ip-address* | *destination-hostname*} [**detail**]

SFP Module Status

You can check the physical or operational status of an SFP module by using the **show interfaces transceiver** privileged EXEC command. This command shows the operational status, such as the temperature and the current for an SFP module on a specific interface and the alarm status. You can also use the command to check the speed and the duplex settings on an SFP module.

Troubleshooting Examples

show platform forward Command

The output from the **show platform forward** privileged EXEC command provides some useful information about the forwarding results if a packet entering an interface is sent through the system. Depending upon the parameters entered about the packet, the output provides lookup table results and port maps used to calculate forwarding destinations, bitmaps, and egress information.

Most of the information in the output from the command is useful mainly for technical support personnel, who have access to detailed information about the switch application-specific integrated circuits (ASICs). However, packet forwarding information can also be helpful in troubleshooting.

This is an example of the output from the **show platform forward** command on port 1 in VLAN 5 when the packet entering that port is addressed to unknown MAC addresses. The packet should be flooded to all other ports in VLAN 5.

```
Switch# show platform forward GigabitEthernet1/17 vlan 5 1.1.1 2.2.2 ip 13.1.1.1 13.2.2.2 udp 10 20
Global Port Number:24, Asic Number:5
Src Real Vlan Id:5, Mapped Vlan Id:5

Ingress:
 Lookup                   Key-Used                    Index-Hit  A-Data
InptACL  40_0D020202_0D010101-00_40000014_000A0000     01FFA   03000000
L2Local  80_00050002_00020002-00_00000000_00000000     00C71   0000002B
Station Descriptor:02340000, DestIndex:0239, RewriteIndex:F005

==========================================
Egress:Asic 2, switch 1
Output Packets:

------------------------------------------
Packet 1
 Lookup                   Key-Used                    Index-Hit  A-Data
OutptACL 50_0D020202_0D010101-00_40000014_000A0000     01FFE   03000000

Port       Vlan      SrcMac          DstMac    Cos  Dscpv
Gi1/17    0005 0001.0001.0001  0002.0002.0002

------------------------------------------
Packet 2
 Lookup                   Key-Used                    Index-Hit  A-Data
OutptACL 50_0D020202_0D010101-00_40000014_000A0000     01FFE   03000000

Port       Vlan      SrcMac          DstMac    Cos  Dscpv
Gi1/17    0005 0001.0001.0001  0002.0002.0002

------------------------------------------
<output truncated>
------------------------------------------
Packet 10
 Lookup                   Key-Used                    Index-Hit  A-Data
OutptACL 50_0D020202_0D010101-00_40000014_000A0000     01FFE   03000000
Packet dropped due to failed DEJA_VU Check on Gi1/0/2
Packet dropped due to failed DEJA_VU Check on Gi1/18
```

This is an example of the output when the packet coming in on port 1 in VLAN 5 is sent to an address already learned on the VLAN on another port. It should be forwarded from the port on which the address was learned.

```
Switch# show platform forward GigabitEthernet1/17 vlan 5 1.1.1 0009.43a8.0145 ip 13.1.1.1 13.2.2.2 udp
10 20
Global Port Number:24, Asic Number:5
Src Real Vlan Id:5, Mapped Vlan Id:5
```

```
Ingress:
 Lookup                   Key-Used                    Index-Hit  A-Data
InptACL  40_0D020202_0D010101-00_40000014_000A0000     01FFA   03000000
L2Local  80_00050009_43A80145-00_00000000_00000000     00086   02010197
Station Descriptor:F0050003, DestIndex:F005, RewriteIndex:0003

==========================================
Egress:Asic 3, switch 1
Output Packets:

------------------------------------------
Packet 1
 Lookup                   Key-Used                    Index-Hit  A-Data
OutptACL 50_0D020202_0D010101-00_40000014_000A0000     01FFE   03000000

Port            Vlan      SrcMac          DstMac    Cos  Dscpv
interface-id    0005  0001.0001.0001  0009.43A8.0145
```

This is an example of the output when the packet coming in on port 1 in VLAN 5 has a destination MAC address set to the router MAC address in VLAN 5 and the destination IP address unknown. Because there is no default route set, the packet should be dropped.

```
Switch# show platform forward GigabitEthernet1/17 vlan 5 1.1.1 03.e319.ee44 ip 13.1.1.1 13.2.2.2 udp 10 
20
Global Port Number:24, Asic Number:5
Src Real Vlan Id:5, Mapped Vlan Id:5

Ingress:
 Lookup                   Key-Used                    Index-Hit  A-Data
InptACL  40_0D020202_0D010101-00_41000014_000A0000     01FFA   03000000
L3Local  00_00000000_00000000-90_00001400_0D020202     010F0   01880290
L3Scndr  12_0D020202_0D010101-00_40000014_000A0000     034E0   000C001D_00000000
Lookup Used:Secondary
Station Descriptor:02260000, DestIndex:0226, RewriteIndex:0000
```

