Consolidated Platform Configuration Guide, Cisco IOS XE 15.2(2)E (Catalyst 2960-XR Switch)

CONTENTS

CHAPTER 20 **Configuring Flex Links and the MAC Address-Table Move Update Feature** **333**

PART XI   **VLAN** **1121**

CHAPTER 51   **Configuring VTP** **1123**

CHAPTER 55   **Configuring VMPS 1209**

CHAPTER 63

Preface

- Document Conventions, on page lxi
- Related Documentation, on page lxiii
- Obtaining Documentation and Submitting a Service Request, on page lxiii

Document Conventions

This document uses the following conventions:

Convention	Description
^ or Ctrl	Both the **^** symbol and Ctrl represent the Control (Ctrl) key on a keyboard. For example, the key combination **^D** or **Ctrl-D** means that you hold down the Control key while you press the D key. (Keys are indicated in capital letters but are not case sensitive.)
bold font	Commands and keywords and user-entered text appear in **bold** font.
Italic font	Document titles, new or emphasized terms, and arguments for which you supply values are in *italic* font.
`Courier` font	Terminal sessions and information the system displays appear in `courier` font.
`Bold Courier` font	**`Bold Courier`** font indicates text that the user must enter.
[x]	Elements in square brackets are optional.
...	An ellipsis (three consecutive nonbolded periods without spaces) after a syntax element indicates that the element can be repeated.
|	A vertical line, called a pipe, indicates a choice within a set of keywords or arguments.
[x | y]	Optional alternative keywords are grouped in brackets and separated by vertical bars.
{x | y}	Required alternative keywords are grouped in braces and separated by vertical bars.

Convention	Description
[x {y | z}]	Nested set of square brackets or braces indicate optional or required choices within optional or required elements. Braces and a vertical bar within square brackets indicate a required choice within an optional element.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.
< >	Nonprinting characters such as passwords are in angle brackets.
[ ]	Default responses to system prompts are in square brackets.
!, #	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.

Reader Alert Conventions

This document may use the following conventions for reader alerts:

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

Tip   Means *the following information will help you solve a problem.*

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

Timesaver   Means *the described action saves time.* You can save time by performing the action described in the paragraph.

Warning   IMPORTANT SAFETY INSTRUCTIONS

This warning symbol means danger. You are in a situation that could cause bodily injury. Before you work on any equipment, be aware of the hazards involved with electrical circuitry and be familiar with standard practices for preventing accidents. Use the statement number provided at the end of each warning to locate its translation in the translated safety warnings that accompanied this device. Statement 1071

SAVE THESE INSTRUCTIONS

Related Documentation

Note   Before installing or upgrading the switch, refer to the switch release notes.

- Catalyst 2960-XR Switch documentation, located at:

  http://www.cisco.com/go/cat2960xr_docs

- Cisco SFP and SFP+ modules documentation, including compatibility matrixes, located at:

  http://www.cisco.com/en/US/products/hw/modules/ps5455/tsd_products_support_series_home.html

- Error Message Decoder, located at:

  https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Obtaining Documentation and Submitting a Service Request

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

http://www.cisco.com/c/en/us/td/docs/general/whatsnew/whatsnew.html

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

Using the Command-Line Interface

Information About Using the Command-Line Interface

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.

You can start a CLI session through a console connection, through Telnet, an SSH, or by using the browser.

When you start a session, 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. Normally, 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 .

This table describes the main command modes, how to access each one, the prompt you see in that mode, and how to exit the mode.

Table 1: Command Mode Summary

Mode	Access Method	Prompt	Exit Method	About This Mode
User EXEC	Begin a session using Telnet, SSH, or console.	`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.
VLAN configuration	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.

Mode	Access Method	Prompt	Exit Method	About This Mode
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.

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

This table lists some error messages that you might encounter while using the CLI to configure your switch.

Table 2: 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 (?) without any 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 of 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 of 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.

Using the 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.

SUMMARY STEPS

1. **help**
2. *abbreviated-command-entry* **?**
3. *abbreviated-command-entry* <Tab>
4. **?**
5. *command* **?**
6. *command keyword* **?**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**help** **Example:** Switch# **help**	Obtains a brief description of the help system in any command mode.
Step 2	*abbreviated-command-entry* **?** **Example:** Switch# **di?** dir disable disconnect	Obtains a list of commands that begin with a particular character string.
Step 3	*abbreviated-command-entry* <Tab> **Example:** Switch# **sh conf**<tab> Switch# **show configuration**	Completes a partial command name.
Step 4	**?** **Example:** Switch> **?**	Lists all commands available for a particular command mode.
Step 5	*command* **?** **Example:** Switch> **show ?**	Lists the associated keywords for a command.
Step 6	*command keyword* **?** **Example:** Switch(config)# **cdp holdtime ?** <10-255> Length of time (in sec) that receiver must keep this packet	Lists the associated arguments for a keyword.

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.

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. This procedure is optional.

SUMMARY STEPS

    1. **terminal history** [**size** *number-of-lines*]

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**terminal history** [**size** *number-of-lines*] **Example:** Switch# **terminal history size 200**	Changes the number of command lines that the switch records during the current terminal session in privileged EXEC mode. You can configure the size from 0 to 256.

Recalling Commands

To recall commands from the history buffer, perform one of the actions listed in this table. These actions are optional.

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

SUMMARY STEPS

    1. **Ctrl-P** or use the **up arrow** key
    2. **Ctrl-N** or use the **down arrow** key
    3. **show history**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**Ctrl-P** or use the **up arrow** key	Recalls commands in the history buffer, beginning with the most recent command. Repeat the key sequence to recall successively older commands.
Step 2	**Ctrl-N** or use the **down arrow** key	Returns 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.
Step 3	**show history** **Example:** Switch# **show history**	Lists the last several commands that you just entered in privileged EXEC mode. The number of commands that appear is controlled by the setting of the **terminal history** global configuration command and the **history** line configuration command.

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. This procedure is optional.

SUMMARY STEPS

    1. **terminal no history**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**terminal no history**	Disables the feature during the current terminal session in privileged EXEC mode.
	Example:	
	Switch# **terminal no history**	

Enabling and Disabling Editing Features

Although enhanced editing mode is automatically enabled, you can disable it and reenable it.

SUMMARY STEPS

    1. **terminal editing**
    2. **terminal no editing**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**terminal editing**	Reenables the enhanced editing mode for the current terminal session in privileged EXEC mode.
	Example:	
	Switch# **terminal editing**	
Step 2	**terminal no editing**	Disables the enhanced editing mode for the current terminal session in privileged EXEC mode.
	Example:	
	Switch# **terminal no editing**	

Editing Commands Through Keystrokes

The keystrokes help you to edit the command lines. These keystrokes are optional.

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

Table 3: Editing Commands

Editing Commands	**Description**
Ctrl-B or use the **left arrow** key	Moves the cursor back one character.
Ctrl-F or use the **right arrow** key	Moves the cursor forward one character.

Ctrl-A	Moves the cursor to the beginning of the command line.
Ctrl-E	Moves the cursor to the end of the command line.
Esc B	Moves the cursor back one word.
Esc F	Moves the cursor forward one word.
Ctrl-T	Transposes the character to the left of the cursor with the character located at the cursor.
Delete or **Backspace** key	Erases the character to the left of the cursor.
Ctrl-D	Deletes the character at the cursor.
Ctrl-K	Deletes all characters from the cursor to the end of the command line.
Ctrl-U or Ctrl-X	Deletes all characters from the cursor to the beginning of the command line.
Ctrl-W	Deletes the word to the left of the cursor.
Esc D	Deletes from the cursor to the end of the word.
Esc C	Capitalizes at the cursor.
Esc L	Changes the word at the cursor to lowercase.
Esc U	Capitalizes letters from the cursor to the end of the word.
Ctrl-V or Esc Q	Designates a particular keystroke as an executable command, perhaps as a shortcut.
Return key	Scrolls 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.
Space bar	Scrolls down one screen.
Ctrl-L or Ctrl-R	Redisplays the current command line if the switch suddenly sends a message to your screen.

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.

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

The following example shows how to wrap a command line that extends beyond a single line on the screen.

SUMMARY STEPS

1. **access-list**
2. **Ctrl-A**
3. **Return** key

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**access-list** **Example:** `Switch(config)# ` **`access-list 101 permit tcp`** **`10.15.22.25 255.255.255.0 10.15.22.35`** `Switch(config)# $ ` **`101 permit tcp 10.15.22.25`** **`255.255.255.0 10.15.22.35 255.25`** `Switch(config)# $`**`t tcp 10.15.22.25 255.255.255.0`** **`131.108.1.20 255.255.255.0 eq`** `Switch(config)# $`**`15.22.25 255.255.255.0 10.15.22.35`** **`255.255.255.0 eq 45`**	Displays the global configuration command entry that extends beyond one line. When the cursor first reaches the end of the line, the line is shifted ten spaces to the left and redisplayed. The dollar sign ($) 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.
Step 2	**Ctrl-A** **Example:** `Switch(config)# ` **`access-list 101 permit tcp`** **`10.15.22.25 255.255.255.0 10.15.2$`**	Checks the complete syntax. The dollar sign ($) appears at the end of the line to show that the line has been scrolled to the right.
Step 3	**Return** key	Execute the commands. The software assumes that 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.

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.

SUMMARY STEPS

1. {**show** | **more**} *command* | {**begin** | **include** | **exclude**} *regular-expression*

DETAILED STEPS

	Command or Action	Purpose
Step 1	{**show**	**more**} *command*	{**begin**	**include**	**exclude**} *regular-expression* **Example:** `Switch# `**`show interfaces	include protocol`** `Vlan1 is up, line protocol is up` `Vlan10 is up, line protocol is down` `GigabitEthernet1/0/1 is up, line protocol is down` `GigabitEthernet1/0/2 is up, line protocol is up`	Searches and filters the output. 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.

Accessing the CLI on a Switch Stack

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

You manage the switch stack and the stack member interfaces through the active switchstack's active switch. You cannot manage stack members on an individual switch basis. You can connect to the active switchstack's active switch through the console port or the Ethernet management port of one or more stack members. Be careful with using multiple CLI sessions on the active switchstack's active switch. Commands that you enter in one session are not displayed in the other sessions. Therefore, it is possible to lose track of the session from which you entered commands.

✎

Note   We recommend using one CLI session when managing the switch stack.

If you want to configure a specific stack member port, you must include the stack member number in the CLI command interface notation.

To debug a specific stack member, you can start a CLI session from the stack master by using the **session** *stack-member-number* privileged EXEC command. The stack member number is appended to the system prompt. For example, *Switch-2#* is the prompt for stack member 2 where the system prompt for the stack master is Switch. Only the **show** and **debug** commands are available in a CLI session to a specific stack member. You can also use the **remote command** *stack-member-number LINE* privileged EXEC command on the stack master to enable debugging on a member switch without first starting a session.

Accessing the CLI Through a Console Connection or Through Telnet

Before you can access the CLI, you must connect a terminal or a PC to the switch console or connect a PC to the Ethernet management port and then power on the switch, as described in the hardware installation guide that shipped with your switch.

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.

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

Procedure

- Connect the switch console port to a management station or dial-up modem, or connect the Ethernet management port to a PC. For information about connecting to the console or Ethernet management port, see the switch hardware installation 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.

    - The switch supports up to five simultaneous secure SSH sessions.

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

PART I

Interface and Hardware Component

CHAPTER 2

Configuring Interface Characteristics

Information About Configuring Interface Characteristics

Interface Types

This section describes the different types of interfaces supported by the switch. The rest of the chapter describes configuration procedures for physical interface characteristics.

> **Note** The stack ports on the rear of the stacking-capable switches are not Ethernet ports and cannot be configured.

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 configured to be associated with the VLAN, when the VLAN Trunking Protocol (VTP) learns of its existence from a neighbor on a trunk, or when a user creates a VLAN. VLANs can be formed with ports across the stack.

To configure VLANs, use the **vlan** *vlan-id* global configuration command to enter VLAN configuration mode. The VLAN configurations for normal-range VLANs (VLAN IDs 1 to 1005) are saved in the VLAN database. If VTP is version 1 or 2, to configure extended-range VLANs (VLAN IDs 1006 to 4094), you must first set VTP mode to transparent. Extended-range VLANs created in transparent mode are not added to the VLAN

database but are saved in the switch running configuration. With VTP version 3, you can create extended-range VLANs in client or server mode. These VLANs are saved in the VLAN database.

In a switch stack, the VLAN database is downloaded to all switches in a stack, and all switches in the stack build the same VLAN database. The running configuration and the saved configuration are the same for all switches in a stack.

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 or a trunk port. You can configure a port as an access port or trunk port or let the Dynamic Trunking Protocol (DTP) operate on a per-port basis to set the switchport mode by negotiating with the port on the other end of the link. 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.

Access Ports

An access port belongs to and carries the traffic of only one VLAN (unless it is configured as a voice VLAN port). 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 a tagged packet (Inter-Switch Link [ISL] or IEEE 802.1Q tagged), the packet is dropped, and the source address is not learned.

The types of access ports supported are:

- Static access ports are manually assigned to a VLAN (or through a RADIUS server for use with IEEE 802.1x.

- VLAN membership of dynamic access ports is learned through incoming packets. By default, a dynamic access port is not a member of any VLAN, and forwarding to and from the port is enabled only when the VLAN membership of the port is discovered. Dynamic access ports on the switch are assigned to a VLAN by a VLAN Membership Policy Server (VMPS). The VMPS can be a Catalyst 6500 series switch; the switch cannot be a VMPS server.

You can also 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.

Trunk Ports

A trunk port carries the traffic of multiple VLANs and by default is a member of all VLANs in the VLAN database. These trunk port types are supported:

- In an ISL trunk port, all received packets are expected to be encapsulated with an ISL header, and all transmitted packets are sent with an ISL header. Native (non-tagged) frames received from an ISL trunk port are dropped.

- An IEEE 802.1Q trunk port supports simultaneous tagged and untagged traffic. An IEEE 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 every VLAN known to the VTP, 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 VTP knows of the VLAN and if the VLAN is in the enabled state. If VTP learns of a new, enabled VLAN and the VLAN is in the allowed list for a trunk port, the trunk port automatically becomes a member of that VLAN and traffic is forwarded to and from the trunk port for that VLAN. If VTP learns of a new, enabled VLAN that is not in the allowed list for a trunk port, the port does not become a member of the VLAN, and no traffic for the VLAN is forwarded to or from the port.

Tunnel Ports

Tunnel ports are used in IEEE 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 IEEE 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 IEEE 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.

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 DTP and 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.

Note   The IP Base image supports static routing and the Routing Information Protocol (RIP). For full Layer 3 routing or for fallback bridging, you must enable the IP Services image on the standalone switch, or the active 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. You can associate only one SVI with a VLAN. You configure an SVI for a VLAN only 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. 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 ISL or 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.

Although the switch stack or 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.

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

SVI Autostate Exclude

The line state of an SVI with multiple ports on a VLAN is in the *up* state when it meets these conditions:

- The VLAN exists and is active in the VLAN database on the switch

- The VLAN interface exists and is not administratively down.

- At least one Layer 2 (access or trunk) port exists, has a link in the *up* state on this VLAN, and is in the spanning-tree forwarding state on the VLAN.

Note   The protocol link state for VLAN interfaces come up when the first switchport belonging to the corresponding VLAN link comes up and is in STP forwarding state.

The default action, when a VLAN has multiple ports, is that the SVI goes down when all ports in the VLAN go down. You can use the SVI autostate exclude feature to configure a port so that it is not included in the SVI line-state up-or-down calculation. For example, if the only active port on the VLAN is a monitoring port, you might configure autostate exclude on that port so that the VLAN goes down when all other ports go down. When enabled on a port, **autostate exclude** applies to all VLANs that are enabled on that port.

The VLAN interface is brought up when one Layer 2 port in the VLAN has had time to converge (transition from STP listening-learning state to forwarding state). This prevents features such as routing protocols from using the VLAN interface as if it were fully operational and minimizes other problems.

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 DTP, the Cisco Discovery Protocol (CDP), and the Port Aggregation Protocol (PAgP), which operate only on physical 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.