This is an example of the output when the packet coming in on port 1 in VLAN 5 has a destination MAC address set to the router MAC address in VLAN 5 and the destination IP address set to an IP address that is in the IP routing table. It should be forwarded as specified in the routing table.

```
Switch# show platform forward GigabitEthernet1/17 vlan 5 1.1.1 03.e319.ee44 ip 110.1.5.5 16.1.10.5
Global Port Number:24, Asic Number:5
Src Real Vlan Id:5, Mapped Vlan Id:5

Ingress:
 Lookup                   Key-Used                    Index-Hit  A-Data
InptACL  40_10010A05_0A010505-00_41000014_000A0000     01FFA   03000000
L3Local  00_00000000_00000000-90_00001400_10010A05     010F0   01880290
L3Scndr  12_10010A05_0A010505-00_40000014_000A0000     01D28   30090001_00000000
Lookup Used:Secondary
Station Descriptor:F0070007, DestIndex:F007, RewriteIndex:0007

==========================================
Egress:Asic 3, switch 1
Output Packets:

------------------------------------------
Packet 1
 Lookup                   Key-Used                    Index-Hit  A-Data
OutptACL 50_10010A05_0A010505-00_40000014_000A0000     01FFE   03000000

Port       Vlan      SrcMac          DstMac    Cos  Dscpv
```

```
Gi1/18    0007 XXXX.XXXX.0246  0009.43A8.0147
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*
Additional troubleshooting information	*Hardware Installation Guide*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Using an SD Card

This document explains how to use a Secure Digital (SD) flash memory module (SD card) .

This document contains these sections:

Overview

The SD card can be used instead of the internal flash memory of the switch to update or restore configuration settings. In addition, the SD card can be used to boot the switch. You can also copy IOS software and switch configuration settings from a PC or from the switch to the SD card, and then use the SD card to copy this software and settings to other switches.

When an SD card is formatted on the switch, the card is formatted with the Disk Operating System Filing System (DOSFS), a platform-independent industry-standard file system that is supported on various Cisco switches and routers.

The switch does not support third-party SD cards or SD High Capacity (SDHC) cards. Attempting to operate the switch with a non-supported card causes the following message to be displayed:

```
WARNING: Non-IT SD flash detected.
Use of this card during normal operation can impact and
severely degrade performance of the system.
Please use supported SD flash cards only.
```

If the write-protect switch on the SD card is in the lock position, the switch can read data on the card and boot from the card, but updates and files cannot be written to the card.

Inserting and Removing an SD Card

To put an SD card in the switch, make sure that the card is oriented properly, and press it into the SD card slot on the switch until the card is seated. To remove the card, press it to release it, then pull it out of the slot.

The SD card is hot-swappable, but it should not be removed from the switch during the boot process or while sdflash write is in progress.

When an SD card is inserted, a syslog message similar to the following is logged:

```
Mar 30 01:38:51.965: %FLASH-6-DEVICE_INSERTED: Flash device inserted
```

When an SD card is removed, a syslog message similar to the following is logged:

```
Mar 30 01:39:12.467: %FLASH-1-DEVICE_REMOVED: Flash device removed
```

SD Card Operation

The SD card can be accessed by either the switch boot-loader or by the IOS. The following sections describe the operations that can be performed by the controlling software:

- Boot Loader Operation, page 1100
- IOS Operation, page 1100

Boot Loader Operation

The following boot loader commands can be executed on the SD card:

- boot—Load and boot an executable IOS image
- cat—Concatenate (type) file or files
- copy—Copy a file
- delete—Delete file of files
- dir—List files in directories
- fsck—Check file system consistency
- format—Format a file system
- mkdir—Create directories
- more—Concatenate (display) file
- rename—Rename a file
- rmdir—Delete empty directories
- sd_init—Initialize sd flash file systems

Important: The switch can be booted from its internal flash memory or from an SD card. The SD card takes precedence over internal flash memory. If an SD card is installed in the switch, the switch attempts to boot in the following order:

1. From the IOS image that is specified in the SD card system boot path
2. From the first IOS image in the SD card
3. From the IOS image that is specified in the internal flash memory system boot path
4. From the first IOS image in the internal flash

IOS Operation

You can insert or remove an SD card while the IOS is running. If you insert a supported Cisco SD card while the IOS is running, the switch validates the Cisco embedded string in the Product Name (PNM) field and displays the product number and the flash capacity of the SD card. If you remove an SD card while the IOS is running, the switch displays a warning message to alert you that the SD card has been removed.

If syslog is enabled, the system also sends a message when the SD card is inserted or removed.

When an SD card is installed in a switch, the following IOS commands operate as described:

- **write** command—Saves the running configuration. If the system boots from an SD card and you run a **write** command, the system saves the running configuration to the SD card, if the card is still installed. If the SD card has been removed, the system saves the running configuration to the internal flash memory and displays this message:

  ```
  WARNING: The SD flash is not present.
  The running-config is saved to the on-board flash.