Power over Ethernet Ports

A PoE-capable switch port automatically supplies power to one of these connected devices if the switch senses that there is no power on the circuit:

- a Cisco pre-standard powered device (such as a Cisco IP Phone or a Cisco Aironet Access Point)

- an IEEE 802.3af-compliant powered device

A powered device can receive redundant power when it is connected to a PoE switch port and to an AC power source. The device does not receive redundant power when it is only connected to the PoE port.

Using the Switch USB Ports

Theswitch has three USB ports on the front panel — a USB mini-Type B console port and two USB Type A ports.

USB Mini-Type B Console Port

The switch has the following console ports:

- USB mini-Type B console connection

- RJ-45 console port

Console output appears on devices connected to both ports, but console input is active on only one port at a time. By default, the USB connector takes precedence over the RJ-45 connector.

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. An 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 is active. Each switch in a stack issues this log. Every switch always first displays the RJ-45 media type.

In the sample output, Switch 1 has a connected USB console cable. Because the bootloader did not change to the USB console, the first log from Switch 1 shows the RJ-45 console. A short time later, the console changes and the USB console log appears. Switch 2 and Switch 3 have connected RJ-45 console cables.

```
switch-stack-1
*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.


switch-stack-2
*Mar  1 00:01:09.835: %USB_CONSOLE-6-MEDIA_RJ45: Console media-type is RJ45.

switch-stack-3
*Mar  1 00:01:10.523: %USB_CONSOLE-6-MEDIA_RJ45: Console media-type is RJ45.
```

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-stack-1
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.

USB Type A Ports

The USB Type A ports provide access to external USB flash devices, also known as thumb drives or USB keys. The switch supports Cisco 64 MB, 256 MB, 512 MB, 1 GB, 4 GB, and 8 GB flash drives. You can use standard Cisco IOS command- line interface (CLI) commands to read, write, erase, and copy to or from the flash device. You can also configure the switch to boot from the USB flash drive.

For information about configuring the switch to boot from a USB flash drive, refer to the *Catalyst 2960-XR Switch System Management Configuration Guide*.

For information about reading, writing, erasing, and copying files to or from the flash device, refer to the *Catalyst 2960-XR Switch Managing Cisco IOS Image Files Configuration Guide*.

Interface Connections

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 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: Connecting VLANs with the Switch
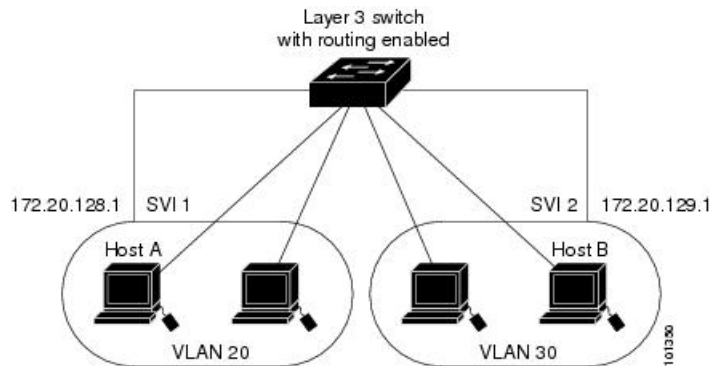
The IP Lite imageIP Lite feature set supports static routing and RIP. The IP Base image supports only static routing. The routing function can be enabled on all SVIs and routed ports. The switch routes only IP traffic. When IP routing protocol parameters and address configuration are added to an SVI or routed port, any IP traffic received from these ports is routed.

Note   There is no limit to the number of static routes on switches running the IP Lite image.

Note   Switches running the LAN Base image support configuring only 16 static routes on SVIs.

Interface Configuration Mode

The switch supports these interface types:

- Physical ports—switch ports and routed ports

- VLANs—switch virtual interfaces

- Port channels—EtherChannel interfaces

You can also configure a range of interfaces.

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

- Type—Gigabit Ethernet (gigabitethernet or gi) for 10/100/1000 Mb/s Ethernet ports, or small form-factor pluggable (SFP) module Gigabit Ethernet interfaces (gigabitethernet or gi).

- Stack member number—The number that identifies the switch within the stack. The range is 1 to 8 for a stack of Catalyst 2960-X switches, and 1 to 4 for a mixed stack of Catalyst 2960-XR, Catalyst 2960-X, and Catalyst 2960-S switches. The switch number is assigned the first time the switch initializes. The default switch number, before it is integrated into a switch stack, is 1. When a switch has been assigned a stack member number, it keeps that number until another is assigned to it.

  You can use the switch port LEDs in Stack mode to identify the stack member number of a switch.

- Stack member number—The number that identifies the switch within the stack. The range is 1 to 8 for a stack of Catalyst 2960-XR switches. The switch number is assigned the first time the switch initializes. The default switch number, before it is integrated into a switch stack, is 1. When a switch has been assigned a stack member number, it keeps that number until another is assigned to it.

  You can use the switch port LEDs in Stack mode to identify the stack member number of a switch.

- Module number—The module or slot number on the switch (always 0).

- Port number—The interface number on the switch. The 10/100/1000 port numbers always begin at 1, starting with the far left port when facing the front of the switch, for example, gigabitethernet1/0/1 or gigabitethernet1/0/8. For a switch with 10/100/1000 ports and SFP module ports, SFP module ports are numbered consecutively following the 10/100/1000 ports.

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.

These are examples of how to identify interfaces on a stacking-capable switch:

- To configure 10/100/1000 port 4 on a standalone switch, enter this command:

```
Switch(config)# interface gigabitethernet1/0/4
```

- To configure 10/100/1000 port 4 on stack member 3, enter this command:

```
Switch(config)# interface gigabitethernet3/0/4
```

Default Ethernet Interface Configuration

This table shows the Ethernet interface default configuration, including some features that apply only to Layer 2 interfaces.

Table 4: Default Layer 2 Ethernet Interface Configuration

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.
Native VLAN (for IEEE 802.1Q trunks)	VLAN 1.
802.1p priority-tagged traffic	Drop all packets tagged with VLAN 0.
VLAN trunking	Switchport mode dynamic auto (supports DTP).
Port enable state	All ports are enabled.
Port description	None defined.

Feature	Default Setting
Speed	Autonegotiate. (Not supported on the 10-Gigabit interfaces.)
Duplex mode	Autonegotiate. (Not supported on the 10-Gigabit interfaces.)
Flow control	Flow control is set to **receive: off**. It is always off for sent packets.
EtherChannel (PAgP)	Disabled on all Ethernet ports.
Port blocking (unknown multicast and unknown unicast traffic)	Disabled (not blocked).
Broadcast, multicast, and unicast storm control	Disabled.
Protected port	Disabled.
Port security	Disabled.
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 IEEE 802.3af—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).
Keepalive messages	Disabled on SFP module ports; enabled on all other ports.

Interface Speed and Duplex Mode

Ethernet interfaces on the switch operate at 10, 100, 1000, or 10,000 Mb/s 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-Mb/s ports operate in half-duplex mode, which means that stations can either receive or send traffic.

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

Speed and Duplex Configuration Guidelines

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

- The 10-Gigabit Ethernet ports do not support the speed and duplex features. These ports operate only at 10,000 Mb/s and in full-duplex mode.

- Do not disable Auto-Negotiation on PoE switches.

- Gigabit Ethernet (10/100/1000-Mb/s) ports support all speed options and all duplex options (auto, half, and full). However, Gigabit Ethernet ports operating at 1000 Mb/s do not support half-duplex mode.

- For SFP module ports, the speed and duplex CLI options change depending on the SFP module type:

    - The 1000BASE-*x* (where -*x* is -BX, -CWDM, -LX, -SX, and -ZX) SFP module ports support the **nonegotiate** keyword in the **speed** interface configuration command. Duplex options are not supported.

    - The 1000BASE-T SFP module ports support the same speed and duplex options as the 10/100/1000-Mb/s ports.

- If both ends of the line support autonegotiation, we highly recommend the default setting of **auto** negotiation.

- 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.

- As best practice, we suggest configuring the speed and duplex options on a link to auto or to fixed on both the ends. If one side of the link is configured to auto and the other side is configured to fixed, the link will not be up and this is expected.

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

IEEE 802.3x Flow Control

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** The switch 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 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**: 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.

Note   For details on the command settings and the resulting flow control resolution on local and remote ports, see
the **flowcontrol** interface configuration command in the command reference for this release.

Layer 3 Interfaces

The switch supports these types of 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.

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

  When configuring SVIs, you can also configure SVI autostate exclude on a port in the SVI to exclude
  that port from being included in determining SVI line-state status.

- 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 or in a
switch stack. 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 its 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 switchport.

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

- If the switch is notified by VLAN Trunking Protocol (VTP) of a new VLAN, 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.

- 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

How to Configure Interface Characteristics

Configuring Interfaces

These general instructions apply to all interface configuration processes.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** **Example:** Switch(config)# **interface gigabitethernet1/0/1** Switch(config-if)#	Identifies the interface type, the switch number (only on stacking-capable switches), and the number of the connector. **Note** You do not need to add a space between the interface type and the interface number. For example, in the preceding line, you can specify either **gigabitethernet 1/0/1**, **gigabitethernet1/0/1**, **gi 1/0/1**, or **gi1/0/1**.
Step 4	Follow each **interface** command with the interface configuration commands that the interface requires.	Defines 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.

	Command or Action	Purpose
Step 5	**interface range** or **interface range macro**	(Optional) Configures a range of interfaces.
		Note Interfaces configured in a range must be the same type and must be configured with the same feature options.
Step 6	**show interfaces**	Displays 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.

Adding a Description for an Interface

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **description** *string*
5. **end**
6. **show interfaces** *interface-id* **description**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/2**	Specifies the interface for which you are adding a description, and enter interface configuration mode.
Step 4	**description** *string* **Example:**	Adds a description (up to 240 characters) for an interface.

	Command or Action	Purpose
	Switch(config-if)# **description Connects to Marketing**	
Step 5	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 6	**show interfaces** *interface-id* **description**	Verifies your entry.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring a Range of Interfaces

To configure multiple interfaces with the same configuration parameters, use the **interface range** global configuration command. 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.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface range** {*port-range* | **macro** *macro_name*}
4. **end**
5. **show interfaces** [*interface-id*]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**interface range** {*port-range* | **macro** *macro_name*} **Example:** Switch(config)# **interface range macro**	Specifies 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. • The **macro** variable is explained in the Configuring and Using Interface Range Macros, on page 29. • 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. **Note** Use the normal configuration commands to apply the configuration parameters to all interfaces in the range. Each command is executed as it is entered.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show interfaces** [*interface-id*] **Example:** Switch# **show interfaces**	Verifies the configuration of the interfaces in the range.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

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.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **define interface-range** *macro_name interface-range*
4. **interface range macro** *macro_name*
5. **end**
6. **show running-config | include define**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**define interface-range** *macro_name interface-range* **Example:** Switch(config)# **define interface-range enet_list** **gigabitethernet1/0/1 - 2**	Defines 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. **Note** 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.
Step 4	**interface range macro** *macro_name* **Example:** Switch(config)# **interface range macro enet_list**	Selects 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.
Step 5	**end** **Example:**	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Switch(config)# **end**	
Step 6	**show running-config | include define** **Example:** Switch# **show running-config | include define**	Shows the defined interface range macro configuration.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Ethernet Interfaces

Setting the Interface Speed and Duplex Parameters

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **speed** {**10** | **100** | **1000** | **2500** | **5000** | **10000** | **auto** [**10** | **100** | **1000** | **2500** | **5000** | **10000**] | **nonegotiate**}
5. **duplex** {**auto** | **full** | **half**}
6. **end**
7. **show interfaces** *interface-id*
8. **copy running-config startup-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.

	Command or Action	**Purpose**
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/3**	Specifies the physical interface to be configured, and enter interface configuration mode.
Step 4	**speed** {**10** | **100** | **1000** | **2500** | **5000** | **10000** | **auto** [**10** | **100** | **1000** | **2500** | **5000** | **10000**] | **nonegotiate**} **Example:** Switch(config-if)# **speed 10**	Enter the appropriate speed parameter for the interface: • Enter **10**, **100**, **1000 2500**, **5000**, or **10000** to set a specific speed for the interface. • Enter **auto** to enable the interface to autonegotiate speed with the connected device. If you specify a speed and also set 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 Mb/s but can be configured to not negotiate if connected to a device that does not support autonegotiation.
Step 5	**duplex** {**auto** | **full** | **half**} **Example:** Switch(config-if)# **duplex half**	This command is not available on a 10-Gigabit Ethernet interface. Enter the duplex parameter for the interface. Enable half-duplex mode (for interfaces operating only at 10 or 100 Mb/s). You cannot configure half-duplex mode for interfaces operating at 1000 Mb/s. You can configure the duplex setting when the speed is set to **auto**.
Step 6	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 7	**show interfaces** *interface-id* **Example:** Switch# **show interfaces gigabitethernet1/0/3**	Displays the interface speed and duplex mode configuration.
Step 8	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

	Command or Action	**Purpose**
Step 9	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring IEEE 802.3x Flow Control

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **flowcontrol** {**receive**} {**on** | **off** | **desired**}
4. **end**
5. **show interfaces** *interface-id*
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode
Step 2	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	Specifies the physical interface to be configured, and enter interface configuration mode.
Step 3	**flowcontrol** {**receive**} {**on**	**off**	**desired**} **Example:** Switch(config-if)# **flowcontrol receive on**	Configures the flow control mode for the port.
Step 4	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 5	**show interfaces** *interface-id* **Example:**	Verifies the interface flow control settings.

	Command or Action	**Purpose**
	Switch# **show interfaces gigabitethernet1/0/1**	
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Layer 3 Interfaces

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** {**gigabitethernet** *interface-id*} | {**vlan** *vlan-id*} | {**port-channel** *port-channel-number*}
4. **no switchport**
5. **ip address** *ip_address subnet_mask*
6. **no shutdown**
7. **end**
8. **show interfaces** [*interface-id*]
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** {**gigabitethernet** *interface-id*}	{**vlan** *vlan-id*}	{**port-channel** *port-channel-number*} **Example:** Switch(config)# **interface gigabitethernet1/0/2**	Specifies the interface to be configured as a Layer 3 interface, and enter interface configuration mode.

	Command or Action	Purpose
Step 4	**no switchport** **Example:** Switch(config-if)# **no switchport**	For physical ports only, enters Layer 3 mode.
Step 5	**ip address** *ip_address subnet_mask* **Example:** Switch(config-if)# **ip address 192.20.135.21 255.255.255.0**	Configures the IP address and IP subnet.
Step 6	**no shutdown** **Example:** Switch(config-if)# **no shutdown**	Enables the interface.
Step 7	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 8	**show interfaces** [*interface-id*]	Verifies the configuration.
Step 9	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring SVI Autostate Exclude

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport autostate exclude**
5. **end**
6. **show running config interface** *interface-id*
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** Example: Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* Example: Switch(config)# **interface gigabitethernet1/0/2**	Specifies a Layer 2 interface (physical port or port channel), and enter interface configuration mode.
Step 4	**switchport autostate exclude** Example: Switch(config-if)# **switchport autostate exclude**	Excludes the access or trunk port when defining the status of an SVI line state (up or down)
Step 5	**end** Example: Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 6	**show running config interface** *interface-id*	(Optional) Shows the running configuration. Verifies the configuration.
Step 7	**copy running-config startup-config** Example: Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

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.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** {**vlan** *vlan-id*} | { **gigabitethernet***interface-id*} | {**port-channel** *port-channel-number*}
4. **shutdown**
5. **no shutdown**
6. **end**
7. **show running-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** {**vlan** *vlan-id*}	{ **gigabitethernet***interface-id*}	{**port-channel** *port-channel-number*} **Example:** Switch(config)# **interface gigabitethernet1/0/2**	Selects the interface to be configured.
Step 4	**shutdown** **Example:** Switch(config-if)# **shutdown**	Shuts down an interface.
Step 5	**no shutdown** **Example:** Switch(config-if)# **no shutdown**	Restarts an interface.
Step 6	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.

	Command or Action	**Purpose**
Step 7	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.

Configuring the Console Media Type

Follow these steps to set the console media type to RJ-45. If you configure the console as RJ-45, USB console operation is disabled, and input comes only through the RJ-45 connector.

This configuration applies to all switches in a stack.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **line console 0**
4. **media-type rj45**
5. **end**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**line console 0** **Example:** Switch(config)# **line console 0**	Configures the console and enters line configuration mode.

	Command or Action	Purpose
Step 4	**media-type rj45** **Example:** Switch(config-line)# **media-type rj45**	Configures the console media type to be only RJ-45 port. If you do not enter this command and both types are connected, the USB port is used by default.
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring the USB Inactivity Timeout

The configurable inactivity timeout reactivates the RJ-45 console port if the USB console port is activated but no input activity occurs on it for a specified time period. When the USB console port is deactivated due to a timeout, you can restore its operation by disconnecting and reconnecting the USB cable.

> ✎
>
> **Note** The configured inactivity timeout applies to all switches in a stack. However, a timeout on one switch does not cause a timeout on other switches in the stack.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **line console 0**
4. **usb-inactivity-timeout** *timeout-minutes*
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	**Purpose**
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**line console 0** **Example:** Switch(config)# **line console 0**	Configures the console and enters line configuration mode.
Step 4	**usb-inactivity-timeout** *timeout-minutes* **Example:** Switch(config-line)# **usb-inactivity-timeout 30**	Specify an inactivity timeout for the console port. The range is 1 to 240 minutes. The default is to have no timeout configured.
Step 5	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Monitoring Interface Characteristics

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 5: Show Commands for Interfaces

Command	**Purpose**
show interfaces *interface-id* **status** [**err-disabled**]	Displays interface status or a list of interfaces in the error-disabled state.
show interfaces [*interface-id*] **switchport**	Displays administrative and operational status of switching (nonrouting) ports. You can use this command to find out if a port is in routing or in switching mode.
show interfaces [*interface-id*] **description**	Displays the description configured on an interface or all interfaces and the interface status.
show ip interface [*interface-id*]	Displays the usability status of all interfaces configured for IP routing or the specified interface.

Command	Purpose
show interface [*interface-id*] **stats**	Displays the input and output packets by the switching path for the interface.
show interfaces *interface-id*	(Optional) Displays speed and duplex on the interface.
show interfaces transceiver dom-supported-list	(Optional) Displays Digital Optical Monitoring (DOM) status on the connect SFP modules.
show interfaces transceiver properties	(Optional) Displays temperature, voltage, or amount of current on the interface.
show interfaces [*interface-id*] [{**transceiver properties** | **detail**}] *module number*]	Displays physical and operational status about an SFP module.
show running-config interface [*interface-id*]	Displays the running configuration in RAM for the interface.
show version	Displays the hardware configuration, software version, the names and sources of configuration files, and the boot images.
show controllers ethernet-controller *interface-id* **phy**	Displays the operational state of the auto-MDIX feature on the interface.

Clearing and Resetting Interfaces and Counters

Table 6: Clear Commands for Interfaces

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

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.

Configuration Examples for Interface Characteristics

Configuring a Range of Interfaces: Examples

This example shows how to use the **interface range** global configuration command to set the speed to 100 Mb/s on ports 1 to 4 on switch 1:

```
Switch# configure terminal
Switch(config)# interface range gigabitethernet1/0/1 - 4
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 Gigabit Ethernet ports 1 to 3 and 10-Gigabit Ethernet ports 1 and 2 to receive flow-control pause frames:

```
Switch# configure terminal
Switch(config)# interface range gigabitethernet1/0/1 - 3 , tengigabitethernet1/0/1 - 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 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: Examples

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

```
Switch# configure terminal
Switch(config)# define interface-range enet_list gigabitethernet1/0/1 - 2
Switch(config)# end
Switch# show running-config | include define
define interface-range enet_list GigabitEthernet1/0/1 - 2
```

This example shows how to create a multiple-interface macro named *macro1*:

```
Switch# configure terminal
Switch(config)# define interface-range macro1 gigabitethernet1/0/1 - 2, gigabitethernet1/0/5
 - 7, tengigabitethernet1/0/1 -2
Switch(config)# 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#
```

Setting Interface Speed and Duplex Mode: Example

This example shows how to set the interface speed to 100 Mb/s and the duplex mode to half on a 10/100/1000 Mb/s port:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/3
Switch(config-if)# speed 10
Switch(config-if)# duplex half
```

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

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# speed 100
```

Configuring Layer 3 Interfaces: Example

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# no switchport
Switch(config-if)# ip address 192.20.135.21 255.255.255.0
Switch(config-if)# no shutdown
```

Configuring the Console Media Type: Example

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
```

This configuration terminates any active USB console media type in the stack. A log shows that this termination has occurred. This example shows that the console on switch 1 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.
```

At this point no switches in the stack allow a USB console to have input. A log entry shows when a console cable is attached. If a USB console cable is connected to switch 2, 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. (switch-stk-2)
```

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

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

Configuring the USB Inactivity Timeout: Example

This example configures the inactivity timeout to 30 minutes:

```
Switch# configure terminal
Switch(config)# line console 0
Switch(config-line)# usb-inactivity-timeout 30
```

To disable the configuration, use these commands:

```
Switch# configure terminal
Switch(config)# line console 0
Switch(config-line)# no usb-inactivity-timeout
```

If there is no (input) activity on a USB console port for the configured number of minutes, the inactivity timeout setting applies to the RJ-45 port, and a log shows this occurrence:

```
*Mar  1 00:47:25.625: %USB_CONSOLE-6-INACTIVITY_DISABLE: Console media-type USB disabled
due to inactivity, media-type reverted to RJ45.
```

At this point, the only way to reactivate the USB console port is to disconnect and reconnect the cable.

When the USB cable on the switch has been disconnected and reconnected, a log similar to this appears:

```
*Mar  1 00:48:28.640: %USB_CONSOLE-6-MEDIA_USB: Console media-type is USB.
```

Additional References for the Interface Characteristics Feature

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	--

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Configuring Interface Characteristics

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Configuring Auto-MDIX

Prerequisites for Auto-MDIX

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.

Automatic medium-dependent interface crossover (auto-MDIX) is enabled by default.

Auto-MDIX is supported on all 10/100/1000-Mb/s and on 10/100/1000BASE-TX small form-factor pluggable (SFP)-module interfaces. It is not supported on 1000BASE-SX or -LX SFP module interfaces.

Restrictions for Auto-MDIX

The switch might not support a pre-standard powered device—such as Cisco IP phones and access points that do not fully support IEEE 802.3af—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.

Information about Configuring Auto-MDIX

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.

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

Table 7: Link Conditions and Auto-MDIX Settings

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

How to Configure Auto-MDIX

Configuring Auto-MDIX on an Interface

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **speed auto**
5. **duplex auto**
6. **end**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	**Purpose**
	Switch> **enable**	
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	Specifies the physical interface to be configured, and enter interface configuration mode.
Step 4	**speed auto** **Example:** Switch(config-if)# **speed auto**	Configures the interface to autonegotiate speed with the connected device.
Step 5	**duplex auto** **Example:** Switch(config-if)# **duplex auto**	Configures the interface to autonegotiate duplex mode with the connected device.
Step 6	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Example for Configuring Auto-MDIX

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

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/1
```

```
Switch(config-if)# speed auto
Switch(config-if)# duplex auto
Switch(config-if)# mdix auto
Switch(config-if)# end
```

Additional References

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Auto-MDIX

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

CHAPTER 4

Configuring Ethernet Management Port

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Ethernet Management Ports

When connecting a PC to the Ethernet management port, you must first assign an IP address.

Information about the Ethernet Management Port

The Ethernet management port, also referred to as the *Fa0* or *fastethernet0* port, is a Layer 3 host port to which you can connect a PC. You can use the Ethernet management port instead of the switch console port for network management. When managing a switch stack, connect the PC to the Ethernet management port on a stack member.

Ethernet Management Port Direct Connection to a Switch

Figure 2: Connecting a Switch to a PC

This figure displays how to connect the Ethernet management port to the PC for a switch or a standalone
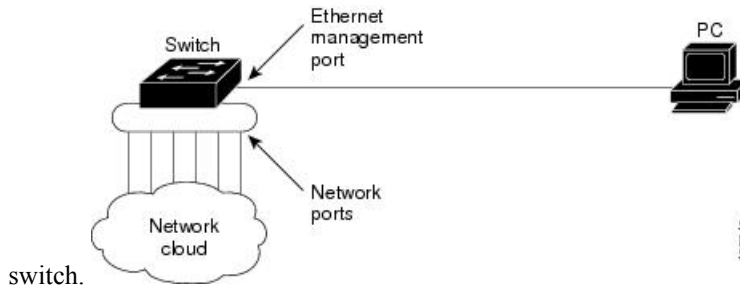
switch.

Ethernet Management Port Connection to Stack Switches using a Hub

In a stack with only stack switches, all the Ethernet management ports on the stack members are connected to a hub to which the PC is connected. The active link is from the Ethernet management port on the active switchstack's active switchthrough the hub, to the PC. If the activeswitch fails and a new active switch is elected, the active link is now from the Ethernet management port on the new active switch to the PC.

Figure 3: Connecting a Switch Stack to a PC

This figure displays how a PC uses a hub to connect to a switch stack.
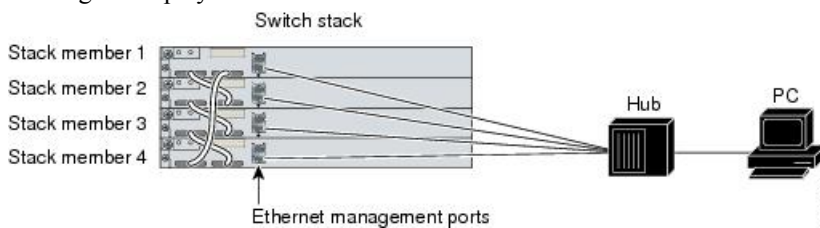
Ethernet Management Port and Routing

By default, the Ethernet management port is enabled. The switch cannot route packets from the Ethernet management port to a network port, and the reverse. Even though the Ethernet management port does not support routing, you may need to enable routing protocols on the port.

Figure 4: Network Example with Routing Protocols Enabled

Enable routing protocols on the Ethernet management port when the PC is multiple hops away from the switch and the packets must pass through multiple Layer 3 devices to reach the PC.
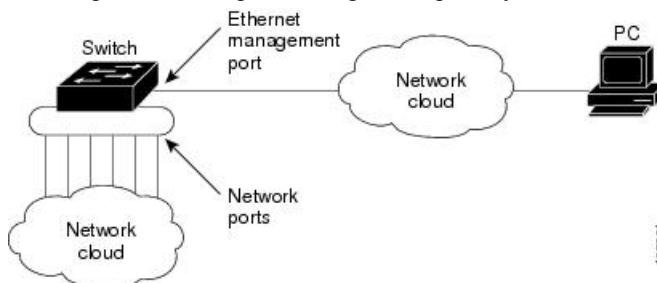
In the above figure , if the Ethernet management port and the network ports are associated with the same routing process, the routes are propagated as follows:

- The routes from the Ethernet management port are propagated through the network ports to the network.

- The routes from the network ports are propagated through the Ethernet management port to the network.

Because routing is not supported between the Ethernet management port and the network ports, traffic between these ports cannot be sent or received. If this happens, data packet loops occur between the ports, which disrupt the switch and network operation. To prevent the loops, configure route filters to avoid routes between the Ethernet management port and the network ports.

Supported Features on the Ethernet Management Port

The Ethernet management port supports these features:

- Express Setup (only in switch stacks)

- Network Assistant

- Telnet with passwords

- TFTP

- Secure Shell (SSH)

- DHCP-based autoconfiguration

- SMNP (only the ENTITY-MIB and the IF-MIB)

- IP ping

- Interface features

   - Speed—10 Mb/s, 100 Mb/s, and autonegotiation

   - Duplex mode—Full, half, and autonegotiation

   - Loopback detection

- Cisco Discovery Protocol (CDP)

- DHCP relay agent

- IPv4 and IPv6 access control lists (ACLs)

- Routing protocols

⚠️

Caution   Before enabling a feature on the Ethernet management port, make sure that the feature is supported. If you try to configure an unsupported feature on the Ethernet Management port, the feature might not work properly, and the switch might fail.

How to Configure the Ethernet Management Port

Disabling and Enabling the Ethernet Management Port

SUMMARY STEPS

1. **configure terminal**
2. **interface fastethernet0**
3. **shutdown**
4. **no shutdown**
5. **exit**
6. **show interfaces fastethernet0**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface fastethernet0** **Example:** Switch(config)# **interface fastethernet0**	Specifies the Ethernet management port in the CLI.
Step 3	**shutdown** **Example:** Switch(config-if)# **shutdown**	Disables the Ethernet management port.
Step 4	**no shutdown** **Example:** Switch(config-if)# **no shutdown**	Enables the Ethernet management port.
Step 5	**exit** **Example:** Switch(config-if)# **exit**	Exits interface configuration mode.
Step 6	**show interfaces fastethernet0** **Example:** Switch# **show interfaces fastethernet0**	Displays the link status. To find out the link status to the PC, you can monitor the LED for the Ethernet management port. The LED is green (on) when the link is active, and the LED is off when the link is down. The LED is amber when there is a POST failure.

What to do next

Proceed to manage or configure your switch using the Ethernet management port. Refer to the *Catalyst 2960-XR Switch Network Management Configuration Guide*.

Additional References

Related Documents

Related Topic	Document Title
Bootloader configuration	*Catalyst 2960-XR Switch System Management Configuration Guide*
Bootloader commands	*Catalyst 2960-XR Switch System Management Command Reference*

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Ethernet Management Ports

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Configuring LLDP, LLDP-MED, and Wired Location Service

LLDP, LLDP-MED, and Wired Location Service Overview

LLDP

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, switches, and controllers). 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 Supported TLVs

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)

LLDP and Cisco Switch Stacks

A switch stack appears as a single switch in the network. Therefore, LLDP discovers the switch stack, not the individual stack members.

LLDP and Cisco Medianet

When you configure LLDP or CDP location information on a per-port basis, remote devices can send Cisco Medianet location information to the switch. For information, go to http://www.cisco.com/en/US/docs/ios/netmgmt/configuration/guide/nm_cdp_discover.html.

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 Supported TLVs

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-signaling 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.

LLDP-MED also supports an extended power TLV to advertise fine-grained power requirements, end-point power priority, and end-point and network connectivity-device power status. LLDP is enabled and power is applied to a port, the power TLV determines the actual power requirement of the endpoint device so that the system power budget can be adjusted accordingly. The switch processes the requests and either grants or denies power based on the current power budget. If the request is granted, the switch updates the power budget. If the request is denied, the switch turns off power to the port, generates a syslog message, and updates the power budget. If LLDP-MED is disabled or if the endpoint does not support the LLDP-MED power TLV, the initial allocation value is used throughout the duration of the connection.

You can change power settings by entering the **power inline** {**auto** [**max** *max-wattage*] | **never** | **static** [**max** *max-wattage*]} interface configuration command. By default the PoE interface is in **auto** mode; If no value is specified, the maximum is allowed (30 W).

- 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 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

Table 8: 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

Feature	Default Setting
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.

Restrictions for LLDP

- 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.

- When Cisco Discovery Protocol and LLDP are both in use within the same switch, it is necessary to disable LLDP on interfaces where Cisco Discover Protocol is in use for power negotiation. LLDP can be disabled at interface level with the commands **no lldp tlv-select power-management** or **no lldp transmit / no lldp receive**.

How to Configure LLDP, LLDP-MED, and Wired Location Service

Enabling LLDP

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **lldp run**
4. **interface** *interface-id*
5. **lldp transmit**
6. **lldp receive**
7. **end**
8. **show lldp**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode.
	Example:	- Enter your password if prompted.

	Command or Action	**Purpose**
	Switch> **enable**	
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**lldp run** **Example:** Switch (config)# **lldp run**	Enables LLDP globally on the switch.
Step 4	**interface** *interface-id* **Example:** Switch (config)# **interface gigabitethernet2/0/1**	Specifies the interface on which you are enabling LLDP, and enter interface configuration mode.
Step 5	**lldp transmit** **Example:** Switch(config-if)# **lldp transmit**	Enables the interface to send LLDP packets.
Step 6	**lldp receive** **Example:** Switch(config-if)# **lldp receive**	Enables the interface to receive LLDP packets.
Step 7	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 8	**show lldp** **Example:** Switch# **show lldp**	Verifies the configuration.
Step 9	**copy running-config startup-config** **Example:**	(Optional) Saves your entries in the configuration file.

Command or Action	Purpose
Switch# **copy running-config startup-config**	

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.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **lldp holdtime** *seconds*
4. **lldp reinit** *delay*
5. **lldp timer** *rate*
6. **lldp tlv-select**
7. **interface** *interface-id*
8. **lldp med-tlv-select**
9. **end**
10. **show lldp**
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**lldp holdtime** *seconds* **Example:** Switch(config)# **lldp holdtime 120**	(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.

	Command or Action	Purpose
Step 4	**lldp reinit** *delay* **Example:** Switch(config)# **lldp reinit 2**	(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.
Step 5	**lldp timer** *rate* **Example:** Switch(config)# **lldp timer 30**	(Optional) Sets the sending frequency of LLDP updates in seconds. The range is 5 to 65534 seconds; the default is 30 seconds.
Step 6	**lldp tlv-select** **Example:** Switch(config)# **tlv-select**	(Optional) Specifies the LLDP TLVs to send or receive.
Step 7	**interface** *interface-id* **Example:** Switch (config)# **interface gigabitethernet2/0/1**	Specifies the interface on which you are enabling LLDP, and enter interface configuration mode.
Step 8	**lldp med-tlv-select** **Example:** Switch (config-if)# **lldp med-tlv-select inventory management**	(Optional) Specifies the LLDP-MED TLVs to send or receive.
Step 9	**end** **Example:** Switch (config-if)# **end**	Returns to privileged EXEC mode.
Step 10	**show lldp** **Example:** Switch# **show lldp**	Verifies the configuration.
Step 11	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring 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, as well. When the LLDP-MED entry has been aged out, it again only sends LLDP packets.

By using the **lldp** interface configuration command, you can configure the interface not to send the TLVs listed in the following table.

Table 9: LLDP-MED TLVs

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

Follow these steps to enable a TLV on an interface:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **lldp med-tlv-select**
5. **end**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch (config)# **interface**	Specifies the interface on which you are enabling LLDP, and enter interface configuration mode.

	Command or Action	**Purpose**
	`gigabitethernet 2/0/1`	
Step 4	**lldp med-tlv-select** **Example:** Switch(config-if)# **lldp med-tlv-select** **inventory management**	Specifies the TLV to enable.
Step 5	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Network-Policy TLV

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **network-policy profile** *profile number*
4. {**voice** | **voice-signaling**} **vlan** [*vlan-id* {**cos** *cvalue* | **dscp** *dvalue*}] | [[**dot1p** {**cos** *cvalue* | **dscp** *dvalue*}] | **none** | **untagged**]
5. **exit**
6. **interface** *interface-id*
7. **network-policy** *profile number*
8. **lldp med-tlv-select network-policy**
9. **end**
10. **show network-policy profile**
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	**Purpose**
	Switch> **enable**	
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**network-policy profile** *profile number* **Example:** Switch(config)# **network-policy profile 1**	Specifies the network-policy profile number, and enter network-policy configuration mode. The range is 1 to 4294967295.
Step 4	{**voice** | **voice-signaling**} **vlan** [*vlan-id* {**cos** *cvalue* | **dscp** *dvalue*}] | [[**dot1p** {**cos** *cvalue* | **dscp** *dvalue*}] | **none** | **untagged**] **Example:** Switch(config-network-policy)# **voice vlan 100 cos 4**	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 4094. • **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 5. • **dscp** *dvalue*—(Optional) Specifies the differentiated services code point (DSCP) value for the configured VLAN. The range is 0 to 63; the default is 46. • **dot1p**—(Optional) Configures the telephone to use IEEE 802.1p priority tagging and use VLAN 0 (the native VLAN). • **none**—(Optional) Do 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. • **untagged**—(Optional) Configures the telephone to send untagged voice traffic. This is the default for the telephone.

	Command or Action	Purpose
Step 5	**exit** **Example:** Switch(config)# **exit**	Returns to global configuration mode.
Step 6	**interface** *interface-id* **Example:** Switch (config)# **interface gigabitethernet2/0/1**	Specifies the interface on which you are configuring a network-policy profile, and enter interface configuration mode.
Step 7	**network-policy** *profile number* **Example:** Switch(config-if)# **network-policy 1**	Specifies the network-policy profile number.
Step 8	**lldp med-tlv-select network-policy** **Example:** Switch(config-if)# **lldp med-tlv-select network-policy**	Specifies the network-policy TLV.
Step 9	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 10	**show network-policy profile** **Example:** Switch# **show network-policy profile**	Verifies the configuration.
Step 11	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Location TLV and Wired Location Service

Beginning in privileged EXEC mode, follow these steps to configure location information for an endpoint and to apply it to an interface.

SUMMARY STEPS

1. **configure terminal**
2. **location** {**admin-tag** *string* | **civic-location identifier** {*id* | **host**} | **elin-location** *string* **identifier** *id* | **custom-location identifier** {*id* | **host**} | **geo-location identifier** {*id* | **host**}}
3. **exit**
4. **interface** *interface-id*
5. **location** {**additional-location-information** *word* | **civic-location-id** {*id* | **host**} | **elin-location-id** *id* | **custom-location-id** {*id* | **host**} | **geo-location-id** {*id* | **host**} }
6. **end**
7. Use one of the following:

    • **show location admin-tag** *string*
    • **show location civic-location identifier** *id*
    • **show location elin-location identifier** *id*

8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**location** {**admin-tag** *string*	**civic-location identifier** {*id*	**host**}	**elin-location** *string* **identifier** *id*	**custom-location identifier** {*id*	**host**}	**geo-location identifier** {*id*	**host**}} **Example:** 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**	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). • **custom-location**—Specifies custom location information. • **geo-location**—Specifies geo-spatial location information. • **identifier** *id*—Specifies the ID for the civic, ELIN, custom, or geo location. • **host**—Specifies the host civic, custom, or geo location. • *string*—Specifies the site or location information in alphanumeric format.
Step 3	**exit** **Example:**	Returns to global configuration mode.

	Command or Action	**Purpose**
	Switch(config-civic)# **exit**	
Step 4	**interface** *interface-id* **Example:** Switch (config)# **interface gigabitethernet2/0/1**	Specifies the interface on which you are configuring the location information, and enter interface configuration mode.
Step 5	**location** {**additional-location-information** *word* | **civic-location-id** {*id* | **host**} | **elin-location-id** *id* | **custom-location-id** {*id* | **host**} | **geo-location-id** {*id* | **host**} } **Example:** Switch(config-if)# **location elin-location-id 1**	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. • **custom-location-id**—Specifies custom location information for an interface. • **geo-location-id**—Specifies geo-spatial location information for an interface. • **host**—Specifies the host location identifier. • *word*—Specifies a word or phrase with additional location information. • *id*—Specifies the ID for the civic, ELIN, custom, or geo location. The ID range is 1 to 4095.
Step 6	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 7	Use one of the following: • **show location admin-tag** *string* • **show location civic-location identifier** *id* • **show location elin-location identifier** *id* **Example:** Switch# **show location admin-tag** or Switch# **show location civic-location**	Verifies the configuration.

	Command or Action	Purpose
	`identifier` or Switch# **show location elin-location** **identifier**	
Step 8	**copy running-config startup-config** **Example:** Switch# **copy running-config** **startup-config**	(Optional) Saves your entries in the configuration file.

Enabling Wired Location Service on the Switch

Before you begin

For wired location to function, you must first enter the **ip device tracking** global configuration command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **nmsp notification interval** {**attachment** | **location**} *interval-seconds*
4. **end**
5. **show network-policy profile**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**nmsp notification interval** {**attachment**	**location**} *interval-seconds* **Example:**	Specifies the NMSP notification interval. **attachment**—Specifies the attachment notification interval.

	Command or Action	Purpose
	Switch(config)# **nmsp notification interval location 10**	**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.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show network-policy profile** **Example:** Switch# **show network-policy profile**	Verifies the configuration.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuration Examples for LLDP, LLDP-MED, and Wired Location Service

Configuring Network-Policy TLV: Examples

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 1
Switch(config-network-policy)# voice vlan 100 cos 4
Switch(config-network-policy)# exit
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# network-policy profile 1
Switch(config-if)# lldp med-tlv-select network-policy
```