  NOTE: This warning message is displayed only once.
  ```

  If the system boots from the internal flash memory and you then insert an SD card and run the **write** command, the system saves the running configuration to the internal flash memory.

- **boot** command—Lets you change the system boot parameters.

  If the system boots from an SD card and you run a **boot** command, the following behavior applies:

  - If the SD card is installed and the system boot path or configuration file path points to the SD card, the system boot path or configuration file path is saved to the SD card

  - If the SD card is installed and the system boot path or configuration file path points to the internal flash memory, the system boot path or configuration file path is saved to the internal flash memory

  - If the SD card has been removed and the system boot path or configuration file path points to the SD card, the system boot path or configuration file path is not saved and the following message displays:

    ```
    WARNING: The BOOT/config file path points to the
    SD flash card and the SD flash card is not present.
    The environment variable(s) is not saved.

    NOTE: This warning message is displayed only once.
    ```

  If the system boots from the internal flash memory and you then insert an SD card and run the **boot** command, the following behavior applies:

  - If the system boot path or configuration file path points to the internal flash memory, the system boot path or configuration file path is saved to the internal flash memory

  - If the system boot path or configuration file path points to the SD card, the system boot path or configuration file path is saved to the SD card and the following message is displayed:

    ```
    WARNING: The BOOT/config file path points to the
    SD flash card. The environment variable(s) is
    saved onto the SD flash card.

    NOTE: This warning message is displayed only once.
    ```

  - If the SD card has been removed and the system boot path or configuration file path points to the SD card, the system boot path or configuration file path is not saved and the following message is displayed:

    ```
    WARNING: The BOOT/config file path points to the
    SD flash card and the SD flash card is not present.
    The environment variable(s) is not saved.

    NOTE: This warning message is displayed only once.
    ```

- **sync** command—Copies the IOS image directory (which includes the IOS image file, FPGA image files, Device Manager files, and Profinet/CIP configuration files), the config.text IOS configuration file, the vlan.dat VLAN configuration file, and IOS boot parameters from the internal flash memory to the SD card or from the SD card to the internal flash memory. This command verifies that the IOS image is appropriate for the switch model and that enough destination flash memory is present, and aborts the sync process if a potential problem is detected. The **sync** command obtains the source IOS image directory path and source IOS configuration files path from the IOS boot parameters on the source flash device that is specified in the **sync** command. By default, this command overwrites the destination IOS image directory and IOS configuration files. The "save-old-files" option can be used to override this default behavior. If the running configuration has not been saved and you run the **sync** command, the switch provides the option for you to save the running configuration before the command executes.

The **sync** command options are:

- Switch# **sync flash: sdflash:**–Sync IOS image directory, configuration files, and boot parameters from internal flash memory to SD card.
- Switch# **sync sdflash: flash:**–Sync IOS image directory, configuration files, and boot parameters from SD card to internal flash memory.
- Switch# **sync flash: sdflash: ios-image-name *IOS_image_path***–Sync the designated IOS image directory, configuration files, and boot parameters from internal flash memory to SD card. For example, *IOS_image_path* might be f**lash:/ie2000-universalk9-mz.150-2.EA1/ie2000-universalk9-mz.150-2.EA1.bin**.
- Switch# **sync sdflash: flash: ios-image-name *IOS_image_path***–Sync the designated IOS image directory, configuration files, and boot parameters from SD card to internal flash memory. For example, *IOS_image_path* might be f**lash:/ie2000-universalk9-mz.150-2.EA1/ie2000-universalk9-mz.150-2.EA1.bin**.
- Switch# **sync flash: sdflash: skip config.text vlan.dat**–Sync only IOS image directory from internal flash memory to SD card.
- Switch# **sync sdflash: flash: skip config.text vlan.dat**–Sync only IOS image directory from SD card to internal flash memory.
- Switch# **sync flash: sdflash: skip ios-image**–Sync only IOS configuration files from internal flash memory to SD card.
- Switch# **sync sdflash: flash: skip ios-image**–Sync only IOS configuration files from SD card to internal flash memory.

SD Card Alarms

The switch supports the following SD card alarms:

- SD Card Alarm–Enabled when the SD card is removed and cleared when the SD card is inserted
- SD Card Unsupported Alarm–Enabled when an unsupported SD card is detected
- SD Card Corrupt Alarm–Enabled when an SD card DOSFS corruption is detected
- SD Card Files Corrupt Alarm–Enabled when the IOS image specified in the SD Card system boot path is corrupted

It takes approximately 2 minutes to trigger the alarm relay (LED output) after an SD card is inserted or removed.

You also can configure alarms and traps that are associated with the SD card alarm to be sent to syslog and the SNMP server.

Enabling SD Card Alarms

SD card alarms are disabled by default.

To use alarms, enter the **alarm facility sd-card enable** global configuration command to enable alarms, then enter the alarm f**acility sd-card** global configuration commands to associate the alarm to the relay:

Switch(config)# **alarm facility sd-card enable**

Switch(config)# **alarm facility sd-card notifies**

```
Switch(config)# alarm facility sd-card sysm
Switch(config)# alarm facility sd-card syslog
Switch(config)# alarm facility sd-card relay major
```

Clearing an SD Card Alarm

To clear the last SD card alarm warning state, enter the following command:

```
Switch# clear facility-alarm
```