This example shows how to configure the voice application type for the native VLAN with priority tagging:

```
Switchconfig-network-policy)# voice vlan dot1p cos 4
Switchconfig-network-policy)# voice vlan dot1p dscp 34
```

Monitoring and Maintaining LLDP, LLDP-MED, and Wired Location Service

Commands for 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

Additional References for LLDP, LLDP-MED, and Wired Location Service

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for LLDP, LLDP-MED, and Wired Location Service

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Configuring System MTU

Information about the MTU

The default maximum transmission unit (MTU) size for frames received and sent on all switch interfaces is 1500 bytes.

You can change the MTU size to support switched jumbo frames on all Gigabit Ethernet and 10-Gigabit Ethernet interfaces and to support routed frames on all routed ports.

Restrictions for System MTU

When configuring the system MTU values, follow these guidelines:

- The switch does not support the MTU on a per-interface basis.

How to Configure MTU

Configuring the System MTU

Beginning in privileged EXEC mode, follow these steps to change the MTU size for switched and routed packets:

SUMMARY STEPS

1. **configure terminal**
2. **system mtu** *bytes*
3. **system mtu jumbo** *bytes*
4. **system mtu routing** *bytes*

5. **end**
6. **copy running-config startup-config**
7. **show system mtu**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**system mtu** *bytes* **Example:** Switch(config)# **system mtu 2500**	(Optional) Changes the MTU size for all Fast Ethernet interfaces on the switch. The range is 1500 to 1998 bytes; the default is 1500 bytes.
Step 3	**system mtu jumbo** *bytes* **Example:** Switch(config)# **system mtu jumbo 7500**	(Optional) Changes the MTU size for all Gigabit Ethernet and 10-Gigabit Ethernet interfaces on the switch or the switch stack. The range is 1500 to 9198 bytes; the default is 1500 bytes.
Step 4	**system mtu routing** *bytes* **Example:** Switch(config)# **system mtu routing 2000**	(Optional) Changes the system MTU for routed ports. You can also set the maximum MTU to be advertised by the routing protocols that support the configured MTU size. The system routing MTU is the maximum MTU for routed packets and is also the maximum MTU that the switch advertises in routing updates for protocols such as OSPF. **Note** This command is not supported on switches running the LAN base feature set.
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	Saves your entries in the configuration file.
Step 7	**show system mtu** **Example:** Switch# **show system mtu**	Verifies your settings.

Configuration Examples for System MTU

This example shows how to set the maximum packet size for a Gigabit Ethernet port to 7500 bytes:

```
Switch(config)# system mtu 7500
Switch(config)# exit
```

This example shows how to set the jumbo packet size for a Gigabit Ethernet port to 7500 bytes:

```
Switch(config)# system mtu jumbo 7500
Switch(config)# exit
```

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

Additional References for System MTU

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for System MTU

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

CHAPTER 7

Configuring Internal Power Supplies

Information About Internal Power Supplies

See the switch installation guide for information about the power supplies.

How to Configure Internal Power Supplies

Configuring Internal Power Supply

You can use the **power supply** EXEC command to configure and manage the internal power supply on the switch. The switch does not support the **no power supply** EXEC command.

Follow these steps beginning in user EXEC mode:

SUMMARY STEPS

1. **power supply** *switch_number* **slot**{**A** | **B**} { **off** | **on** }
2. **show environment power**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**power supply** *switch_number* **slot**{**A** | **B**} { **off** | **on** } **Example:** Switch# **power supply 1 slot A on**	Sets the specified power supply to **off** or **on** by using one of these keywords: • **A** —Selects the power supply in slot A. • **B** —Selects power supply in slot B.

Command or Action	Purpose
	Note Power supply slot B is the closest to the outer edge of the switch. • **off** —Set the power supply off. • **on** —Set the power supply on. By default, the switch power supply is **on**.
Step 2	**show environment power** **Example:** Switch# **show environment power**	Verifies your settings.

Monitoring Internal Power Supplies

Table 10: Show Commands for Power Supplies

Command	Purpose
show environment power [**all** | **switch** *switch_number*]	(Optional) Displays the status of the internal power supplies for each switch in the stack or for the specified switch. The range is 1 to 9, depending on the switch member numbers in the stack. The switch keywords are available only on stacking-capable switches.

Configuration Examples for Internal Power Supplies

This example shows how to set the power supply in slot A to off:

```
Switch# power supply 1 slot A off
Disabling Power supply A may result in a power loss to PoE devices and/or switches ...
Continue? (yes/[no]): yes
Switch#
Jun 10 04:52:54.389: %PLATFORM_ENV-6-FRU_PS_OIR: FRU Power Supply 1 powered off
Jun 10 04:52:56.717: %PLATFORM_ENV-1-FAN_NOT_PRESENT: Fan is not present
Switch#
```

This example shows how to set the power supply in slot A to on:

```
Switch# power supply 1 slot A on
Jun 10 04:54:39.600: %PLATFORM_ENV-6-FRU_PS_OIR: FRU Power Supply 1 powered on
```

This example shows the output of the **show env power** command:

Table 11: show env power Status Descriptions

Field	Description
OK	The power supply is present and power is good.
Not Present	No power supply is installed.
No Input Power	The power supply is present but there is no input power.
Disabled	The power supply and input power are present, but power supply is switched off by CLI.
Not Responding	The power supply is not recognizable or is faulty.
Failure-Fan	The power supply fan is faulty.

Additional References

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Internal Power Supplies

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Configuring PoE

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information about PoE

Power over Ethernet Ports

A PoE-capable switch port automatically supplies power to one of these connected devices if the switch senses that there is no power on the circuit:

- a Cisco pre-standard powered device (such as a Cisco IP Phone or a Cisco Aironet Access Point)

- an IEEE 802.3af-compliant powered device

A powered device can receive redundant power when it is connected to a PoE switch port and to an AC power source. The device does not receive redundant power when it is only connected to the PoE port.

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-upon 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—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:

- 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 12: IEEE Power Classifications, on page 84 lists these levels.

Table 12: IEEE Power Classifications

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	30 W (For IEEE 802.3at Type 2 powered devices)

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). Theswitch 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 *CDP-specific* power consumption requirement of the connected Cisco powered devices, which is the amount of power to allocate based on the CDP messages. 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.

With PoE+, powered devices use IEEE 802.3at and LLDP power with media dependent interface (MDI) type, length, and value descriptions (TLVs), Power-via-MDI TLVs, for negotiating power up to 30 W. Cisco pre-standard devices and Cisco IEEE powered devices can use CDP or the IEEE 802.3at power-via-MDI power negotiation mechanism to request power levels up to 30 W.

Note   The initial allocation for Class 0, Class 3, and Class 4 powered devices is 15.4 W. When a device starts up and uses CDP or LLDP to send a request for more than 15.4 W, it can be allocated up to the maximum of 30 W.

Note   The CDP-specific power consumption requirement is referred to as the *actual* power consumption requirement in the software configuration guides and command references.

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.

The PoE feature operates the same whether or not the switch is a stack member. The power budget is per switch and independent of any other switch in the stack. Election of a new active switch does not affect PoE operation. The active switch keeps track of the PoE status for all switches and ports in the stack and includes the status in output displays.

Power Management Modes

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 is consuming more than the maximum wattage, the switch shuts down the powered device.

  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 that power is never applied to a PoE-capable port, making the port a data-only port.

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

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.

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. The maximum power consumption is also referred to as the *cutoff power* on a PoE port.

   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 .

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.

You should use **power inline consumption default** *wattage* command to manually set the power level for a port only in situations where CDP/LLDP power negotiations are not supported.

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.

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 allocated amount, which could adversely affect the switch and the devices connected to the other PoE ports.

How to Configure PoE

Configuring a Power Management Mode on a PoE 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.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **power inline** {**auto** [**max** *max-wattage*] | **never** | **static** [**max** *max-wattage*]}
5. **end**
6. **show power inline** [*interface-id* | **module** *switch-number*]
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet2/0/1**	Specifies the physical port to be configured, and enters interface configuration mode.

	Command or Action	**Purpose**
Step 4	**power inline** {**auto** [**max** *max-wattage*]	**never**	**static** [**max** *max-wattage*]} **Example:** Switch(config-if)# **power inline auto**	Configures the PoE mode on the port. The keywords have these meanings: **auto**—Enables powered-device detection. If enough power is available, automatically allocates power to the PoE port after device detection. This is the default setting. **max** *max-wattage*—Limits the power allowed on the port. If no value is specified, the maximum is allowed. **max** *max-wattage*—Limits the power allowed on the port. If no value is specified, the maximum is allowed. **never** —Disables 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 the error-disabled state. **static**—Enables 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.
Step 5	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 6	**show power inline** [*interface-id*	**module** *switch-number*] **Example:** Switch# **show power inline**	Displays PoE status for a switch or a switch stack, for the specified interface, or for a specified stack member. The **module** *switch-number* keywords are supported only on stacking-capable switches.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves 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 *CDP-specific* power consumption of the devices, and the switch adjusts the power budget accordingly. This 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 15,400 mW for the device, regardless of the CDP-specific amount of power needed. If the powered device reports a higher class than its CDP-specific 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* interface configuration command or the **power inline consumption default** *wattage* global 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, enable the power monitoring feature, 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.

Budgeting Power to All PoE ports

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no cdp run**
4. **power inline consumption default** *wattage*
5. **end**
6. **show power inline consumption default**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	**Purpose**
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**no cdp run** **Example:** Switch(config)# **no cdp run**	(Optional) Disables CDP.
Step 4	**power inline consumption default** *wattage* **Example:** Switch(config)# **power inline consumption default 5000**	Configures the power consumption of powered devices connected to each PoE port. The range for each device is 4000 to 30000 mW (PoE+). The default is 30000 mW. **Note** When you use this command, we recommend you also enable power policing.
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**show power inline consumption default** **Example:** Switch# **show power inline consumption default**	Displays the power consumption status.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Budgeting Power to a Specific PoE Port

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no cdp run**
4. **interface** *interface-id*
5. **power inline consumption** *wattage*
6. **end**
7. **show power inline consumption**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** Example: Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 3	**no cdp run** Example: Switch(config)# **no cdp run**	(Optional) Disables CDP.
Step 4	**interface** *interface-id* Example: Switch(config)# **interface gigabitethernet2/0/1**	Specifies the physical port to be configured, and enter interface configuration mode.
Step 5	**power inline consumption** *wattage* Example: Switch(config-if)# **power inline consumption 5000**	Configures the power consumption of a powered device connected to a PoE port on the switch. The range for each device is 4000 to 30000 mW (PoE+). The default is 30000 mW (PoE+).
Step 6	**end** Example: Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 7	**show power inline consumption** Example: Switch# **show power inline consumption**	Displays the power consumption data.
Step 8	**copy running-config startup-config** Example: Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Power Policing

By default, the switch monitors the real-time power consumption of connected powered devices. You can configure the switch to police the power usage. By default, policing is disabled.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **power inline police** [**action**{**log** | **errdisable**}]
5. **exit**
6. Use one of the following:

   - **errdisable detect cause inline-power**
   - **errdisable recovery cause inline-power**
   - **errdisable recovery interval** *interval*

7. **exit**
8. Use one of the following:

   - **show power inline police**
   - **show errdisable recovery**

9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet2/0/1**	Specifies the physical port to be configured, and enter interface configuration mode.
Step 4	**power inline police** [**action**{**log**	**errdisable**}] **Example:** Switch(config-if)# **power inline police**	If the real-time power consumption exceeds the maximum power allocation on the port, configures the switch to take one of these actions: • **power inline police**—Shuts down the PoE port, turns off power to it, and puts it in the error-disabled state.

	Command or Action	**Purpose**
		Note You can enable error detection for the PoE error-disabled cause by using the **errdisable detect cause inline-power** global configuration command. You can also enable the timer to recover from the PoE error-disabled state by using the **errdisable recovery cause inline-power interval** *interval* global configuration command.
		• **power inline police action errdisable**—Turns off power to the port if the real-time power consumption exceeds the maximum power allocation on the port.
		• **power inline police action log**—Generates a syslog message while still providing power to the port.
		If you do not enter the **action log** keywords, the default action shuts down the port and puts the port in the error-disabled state.
Step 5	**exit** **Example:** Switch(config-if)# **exit**	Returns to global configuration mode.

Step 6	Use one of the following:
	
	    • **errdisable detect cause inline-power**
	    • **errdisable recovery cause inline-power**
	    • **errdisable recovery interval** *interval*
	
	Example:
	Switch(config)# **errdisable detect cause inline-power**
	
	Switch(config)# **errdisable recovery cause inline-power**
	
	Switch(config)# **errdisable recovery interval 100**
		(Optional) Enables error recovery from the PoE error-disabled state, and configures the PoE recover mechanism variables.
		
		By default, the recovery interval is 300 seconds.
		
		For **interval** *interval*, specifies the time in seconds to recover from the error-disabled state. The range is 30 to 86400.

Step 7	**exit** **Example:** Switch(config)# **exit**	Returns to privileged EXEC mode.

Step 8	Use one of the following:
	
	    • **show power inline police**
	    • **show errdisable recovery**
	
	Example:
		Displays the power monitoring status, and verify the error recovery settings.

	Command or Action	Purpose
	Switch# **show power inline police**	
	Switch# **show errdisable recovery**	
Step 9	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Monitoring Power Status

Table 13: Show Commands for Power Status

Command	Purpose
show env power switch [*switch-number*]	(Optional) Displays the status of the internal power supplies for each switch in the stack or for the specified switch. The range is 1 to 8, depending on the switch member numbers in the stack. These keywords are available only on stacking-capable switches.
show power inline [*interface-id* | **module** *switch-number*]	Displays PoE status for a switch or switch stack, for an interface, or for a specific switch in the stack.
show power inline police	Displays the power policing data.

Configuration Examples for Configuring PoE

Budgeting Power: Example

When you enter one of the following commands,

- [**no**] **power inline consumption default** *wattage* global configuration command
- [**no**] **power inline consumption** *wattage*

  interface configuration command

this caution message appears:

```
%CAUTION: Interface Gi1/0/1: 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.
```

Additional References

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Configuring EEE

Restrictions for EEE

EEE has the following restrictions:

- Changing the EEE configuration resets the interface because the device has to restart Layer 1 autonegotiation.

- You might want to enable the Link Layer Discovery Protocol (LLDP) for devices that require longer wakeup times before they are able to accept data on their receive paths. Doing so enables the device to negotiate for extended system wakeup times from the transmitting link partner.

Information About EEE

EEE Overview

Energy Efficient Ethernet (EEE) is an IEEE 802.3az standard that is designed to reduce power consumption in Ethernet networks during idle periods.

EEE can be enabled on devices that support low power idle (LPI) mode. Such devices can save power by entering LPI mode during periods of low utilization. In LPI mode, systems on both ends of the link can save power by shutting down certain services. EEE provides the protocol needed to transition into and out of LPI mode in a way that is transparent to upper layer protocols and applications.

Default EEE Configuration

EEE is disabled by default.

How to Configure EEE

You can enable or disable EEE on an interface that is connected to an EEE-capable link partner.

Enabling or Disabling EEE

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **power efficient-ethernet auto**
4. **no power efficient-ethernet auto**
5. **end**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	Specifies the interface to be configured, and enter interface configuration mode.
Step 3	**power efficient-ethernet auto** **Example:** Switch(config-if)# **power efficient-ethernet auto**	Enables EEE on the specified interface. When EEE is enabled, the device advertises and autonegotiates EEE to its link partner.
Step 4	**no power efficient-ethernet auto** **Example:** Switch(config-if)# **no power efficient-ethernet auto**	Disables EEE on the specified interface.

	Command or Action	Purpose
Step 5	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Monitoring EEE

Table 14: Commands for Displaying EEE Settings

Command	Purpose
show eee capabilities interface *interface-id*	Displays EEE capabilities for the specified interface.
show eee status interface *interface-id*	Displays EEE status information for the specified interface.

Configuration Examples for Configuring EEE

This example shows how to enable EEE for an interface:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# power efficient-ethernet auto
```

This example shows how to disable EEE for an interface:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# no power efficient-ethernet auto
```

Additional References

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Configuring EEE

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

PART II

First Hop Protocols

Configuring HSRP and VRRP

Configuring HSRP and VRRP

This chapter describes how to use Hot Standby Router Protocol (HSRP) to provide routing redundancy for routing IP traffic without being dependent on the availability of any single router.

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.

Virtual Router Redundancy Protocol (VRRP) is supported for IPv4 on switches running IP Lite image.

Information About Configuring HSRP

HSRP Overview

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 automatically enabled for the interface.

You can configure multiple Hot Standby groups among switches and switch stacks 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.

The following figure 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 5: Typical HSRP Configuration
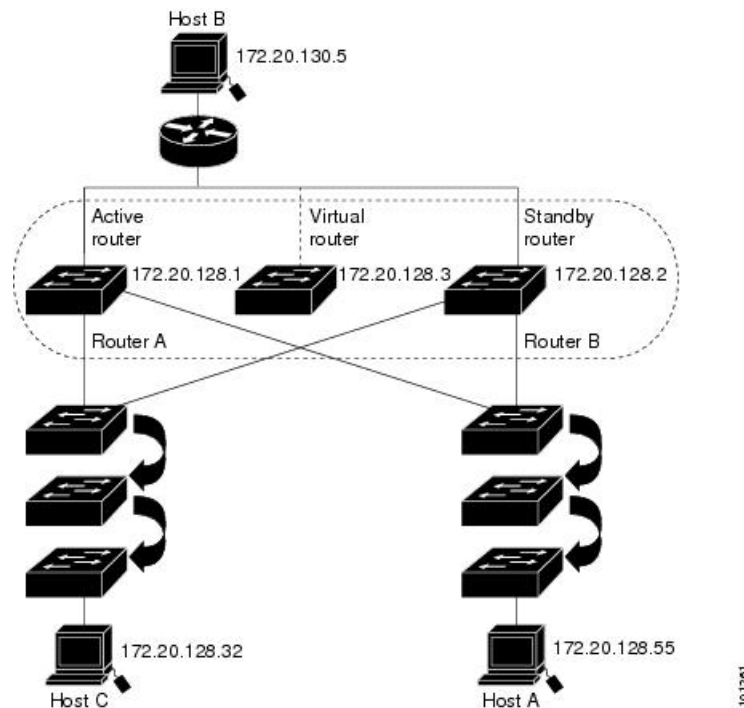
HSRP Versions

and later support these Hot Standby Router Protocol (HSRP) versions:

The switch supports these 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 groupnumber 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 the figure below, 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 6: MHSRP Load Sharing
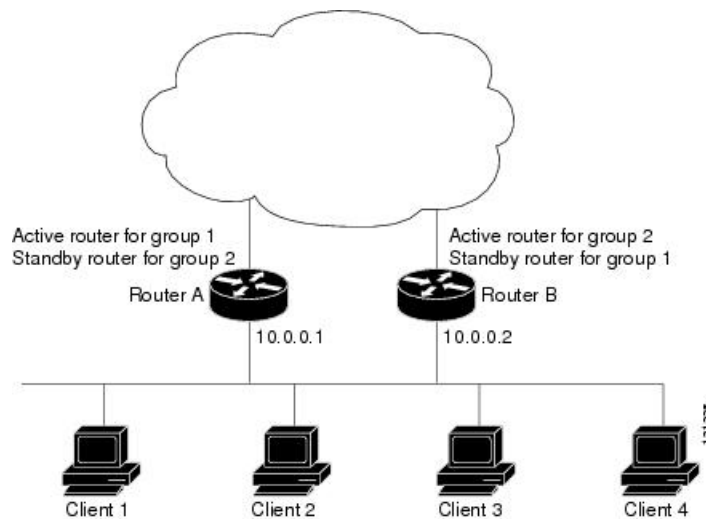
SSO HSRP

SSO HSRP alters the behavior of HSRP when a device with redundant Route Processors (RPs) is configured for stateful switchover (SSO) redundancy mode. When an RP is active and the other RP is standby, SSO enables the standby RP to take over if the active RP fails.

With this functionality, HSRP SSO information is synchronized to the standby RP, allowing traffic that is sent using the HSRP virtual IP address to be continuously forwarded during a switchover without a loss of data or a path change. Additionally, if both RPs fail on the active HSRP device, then the standby HSRP device takes over as the active HSRP device.

The feature is enabled by default when the redundancy mode of operation is set to SSO.

HSRP and Switch Stacks

HSRP hello messages are generated by the active switch. If HSRP fails on the active switch, a flap in the HSRP active state might occur. This is because HSRP hello messages are not generated while a new active switch is elected and initialized, and the standby switch might become active after the active switch fails.

How to Configure HSRP

Default HSRP Configuration

Table 15: Default HSRP Configuration

Feature	Default Setting
HSRP version	Version 1
HSRP groups	None configured
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

- HSRPv2 and HSRPv1 are mutually exclusive. HSRPv2 is not interoperable with HSRPv1 on an interface and the reverse.

- In the procedures, the specified interface must be one of these Layer 3 interfaces:

  - Routed port: A physical port configured as a Layer 3 port by entering the **no switchport** command in interface configuration mode.

  - SVI: A VLAN interface created by using the **interface vlan** *vlan_id* in global configuration mode, 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* in global configuration mode, and binding the Ethernet interface into the channel group.

- All Layer 3 interfaces must have IP addresses assigned to them.

- The version of an HSRP group can be changed from HSRPv2 to HSRPv1 only if the group number is less than 256.

- If you change the HSRP version on an interface, each HSRP group resets because it now has a new virtual MAC address.

- Only on mixed stacks of Catalyst switches:

- HSRP groups can be configured up to 32 instances.

- Configure only one instance of a First Hop Redundancy Protocol (FHRP). The switches support HSRPv1, HSRPv2, and HSRP for IPv6.

- HSRP for IPv4 and HSRP for IPv6 are mutually exclusive. You cannot enable both at the same time.

- When configuring group numbers for HSRPv2 and HSRP, 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.

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.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **standby version** { **1** | **2** }
4. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
5. **end**
6. **show standby** [*interface-id* [*group*]]
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch(config)# **configure terminal**	Enters global configuration mode.

	Command or Action	**Purpose**
Step 2	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	Enters interface configuration mode, and enter the Layer 3 interface on which you want to enable HSRP.
Step 3	**standby version** { **1**	**2** } **Example:** Switch(config-if)# **standby version 1**	(Optional) Configures the HSRP version on the interface. • 1- Selects HSRPv1. • 2- Selects 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**]] **Example:** Switch(config-if)# **standby 1 ip**	Creates (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** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode
Step 6	**show standby** [*interface-id* [*group*]] **Example:** Switch # **show standby**	Verifies the configuration of the standby groups.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

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:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **standby** [*group-number*] **priority***priority*
4. **standby** [*group-number*] **preempt** [**delay** [**minimum***seconds*] [**reload***seconds*] [**sync***seconds*]]
5. **standby** [*group-number*] **track** *type number* [*interface-priority*]
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch # **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	Enters interface configuration mode, and enter the HSRP interface on which you want to set priority.
Step 3	**standby** [*group-number*] **priority***priority* **Example:** Switch(config-if)# **standby 120 priority 50**	Sets 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. Use the **no** form of the command to restore the default values.
Step 4	**standby** [*group-number*] **preempt** [**delay** [**minimum***seconds*] [**reload***seconds*] [**sync***seconds*]] **Example:** Switch(config-if)# **standby 1 preempt delay 300**	Configures the router to **preempt**, which means that when the local router has a higher priority than the active router, it becomes the active router. • (Optional) group-number-The group number to which the command applies. • (Optional) **delay minimum**—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 seconds (1 hour); the default is 0 (no delay before taking over). • (Optional) **delay reload**—Set to cause the local router to postpone taking over the active role after a reload for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over after a reload). • (Optional) **delay sync**—Set to cause the local router to postpone taking over the active role so that IP redundancy clients can reply (either with an ok or wait reply) for the number of seconds shown. The range is 0 to 3600 seconds (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*] **Example:** Switch(config-if)# **standby track interface gigabitethernet1/1/1**	Configures 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.

Command or Action	Purpose
	• 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** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 7	**show running-config**	Verifies the configuration of the standby groups.
Step 8	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring MHSRP

To enable MHSRP and load-balancing, you configure two routers as active routers for their groups, with virtual routers as standby routers as shown in the *MHSRP Load Sharing* figure in the Multiple HSRP section. 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.

Configuring Router A

SUMMARY STEPS

1. **configure terminal**
2. **interface** *type number*
3. **no switchport**
4. **ip address** *ip-address mask*
5. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
6. **standby** [*group-number*] **priority** *priority*
7. **standby** [*group-number*] **preempt** [**delay** [**minimum** *seconds*] [**reload** *seconds*] [**sync** *seconds*]]
8. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
9. **standby** [*group-number*] **preempt** [**delay** [**minimum** *seconds*] [**reload** *seconds*] [**sync** *seconds*]]
10. **end**
11. **show running-config**
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch # **configure terminal**	Enters global configuration mode.
Step 2	**interface** *type number* **Example:** Switch (config)# **interface gigabitethernet1/0/1**	Configures an interface type and enters interface configuration mode.
Step 3	**no switchport** **Example:** Switch (config)# **no switchport**	Switches an interface that is in Layer 2 mode into Layer 3 mode for Layer 3 configuration.
Step 4	**ip address** *ip-address mask* **Example:** Switch (config-if)# **ip address 10.0.0.1 255.255.255.0**	Specifies an IP address for an interface.
Step 5	**standby** [*group-number*] **ip** [*ip-address* [**secondary**]] **Example:** Switch (config-if)# **standby 1 ip 10.0.0.3**	Creates 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 6	**standby** [*group-number*] **priority** *priority* **Example:** Switch(config-if)# **standby 1 priority 110**	Sets 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. Use the **no** form of the command to restore the default values.

	Command or Action	**Purpose**
Step 7	**standby** [*group-number*] **preempt** [**delay** [**minimum** *seconds*] [**reload** *seconds*] [**sync** *seconds*]] **Example:** Switch(config-if)# **standby 1 preempt delay 300**	Configures the router to **preempt**, which means that when the local router has a higher priority than the active router, it becomes the active router. • (Optional) group-number-The group number to which the command applies. • (Optional) **delay minimum**—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 seconds (1 hour); the default is 0 (no delay before taking over). • (Optional) **delay reload**—Set to cause the local router to postpone taking over the active role after a reload for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over after a reload). • (Optional) **delay sync**—Set to cause the local router to postpone taking over the active role so that IP redundancy clients can reply (either with an ok or wait reply) for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). Use the **no** form of the command to restore the default values.
Step 8	**standby** [*group-number*] **ip** [*ip-address* [**secondary**]] **Example:** Switch (config-if)# **standby 2 ip 10.0.0.4**	Creates 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 9	**standby** [*group-number*] **preempt** [**delay** [**minimum** *seconds*] [**reload** *seconds*] [**sync** *seconds*]] **Example:** Switch(config-if)# **standby 2 preempt delay 300**	Configures the router to **preempt**, which means that when the local router has a higher priority than the active router, it becomes the active router. • (Optional) group-number-The group number to which the command applies.

Command or Action	Purpose
	• (Optional) **delay minimum**—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 seconds (1 hour); the default is 0 (no delay before taking over).
	• (Optional) **delay reload**—Set to cause the local router to postpone taking over the active role after a reload for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over after a reload).
	• (Optional) **delay sync**—Set to cause the local router to postpone taking over the active role so that IP redundancy clients can reply (either with an ok or wait reply) for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over).
	Use the **no** form of the command to restore the default values.
Step 10	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 11	**show running-config**	Verifies the configuration of the standby groups.
Step 12	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Router B

SUMMARY STEPS

1. **configure terminal**
2. **interface** *type number*
3. **no switchport**
4. **ip address** *ip-address mask*
5. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
6. **standby** [*group-number*] **priority** *priority*
7. **standby** [*group-number*] **preempt** [**delay** [**minimum** *seconds*] [**reload** *seconds*] [**sync** *seconds*]]
8. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
9. **standby** [*group-number*] **preempt** [**delay** [**minimum** *seconds*] [**reload** *seconds*] [**sync** *seconds*]]
10. **end**
11. **show running-config**
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch # **configure terminal**	Enters global configuration mode.
Step 2	**interface** *type number* **Example:** Switch (config)# **interface gigabitethernet1/0/1**	Configures an interface type and enters interface configuration mode.
Step 3	**no switchport** **Example:** Switch (config)# **no switchport**	Switches an interface that is in Layer 2 mode into Layer 3 mode for Layer 3 configuration.
Step 4	**ip address** *ip-address mask* **Example:** Switch (config-if)# **10.0.0.2 255.255.255.0**	Specifies an IP address for an interface.
Step 5	**standby** [*group-number*] **ip** [*ip-address* [**secondary**]] **Example:** Switch (config-if)# **standby 1 ip 10.0.0.3**	Creates 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 6	**standby** [*group-number*] **priority** *priority* **Example:** Switch(config-if)# **standby 1 priority 110**	Sets 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. Use the **no** form of the command to restore the default values.

	Command or Action	**Purpose**
Step 7	**standby** [*group-number*] **preempt** [**delay** [**minimum** *seconds*] [**reload** *seconds*] [**sync** *seconds*]] **Example:** `Switch(config-if)# standby 1 preempt delay 300`	Configures the router to **preempt**, which means that when the local router has a higher priority than the active router, it becomes the active router. • (Optional) group-number-The group number to which the command applies. • (Optional) **delay minimum**—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 seconds (1 hour); the default is 0 (no delay before taking over). • (Optional) **delay reload**—Set to cause the local router to postpone taking over the active role after a reload for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over after a reload). • (Optional) **delay sync**—Set to cause the local router to postpone taking over the active role so that IP redundancy clients can reply (either with an ok or wait reply) for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). Use the **no** form of the command to restore the default values.
Step 8	**standby** [*group-number*] **ip** [*ip-address* [**secondary**]] **Example:** `Switch (config-if)# standby 2 ip 10.0.0.4`	Creates 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 9	**standby** [*group-number*] **preempt** [**delay** [**minimum** *seconds*] [**reload** *seconds*] [**sync** *seconds*]] **Example:** `Switch(config-if)# standby 2 preempt delay 300`	Configures the router to **preempt**, which means that when the local router has a higher priority than the active router, it becomes the active router. • (Optional) group-number-The group number to which the command applies.

	Command or Action	Purpose
		• (Optional) **delay minimum**—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 seconds (1 hour); the default is 0 (no delay before taking over).
		• (Optional) **delay reload**—Set to cause the local router to postpone taking over the active role after a reload for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over after a reload).
		• (Optional) **delay sync**—Set to cause the local router to postpone taking over the active role so that IP redundancy clients can reply (either with an ok or wait reply) for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over).
		Use the **no** form of the command to restore the default values.
Step 10	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 11	**show running-config**	Verifies the configuration of the standby groups.
Step 12	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

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:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*

3. **standby** [*group-number*] **authentication** *string*

4. **standby** [*group-number*] **timers** *hellotime holdtime*

5. **end**

6. **show running-config**

7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch # **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** Switch(config) # **interface gigabitethernet1/0/1**	Enters interface configuration mode, and enter the HSRP interface on which you want to set priority.
Step 3	**standby** [*group-number*] **authentication** *string* **Example:** Switch(config-if) # **standby 1 authentication word**	(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* **Example:** Switch(config-if) # **standby 1 timers 5 15**	(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* —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 seconds (1 hour); the default is 0 (no delay before taking over). • *holdtime*—Set to cause the local router to postpone taking over the active role after a reload for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over after a reload).
Step 5	**end** **Example:** Switch(config-if) # **end**	Returns to privileged EXEC mode.
Step 6	**show running-config**	Verifies the configuration of the standby groups.
Step 7	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Enabling HSRP Support for ICMP Redirect Messages

ICMP redirect messages are automatically enabled on interfaces configured with HSRP. 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. 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. For more information, see the Cisco IOS IP Configuration Guide, Release 12.4.

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.

Troubleshooting HSRP

If one of the situations as shown in the following table occurs, this message appears:

```
%FHRP group not consistent with already configured groups on the switch stack - virtual MAC
reservation failed
```

Table 16: 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.

Verifying HSRP

Verifying 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.

Example

```
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 Standby, 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 172.20.128.1 expires in 00:00:09
Active router is local
Standby router is unknown expired
Standby virtual mac address is 0000.0c07.ac64
Name is test
```

Configuration Examples for Configuring HSRP

Enabling HSRP: Example

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 configurations are optional.

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if)# no switchport
Switch(config-if)# standby 1 ip
Switch(config-if)# end
Switch # show standby
```

Configuring HSRP Priority: Example

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 gigabitethernet1/0/1
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
Switch # show standby
```

Configuring MHSRP: Example

This example shows how to enable the MHSRP configuration shown in the figure *MHSRP Load Sharing*

Router A Configuration

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.0.0.1 255.255.255.0
Switch(config-if)# standby 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 gigabitethernet1/0/1
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.0.0.2 255.255.255.0
Switch(config-if)# standby 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 1 priority 110
Switch(config-if)#  standby 2 preempt
Switch(config-if)# end
```

Configuring HSRP Authentication and Timer: Example

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 gigabitethernet1/0/1
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 gigabitethernet1/0/1
Switch(config-if)# no switchport
Switch(config-if)# standby 1 ip
Switch(config-if)# standby 1 timers 5 15
Switch(config-if)# end
```

Configuring HSRP Groups and Clustering: Example

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-if)# end
```

Information About VRRP

Configuring VRRP

Virtual Router Redundancy Protocol (VRRP) is an election protocol that enables a group of routers to form a single virtual router to provide redundancy. In a VRRP configuration, one router is elected as the virtual router primary, and the other routers act as backups in case it fails. The LAN clients can then be configured with the virtual router as their default gateway, allowing several routers on a multi-access link to use the same virtual IP address. The virtual router, representing a group of routers, forms a VRRP group.

Both HSRP and VRRP perform the same function. You can choose to configure either IETF standard VRRP or Cisco's more powerful HSRP protocol on a device or stack.

Restrictions for VRRP

- 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.

- VRRPv3 is not supported.

Additional References for Configuring HSRP

Standards and RFCs

Standard/RFC	Title
RFC 2281	Cisco Hot Standby Router Protocol

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for HSRP

Release	Modification
Cisco IOS 15.0(2)EX1	This feature was introduced.

PART **III**

IPv6

C H A P T E R 11

Configuring MLD Snooping

This module contains details of configuring MLD snooping

Information About Configuring IPv6 MLD Snooping

Note To use IPv6 MLD Snooping, the switch must be running the LAN Base image.

You can use Multicast Listener Discovery (MLD) snooping to enable efficient distribution of IP Version 6 (IPv6) multicast data to clients and routers in a switched network on the switch. Unless otherwise noted, the term switch refers to a standalone switch and to a switch stack.

Note Stacking is supported only on Catalyst 2960-X switches running the LAN base image.

Note To use IPv6, you must configure the dual IPv4 and IPv6 Switch Database Management (SDM) template on the switch.

On switches running the LAN Base feature set, the routing template is not supported.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release or the Cisco IOS documentation referenced in the procedures.

Understanding 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 the links that are directly attached to the routers 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, 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 address table is constructed in software and hardware. The switch then performs IPv6 multicast-address based bridging in hardware.

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 group 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 the Catalyst 2960, 2960-S, 2960-C, 2960-X or 2960-CX 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 is 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.

MLD Snooping in Switch Stacks

The MLD IPv6 group address databases are maintained on all switches in the stack, regardless of which switch learns of an IPv6 multicast group. Report suppression and proxy reporting are done stack-wide. During the maximum response time, only one received report for a group is forwarded to the multicast routers, regardless of which switch the report arrives on.

The election of a new active stack does not affect the learning or bridging of IPv6 multicast data; bridging of IPv6 multicast data does not stop during an active stack re-election. When a new switch is added to the stack, it synchronizes the learned IPv6 multicast information from the active stack . Until the synchronization is complete, data ingress on the newly added switch is treated as unknown multicast data.

How to Configure IPv6 MLD Snooping

Default MLD Snooping Configuration

Table 17: Default MLD Snooping Configuration

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.

Feature	Default Setting
MLD listener suppression	

MLD Snooping Configuration Guidelines

When configuring MLD snooping, consider these guidelines:

- 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 the 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 or switch stack is determined by the configured SDM template.

- The maximum number of address entries allowed for the switch or switch stack is 1000.

Enabling or Disabling MLD Snooping on the Switch (CLI)

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).

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.

Beginning in privileged EXEC mode, follow these steps to globally enable MLD snooping on the switch:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**ipv6 mld snooping** **Example:** Switch(config)# **ipv6 mld snooping**	Enables MLD snooping on the switch.

	Command or Action	**Purpose**
Step 3	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 4	**copy running-config startup-config** **Example:** Switch(config)# **copy running-config startup-config**	(Optional) Save your entries in the configuration file.
Step 5	**reload** **Example:** Switch(config)# **reload**	Reload the operating system.

Enabling or Disabling MLD Snooping on a VLAN (CLI)

Beginning in privileged EXEC mode, follow these steps to enable MLD snooping on a VLAN.

Procedure

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**ipv6 mld snooping** **Example:** Switch(config)# **ipv6 mld snooping**	Enables MLD snooping on the switch.
Step 3	**ipv6 mld snooping vlan** *vlan-id* **Example:** Switch(config)# **ipv6 mld snooping vlan 1**	Enables 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.
Step 4	**end** **Example:**	Returns to privileged EXEC mode.

Command or Action	Purpose
Switch(config)# **ipv6 mld snooping vlan 1**	

Configuring a Static Multicast Group (CLI)

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.

Beginning in privileged EXEC mode, follow these steps to add a Layer 2 port as a member of a multicast group:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode
Step 2	**ipv6 mld snooping vlan** *vlan-id* **static** *ipv6_multicast_address* **interface** *interface-id* **Example:** Switch(config)# **ipv6 mld snooping vlan 1 static FF12::3 interface gigabitethernet 0/1**	Configures 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 48).
Step 3	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 4	Use one of the following: • **show ipv6 mld snooping address** • **show ipv6 mld snooping address vlan** *vlan-id* **Example:** Switch# **show ipv6 mld snooping address** or	Verifies the static member port and the IPv6 address.

Command or Action	Purpose
Switch# **show ipv6 mld snooping vlan 1**	

Configuring a Multicast Router Port (CLI)

Note   Static connections to multicast routers are supported only on switch ports.

Beginning in privileged EXEC mode, follow these steps to add a multicast router port to a VLAN:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**ipv6 mld snooping vlan** *vlan-id* **mrouter interface** *interface-id* **Example:** Switch(config)# **ipv6 mld snooping vlan 1 mrouter interface gigabitethernet 0/2**	Specifies 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 48.
Step 3	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 4	**show ipv6 mld snooping mrouter** [**vlan** *vlan-id*] **Example:** Switch# **show ipv6 mld snooping mrouter vlan 1**	Verifies that IPv6 MLD snooping is enabled on the VLAN interface.

Enabling MLD Immediate Leave (CLI)

Beginning in privileged EXEC mode, follow these steps to enable MLDv1 Immediate Leave:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:**	Enters global configuration mode.

	Command or Action	Purpose
	Switch# **configure terminal**	
Step 2	**ipv6 mld snooping vlan** *vlan-id* **immediate-leave** **Example:** Switch(config)# **ipv6 mld snooping vlan 1 immediate-leave**	Enables MLD Immediate Leave on the VLAN interface.
Step 3	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 4	**show ipv6 mld snooping vlan** *vlan-id* **Example:** Switch# **show ipv6 mld snooping vlan 1**	Verifies that Immediate Leave is enabled on the VLAN interface.

Configuring MLD Snooping Queries (CLI)

Beginning in privileged EXEC mode, follow these steps to configure MLD snooping query characteristics for the switch or for a VLAN:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**ipv6 mld snooping robustness-variable** *value* **Example:** Switch(config)# **ipv6 mld snooping robustness-variable 3**	(Optional) Sets 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.
Step 3	**ipv6 mld snooping vlan** *vlan-id* **robustness-variable** *value* **Example:** Switch(config)# **ipv6 mld snooping vlan 1 robustness-variable 3**	(Optional) Sets 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.

	Command or Action	**Purpose**
Step 4	**ipv6 mld snooping last-listener-query-count** *count* **Example:** Switch(config)# **ipv6 mld snooping last-listener-query-count 7**	(Optional) Sets 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.
Step 5	**ipv6 mld snooping vlan** *vlan-id* **last-listener-query-count** *count* **Example:** Switch(config)# **ipv6 mld snooping vlan 1 last-listener-query-count 7**	(Optional) Sets 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.
Step 6	**ipv6 mld snooping last-listener-query-interval** *interval* **Example:** Switch(config)# **ipv6 mld snooping last-listener-query-interval 2000**	(Optional) Sets 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).
Step 7	**ipv6 mld snooping vlan** *vlan-id* **last-listener-query-interval** *interval* **Example:** Switch(config)# **ipv6 mld snooping vlan 1 last-listener-query-interval 2000**	(Optional) Sets 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.
Step 8	**ipv6 mld snooping tcn query solicit** **Example:** Switch(config)# **ipv6 mld snooping tcn query solicit**	(Optional) Enables 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.
Step 9	**ipv6 mld snooping tcn flood query count** *count* **Example:** Switch(config)# **ipv6 mld snooping tcn flood query count 5**	(Optional) When TCN is enabled, specifies the number of TCN queries to be sent. The range is from 1 to 10; the default is 2.
Step 10	**end**	Returns to privileged EXEC mode.
Step 11	**show ipv6 mld snooping querier** [**vlan** *vlan-id*] **Example:** Switch(config)# **show ipv6 mld snooping querier vlan 1**	(Optional) Verifies that the MLD snooping querier information for the switch or for the VLAN.

Disabling MLD Listener Message Suppression (CLI)

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.

Beginning in privileged EXEC mode, follow these steps to disable MLD listener message suppression:

Procedure

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enter global configuration mode.
Step 2	**no ipv6 mld snooping listener-message-suppression** **Example:** Switch(config)# **no ipv6 mld snooping** **listener-message-suppression**	Disable MLD message suppression.
Step 3	**end** **Example:** Switch(config)# **end**	Return to privileged EXEC mode.
Step 4	**show ipv6 mld snooping** **Example:** Switch# **show ipv6 mld snooping**	Verify that IPv6 MLD snooping report suppression is disabled.

Displaying MLD Snooping Information

You can display MLD snooping information for dynamically learned and statically configured router ports and VLAN interfaces. You can also display IPv6 group address multicast entries for a VLAN configured for MLD snooping.

Table 18: Commands for Displaying MLD Snooping Information

Command	**Purpose**
show ipv6 mld snooping [**vlan** *vlan-id*]	Displays 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.

Command	Purpose
show ipv6 mld snooping mrouter [**vlan** *vlan-id*]	Displays 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) Enters **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*]	Displays information about the IPv6 address and incoming port for the most-recently received MLD query messages in the VLAN.
	(Optional) Enters **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 address [**vlan** *vlan-id*] [**count** | **dynamic** | **user**]	Displays all IPv6 multicast address information or specific IPv6 multicast address information for the switch or a VLAN. - Enters **count** to show the group count on the switch or in a VLAN. - Enters **dynamic** to display MLD snooping learned group information for the switch or for a VLAN. - Entesr **user** to display MLD snooping user-configured group information for the switch or for a VLAN.
show ipv6 mld snooping address vlan *vlan-id* [*ipv6-multicast-address*]	Displays MLD snooping for the specified VLAN and IPv6 multicast address.

Configuration Examples for Configuring MLD Snooping

Configuring a Static Multicast Group: 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 gigabitethernet

        1/0/1
Switch(config)# end
```

Configuring a Multicast Router Port: 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 gigabitethernet
```

```
                0/2
Switch(config)# exit
```

Enabling MLD Immediate Leave: 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: 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
```

C H A P T E R **12**

Configuring IPv6 Unicast Routing

Information About Configuring IPv6 Host Functions

This chapter describes how to configure IPv6 host functions on the Catalyst 2960, 2960-S, and 2960-C.

Note To use IPv6 Host Functions, the switch must be running the LAN Base image.

For information about configuring IPv6 Multicast Listener Discovery (MLD) snooping, see *Configuring MLD Snooping*.

To enable dual stack environments (supporting both IPv4 and IPv6) on a Catalyst 2960 switch, you must configure the switch to use the a dual IPv4 and IPv6 switch database management (SDM) template. See the "Dual IPv4 and IPv6 Protocol Stacks" section. This template is not required on Catalyst 2960-S switches.

Note For complete syntax and usage information for the commands used in this chapter, see the Cisco IOS documentation referenced in the procedures.

Understanding 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:

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*.

• Use the Search field on Cisco.com to locate the Cisco IOS software documentation. For example, if you want information about static routes, you can enter *Implementing Static Routes for IPv6* in the search field to learn about static routes.

IPv6 Addresses

The switch supports only IPv6 unicast addresses. It does not support site-local unicast addresses, or anycast 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.

• IPv6 Address Formats

• IPv6 Address Type: Multicast

• IPv6 Address Output Display

• Simplified IPv6 Packet Header

Supported IPv6 Unicast Routing Features

These sections describe the IPv6 protocol features supported by the switch:

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.

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

- 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 *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Dual IPv4 and IPv6 Protocol Stacks

This figure shows a router forwarding both IPv4 and IPv6 traffic through the same interface, based on the IP packet and destination addresses.

Figure 7: Dual IPv4 and IPv6 Support on an Interface
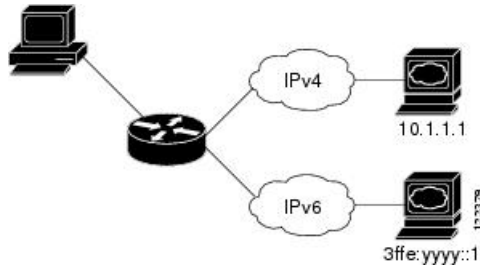
Use the dual IPv4 and IPv6 switch database management (SDM) template to enable IPv6 routing dual stack environments (supporting both IPv4 and IPv6). For more information about the dual IPv4 and IPv6 SDM template, see *Configuring SDM Templates*.

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 routes IPv4 packets and 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 .

   • If you do not plan to use IPv6, do not use the dual stack template because this template results in less hardware memory 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.

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(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 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.

For more information, see the "Managing Cisco IOS Applications over IPv6" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

EIGRP IPv6

Switches support the 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. Switches running IP Lite only support EIGRPv6 stub routing.

Before running, an instance of EIGRP IPv6 requires an implicit or explicit router ID. An implicit router ID is derived from a local IPv6 address, so any IPv6 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 IPv6 router ID.

For more information about EIGRP for IPv6, see the "Implementing EIGRP for IPv6" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

EIGRPv6 Stub Routing

The EIGRPv6 stub routing feature, reduces resource utilization by moving routed traffic closer to the end user.

In a network using EIGRPv6 stub routing, the only allowable route for IPv6 traffic to the user is through a switch that is configured with EIGRPv6 stub routing. The switch sends the routed traffic to interfaces that are configured as user interfaces or are connected to other devices.

When using EIGRPv6 stub routing, you need to configure the distribution and remote routers to use EIGRPv6 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.

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 the figure given below, switch B is configured as an EIGRPv6 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 8: EIGRP Stub Router Configuration
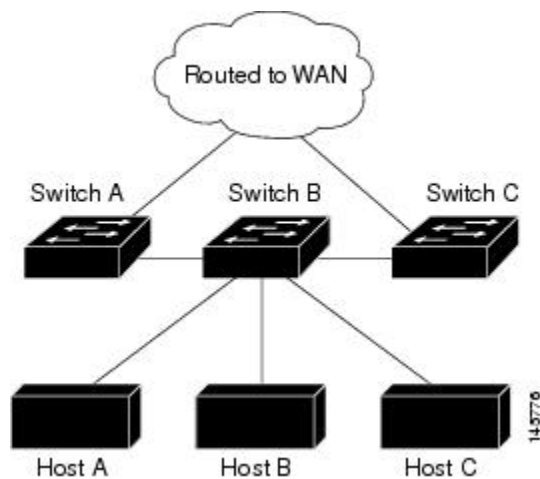
For more information about EIGRPv6 stub routing, see "Implementing EIGRP for IPv6" section of the *Cisco IOS IP Configuration Guide, Volume 2 of 3: Routing Protocols, Release 12.4*.

IPv6 and Switch Stacks

The switch supports IPv6 forwarding across the stack and IPv6 host functionality on the active stack. The active stack runs IPv6 host functionality and IPv6 applications.

While the new active stack is being elected and is resetting, the switch stack does not forward IPv6 packets. The stack MAC address changes, which also changes the IPv6 address. When you specify the stack IPv6 address with an extended unique identifier (EUI) by using the **ipv6 address**ipv6-prefix/prefix length eui-64 interface configuration command, the address is based on the interface MAC address. See the "Configuring IPv6 Addressing and Enabling IPv6 Host" section.

If you configure the persistent MAC address feature on the stack and the active stack changes, the stack MAC address does not change for approximately 4 minutes. For more information, see the "Enabling Persistent MAC Address" section in "Managing Switch Stacks."

Default IPv6 Configuration

Table 19: Default IPv6 Configuration

Feature	Default Setting
SDM template	Advance desktop. Default is advanced template
IPv6 addresses	None configured

Configuring IPv6 Addressing and Enabling IPv6 Routing

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 routing, see the "Implementing Addressing and Basic Connectivity for IPv6" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Beginning in privileged EXEC mode, follow these steps to assign an IPv6 address to a Layer 3 interface and enable IPv6 forwarding:

Procedure

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**sdm prefer dual-ipv4-and-ipv6** {**default**}	Selects an SDM template that supports IPv4 and IPv6.

	Command or Action	Purpose
	Example: Switch(config)# **sdm prefer dual-ipv4-and-ipv6 default**	• **default**—Sets the switch to the default template to balance system resources.
Step 3	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 4	**reload** **Example:** Switch# **reload**	Reloads the operating system.
Step 5	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode after the switch reloads.
Step 6	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet 1/0/1**	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 7	Use one of the following: • **ipv6 address** *ipv6-prefix/prefix length* **eui-64** • **ipv6 address** *ipv6-address/prefix length* • **ipv6 address** *ipv6-address* **link-local** • **ipv6 enable** **Example:** Switch(config-if)# **ipv6 address 2001:0DB8:c18:1::/64 eui 64** Switch(config-if)# **ipv6 address 2001:0DB8:c18:1::/64** Switch(config-if)# **ipv6 address 2001:0DB8:c18:1:: link-local** Switch(config-if)# **ipv6 enable**	• Specifies 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. • Manually configures an IPv6 address 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 enables the interface for IPv6 processing. The link-local address can only be used to communicate with nodes on the same link.

	Command or Action	Purpose
Step 8	**exit** **Example:** Switch(config-if)# **exit**	Returns to global configuration mode.
Step 9	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 10	**show ipv6 interface** *interface-id* **Example:** Switch# **show ipv6 interface gigabitethernet 1/0/1**	Verifies your entries.
Step 11	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring IPv6 ICMP Rate Limiting (CLI)

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.

Beginning in privileged EXEC mode, follow these steps to change the ICMP rate-limiting parameters:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**ipv6 icmp error-interval** *interval* [*bucketsize*] **Example:** Switch(config)# **ipv6 icmp error-interval 50 20**	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.

	Command or Action	**Purpose**
Step 3	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 4	**show ipv6 interface** [*interface-id*] **Example:** Switch# **show ipv6 interface gigabitethernet 1/0/1**	Verifies your entries.
Step 5	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Static Routing for IPv6 (CLI)

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, and enable IPv6 on at least one Layer 3 interface by configuring an IPv6 address on the interface.

For more information about configuring static IPv6 routing, see the "Implementing Static Routes for IPv6" chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Procedure

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**ipv6 route** *ipv6-prefix/prefix length* {*ipv6-address* | *interface-id* [*ipv6-address*]} [*administrative distance*] **Example:** Switch(config)# **ipv6 route 2001:0DB8::/32 gigabitethernet2/0/1 130**	Configures 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 of the address comprise the prefix (the network portion of the address). A slash mark must precede the decimal value.

	Command or Action	**Purpose**
		• *ipv6-address*—The IPv6 address of the next hop that can be used to reach the specified network. The IPv6 address of the next hop need not be directly connected; recursion is done to find the IPv6 address of the directly connected next hop. The address must be in the form documented in RFC 2373, specified in hexadecimal using 16-bit values between colons.
		• *interface-id*—Specifies direct static routes from point-to-point and broadcast interfaces. With point-to-point interfaces, there is no need to specify the IPv6 address of the next hop. With 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 also 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 any other type of route except connected routes. To configure a floating static route, use an administrative distance greater than that of the dynamic routing protocol.
Step 3	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Step 4	Use one of the following:
	
	  • **show ipv6 static** [*ipv6-address* | *ipv6-prefix/prefix length*] [**interface** *interface-id*] [**detail**]][**recursive**] [**detail**]
	  • **show ipv6 route static** [*updated*]
	
	Example:
	
	Switch# **show ipv6 static 2001:0DB8::/32 interface gigabitethernet2/0/1**
	
	or
		Verifies your entries by displaying the contents of the IPv6 routing table.
		
		  • **interface** *interface-id*—(Optional) Displays only those static routes with the specified interface as an egress interface.
		
		  • **recursive**—(Optional) Displays 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 included in the command syntax.
		
		  • **detail**—(Optional) Displays this additional information:

	Command or Action	Purpose
	Switch# **show ipv6 route static**	• For valid recursive routes, the output path set, and maximum resolution depth.
		• For invalid routes, the reason why the route is not valid.
Step 5	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Displaying IPv6

For complete syntax and usage information on these commands, see the Cisco IOS command reference publications.

Table 20: Command for Monitoring IPv6

Command	Purpose
show ipv6 access-list	Displays a summary of access lists.
show ipv6 cef	Displays Cisco Express Forwarding for IPv6.
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 a list of IPv6 routing protocols on the switch.
show ipv6 rip	Displays IPv6 RIP routing protocol status.
show ipv6 route	Displays IPv6 route table entries.
show ipv6 static	Displays IPv6 static routes.
show ipv6 traffic	Displays IPv6 traffic statistics.

Configuration Examples for IPv6 Unicast Routing

Configuring IPv6 Addressing and Enabling IPv6 Routing: 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)# ipv6 unicast-routing
Switch(config)# interface gigabitethernet1/0/11

Switch(config-if)# ipv6 address 2001:0DB8:c18:1::/64 eui 64
Switch(config-if)# end
Switch# show ipv6 interface gigabitethernet1/0/11
GigabitEthernet1/0/11 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 IPv6 ICMP Rate Limiting: 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 Static Routing for IPv6: Example

This example shows how to configure a floating static route to an interface with an administrative distance of 130:

```
Switch(config)# ipv6 route 2001:0DB8::/32 gigabitethernet2/0/1 130
```

Displaying IPv6: 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>
```

CHAPTER 13

Implementing IPv6 Multicast

Information About Implementing IPv6 Multicast Routing

This chapter describes how to implement IPv6 multicast routing on the switch.

Traditional IP communication allows a host to send packets to a single host (unicast transmission) or to all hosts (broadcast transmission). IPv6 multicast provides a third scheme, allowing a host to send a single data stream to a subset of all hosts (group transmission) simultaneously.

IPv6 Multicast Overview

An IPv6 multicast group is an arbitrary group of receivers that want to receive a particular data stream. This group has no physical or geographical boundaries--receivers can be located anywhere on the Internet or in any private network. Receivers that are interested in receiving data flowing to a particular group must join the group by signaling their local switch. This signaling is achieved with the MLD protocol.

Switches use the MLD protocol to learn whether members of a group are present on their directly attached subnets. Hosts join multicast groups by sending MLD report messages. The network then delivers data to a potentially unlimited number of receivers, using only one copy of the multicast data on each subnet. IPv6 hosts that wish to receive the traffic are known as group members.

Packets delivered to group members are identified by a single multicast group address. Multicast packets are delivered to a group using best-effort reliability, just like IPv6 unicast packets.

The multicast environment consists of senders and receivers. Any host, regardless of whether it is a member of a group, can send to a group. However, only members of a group can listen to and receive the message.

A multicast address is chosen for the receivers in a multicast group. Senders use that address as the destination address of a datagram to reach all members of the group.

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, its duration, and its membership can vary from group to group and from time to time. A group that has members may have no activity.

IPv6 Multicast Routing Implementation

The Cisco IOS software supports the following protocols to implement IPv6 multicast routing:

- MLD is used by IPv6 switches to discover multicast listeners (nodes that want to receive multicast packets destined for specific multicast addresses) on directly attached links. There are two versions of MLD: MLD version 1 is based on version 2 of the Internet Group Management Protocol (IGMP) for IPv4, and MLD version 2 is based on version 3 of the IGMP for IPv4. IPv6 multicast for Cisco IOS software uses both MLD version 2 and MLD version 1. MLD version 2 is fully backward-compatible with MLD version 1 (described in RFC 2710). Hosts that support only MLD version 1 will interoperate with a switch running MLD version 2. Mixed LANs with both MLD version 1 and MLD version 2 hosts are likewise supported.

- PIM-SM is used between switches so that they can track which multicast packets to forward to each other and to their directly connected LANs.

- PIM in Source Specific Multicast (PIM-SSM) is similar to PIM-SM with the additional ability to report interest in receiving packets from specific source addresses (or from all but the specific source addresses) to an IP multicast address.

MLD Access Group

The MLD access group provides receiver access control in Cisco IOS IPv6 multicast switches. This feature limits the list of groups a receiver can join, and it allows or denies sources used to join SSM channels.

Explicit Tracking of Receivers

The explicit tracking feature allows a switch to track the behavior of the hosts within its IPv6 network. This feature also enables the fast leave mechanism to be used with MLD version 2 host reports.

Protocol Independent Multicast

Protocol Independent Multicast (PIM) is used between switches so that they can track which multicast packets to forward to each other and to their directly connected LANs. PIM works independently of the unicast routing protocol to perform send or receive multicast route updates like other protocols. Regardless of which unicast routing protocols are being used in the LAN to populate the unicast routing table, Cisco IOS PIM uses the existing unicast table content to perform the Reverse Path Forwarding (RPF) check instead of building and maintaining its own separate routing table.

You can configure IPv6 multicast to use either PIM-SM or PIM-SSM operation, or you can use both PIM-SM and PIM-SSM together in your network.

PIM-Sparse Mode

IPv6 multicast provides support for intradomain multicast routing using PIM-SM. PIM-SM uses unicast routing to provide reverse-path information for multicast tree building, but it is not dependent on any particular unicast routing protocol.

PIM-SM is used in a multicast network when relatively few switches are involved in each multicast and these switches do not forward multicast packets for a group, unless there is an explicit request for the traffic. PIM-SM distributes information about active sources by forwarding data packets on the shared tree. PIM-SM initially uses shared trees, which requires the use of an RP.

Requests are accomplished via PIM joins, which are sent hop by hop toward the root node of the tree. The root node of a tree in PIM-SM is the RP in the case of a shared tree or the first-hop switch that is directly connected to the multicast source in the case of a shortest path tree (SPT). The RP keeps track of multicast groups and the hosts that send multicast packets are registered with the RP by that host's first-hop switch.

As a PIM join travels up the tree, switches along the path set up multicast forwarding state so that the requested multicast traffic will be forwarded back down the tree. When multicast traffic is no longer needed, a switch sends a PIM prune up the tree toward the root node to prune (or remove) the unnecessary traffic. As this PIM prune travels hop by hop up the tree, each switch updates its forwarding state appropriately. Ultimately, the forwarding state associated with a multicast group or source is removed.

A multicast data sender sends data destined for a multicast group. The designated switch (DR) of the sender takes those data packets, unicast-encapsulates them, and sends them directly to the RP. The RP receives these encapsulated data packets, de-encapsulates them, and forwards them onto the shared tree. The packets then follow the (*, G) multicast tree state in the switches on the RP tree, being replicated wherever the RP tree branches, and eventually reaching all the receivers for that multicast group. The process of encapsulating data packets to the RP is called registering, and the encapsulation packets are called PIM register packets.

IPv6 BSR: Configure RP Mapping

PIM switches in a domain must be able to map each multicast group to the correct RP address. The BSR protocol for PIM-SM provides a dynamic, adaptive mechanism to distribute group-to-RP mapping information rapidly throughout a domain. With the IPv6 BSR feature, if an RP becomes unreachable, it will be detected and the mapping tables will be modified so that the unreachable RP is no longer used, and the new tables will be rapidly distributed throughout the domain.

Every PIM-SM multicast group needs to be associated with the IP or IPv6 address of an RP. When a new multicast sender starts sending, its local DR will encapsulate these data packets in a PIM register message and send them to the RP for that multicast group. When a new multicast receiver joins, its local DR will send a PIM join message to the RP for that multicast group. When any PIM switch sends a (*, G) join message, the PIM switch needs to know which is the next switch toward the RP so that G (Group) can send a message to that switch. Also, when a PIM switch is forwarding data packets using (*, G) state, the PIM switch needs to know which is the correct incoming interface for packets destined for G, because it needs to reject any packets that arrive on other interfaces.

A small set of switches from a domain are configured as candidate bootstrap switches (C-BSRs) and a single BSR is selected for that domain. A set of switches within a domain are also configured as candidate RPs (C-RPs); typically, these switches are the same switches that are configured as C-BSRs. Candidate RPs periodically unicast candidate-RP-advertisement (C-RP-Adv) messages to the BSR of that domain, advertising their willingness to be an RP. A C-RP-Adv message includes the address of the advertising C-RP, and an optional list of group addresses and mask length fields, indicating the group prefixes for which the candidacy is advertised. The BSR then includes a set of these C-RPs, along with their corresponding group prefixes, in bootstrap messages (BSMs) it periodically originates. BSMs are distributed hop-by-hop throughout the domain.

Bidirectional BSR support allows bidirectional RPs to be advertised in C-RP messages and bidirectional ranges in the BSM. All switches in a system must be able to use the bidirectional range in the BSM; otherwise, the bidirectional RP feature will not function.

PIM-Source Specific Multicast

PIM-SSM is the routing protocol that supports the implementation of SSM and is derived from PIM-SM. However, unlike PIM-SM where data from all multicast sources are sent when there is a PIM join, the SSM feature forwards datagram traffic to receivers from only those multicast sources that the receivers have explicitly joined, thus optimizing bandwidth utilization and denying unwanted Internet broadcast traffic. Further, instead

of the use of RP and shared trees, SSM uses information found on source addresses for a multicast group. This information is provided by receivers through the source addresses relayed to the last-hop switches by MLD membership reports, resulting in shortest-path trees directly to the sources.

In SSM, delivery of datagrams is based on (S, G) channels. Traffic for one (S, G) channel consists of datagrams with an IPv6 unicast source address S and the multicast group address G as the IPv6 destination address. Systems will receive this traffic by becoming members of the (S, G) channel. Signaling is not required, but receivers must subscribe or unsubscribe to (S, G) channels to receive or not receive traffic from specific sources.

MLD version 2 is required for SSM to operate. MLD allows the host to provide source information. Before SSM can run with MLD, SSM must be supported in the Cisco IOS IPv6 switch, the host where the application is running, and the application itself.

Routable Address Hello Option

When an IPv6 interior gateway protocol is used to build the unicast routing table, the procedure to detect the upstream switch address assumes the address of a PIM neighbor is always same as the address of the next-hop switch, as long as they refer to the same switch. However, it may not be the case when a switch has multiple addresses on a link.

Two typical situations can lead to this situation for IPv6. The first situation can occur when the unicast routing table is not built by an IPv6 interior gateway protocol such as multicast BGP. The second situation occurs when the address of an RP shares a subnet prefix with downstream switches (note that the RP switch address has to be domain-wide and therefore cannot be a link-local address).

The routable address hello option allows the PIM protocol to avoid such situations by adding a PIM hello message option that includes all the addresses on the interface on which the PIM hello message is advertised. When a PIM switch finds an upstream switch for some address, the result of RPF calculation is compared with the addresses in this option, in addition to the PIM neighbor's address itself. Because this option includes all the possible addresses of a PIM switch on that link, it always includes the RPF calculation result if it refers to the PIM switch supporting this option.

Because of size restrictions on PIM messages and the requirement that a routable address hello option fits within a single PIM hello message, a limit of 16 addresses can be configured on the interface.

PIM IPv6 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 IPv6 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 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 IPv6 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.

You must also configure EIGRP stub routing when configuring PIM stub routing on the switch. For more information, see the EIGRPv6 Stub Routing, on page 145 section.

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 non-redundant access router topology is supported by the PIM stub feature. By using a non-redundant topology, the PIM passive interface assumes that it is the only interface and designated router on that access domain.

In the figure shown below, 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. See the section for more information.

Figure 9: PIM Stub Router Configuration
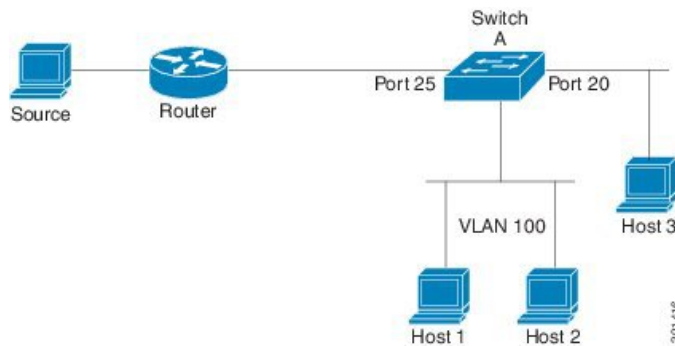
Static Mroutes

IPv6 static mroutes behave much in the same way as IPv4 static mroutes used to influence the RPF check. IPv6 static mroutes share the same database as IPv6 static routes and are implemented by extending static route support for RPF checks. Static mroutes support equal-cost multipath mroutes, and they also support unicast-only static routes.

MRIB

The Multicast Routing Information Base (MRIB) is a protocol-independent repository of multicast routing entries instantiated by multicast routing protocols (routing clients). Its main function is to provide independence between routing protocols and the Multicast Forwarding Information Base (MFIB). It also acts as a coordination and communication point among its clients.

Routing clients use the services provided by the MRIB to instantiate routing entries and retrieve changes made to routing entries by other clients. Besides routing clients, MRIB also has forwarding clients (MFIB instances) and special clients such as MLD. MFIB retrieves its forwarding entries from MRIB and notifies the MRIB of any events related to packet reception. These notifications can either be explicitly requested by routing clients or spontaneously generated by the MFIB.

Another important function of the MRIB is to allow for the coordination of multiple routing clients in establishing multicast connectivity within the same multicast session. MRIB also allows for the coordination between MLD and routing protocols.

MFIB

The MFIB is a platform-independent and routing-protocol-independent library for IPv6 software. Its main purpose is to provide a Cisco IOS platform with an interface with which to read the IPv6 multicast forwarding

table and notifications when the forwarding table changes. The information provided by the MFIB has clearly defined forwarding semantics and is designed to make it easy for the platform to translate to its specific hardware or software forwarding mechanisms.

When routing or topology changes occur in the network, the IPv6 routing table is updated, and those changes are reflected in the MFIB. The MFIB maintains next-hop address information based on the information in the IPv6 routing table. Because there is a one-to-one correlation between MFIB entries and routing table entries, the MFIB contains all known routes and eliminates the need for route cache maintenance that is associated with switching paths such as fast switching and optimum switching.

Distributed MFIB

> **Note** Distributed MFIB has its significance only in a stacked environment where the active switch distributes the MFIB information to the other stack's member switches. In the following section the line cards are nothing but the member switches in the stack.

Distributed MFIB (dMFIB) is used to switch multicast IPv6 packets on distributed platforms. dMFIB may also contain platform-specific information on replication across line cards. The basic MFIB routines that implement the core of the forwarding logic are common to all forwarding environments.

dMFIB implements the following functions:

- Distributes a copy of the MFIB to the line cards.

- Relays data-driven protocol events generated in the line cards to PIM.

- Provides an MFIB platform application program interface (API) to propagate MFIB changes to platform-specific code responsible for programming the hardware acceleration engine. This API also includes entry points to switch a packet in software (necessary if the packet is triggering a data-driven event) and to upload traffic statistics to the software.

- Provides hooks to allow clients residing on the RP to read traffic statistics on demand. (dMFIB does not periodically upload these statistics to the RP.)

The combination of dMFIB and MRIB subsystems also allows the switch to have a "customized" copy of the MFIB database in each line card and to transport MFIB-related platform-specific information from the RP to the line cards.

IPv6 Multicast Process Switching and Fast Switching

A unified MFIB is used to provide both fast switching and process switching support for PIM-SM and PIM-SSM in IPv6 multicast. In process switching, the Route Processor must examine, rewrite, and forward each packet. The packet is first received and copied into the system memory. The switch then looks up the Layer 3 network address in the routing table. The Layer 2 frame is then rewritten with the next-hop destination address and sent to the outgoing interface. The RP also computes the cyclic redundancy check (CRC). This switching method is the least scalable method for switching IPv6 packets.

IPv6 multicast fast switching allows switches to provide better packet forwarding performance than process switching. Information conventionally stored in a route cache is stored in several data structures for IPv6 multicast switching. The data structures provide optimized lookup for efficient packet forwarding.

In IPv6 multicast forwarding, the first packet is fast-switched if the PIM protocol logic allows it. In IPv6 multicast fast switching, the MAC encapsulation header is precomputed. IPv6 multicast fast switching uses the MFIB to make IPv6 destination prefix-based switching decisions. In addition to the MFIB, IPv6 multicast fast switching uses adjacency tables to prepend Layer 2 addressing information. The adjacency table maintains Layer 2 next-hop addresses for all MFIB entries.

The adjacency table is populated as adjacencies are discovered. Each time an adjacency entry is created (such as through ARP), a link-layer header for that adjacent node is precomputed and stored in the adjacency table. Once a route is determined, it points to a next hop and corresponding adjacency entry. It is subsequently used for encapsulation during switching of packets.

A route might have several paths to a destination prefix, such as when a switch is configured for simultaneous load balancing and redundancy. For each resolved path, a pointer is added for the adjacency corresponding to the next-hop interface for that path. This mechanism is used for load balancing across several paths.

Implementing IPv6 Multicast

Enabling IPv6 Multicast Routing

Beginning in privileged EXEC mode, follow these steps:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ipv6 multicast-routing** **Example:** `Switch (config)# ipv6 multicast-routing`	Enables multicast routing on all IPv6-enabled interfaces and enables multicast forwarding for PIM and MLD on all enabled interfaces of the switch.
Step 3	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Customizing and Verifying the MLD Protocol

Customizing and Verifying MLD on an Interface

Beginning in privileged EXEC mode, follow these steps:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**interface** *type number* **Example:**	Specifies an interface type and number, and places the switch in interface configuration mode.

	Command or Action	**Purpose**
	Switch(config)# **interface GigabitEthernet 1/0/1**	
Step 3	**ipv6 mld join-group** [*group-address*] [**include**	**exclude**] {*source-address*	**source-list** [*acl*]} **Example:** Switch (config-if) # **ipv6 mld join-group FF04::10**	Configures MLD reporting for a specified group and source.
Step 4	**ipv6 mld access-group** *access-list-name* **Example:** Switch (config-if) # **ipv6 access-list acc-grp-1**	Allows the user to perform IPv6 multicast receiver access control.
Step 5	**ipv6 mld static-group** [*group-address*] [**include**	**exclude**] {*source-address*	source-list [*acl*]} **Example:** Switch (config-if) # **ipv6 mld static-group ff04::10 include 100::1**	Statically forwards traffic for the multicast group onto a specified interface and cause the interface to behave as if a MLD joiner were present on the interface.
Step 6	**ipv6 mld query-max-response-time** *seconds* **Example:** Switch (config-if) # **ipv6 mld query-max-response-time 20**	Configures the maximum response time advertised in MLD queries.
Step 7	**ipv6 mld query-timeout** *seconds* **Example:** Switch (config-if) # **ipv6 mld query-timeout 130**	Configures the timeout value before the switch takes over as the querier for the interface.
Step 8	**exit** **Example:** Switch (config-if) # **exit**	Enter this command twice to exit interface configuration mode and enter privileged EXEC mode.
Step 9	**show ipv6 mld groups [link-local]** [*group-name*	*group-address*] [*interface-type interface-number*] [**detail**	**explicit**] **Example:** Switch # **show ipv6 mld groups GigabitEthernet 1/0/1**	Displays the multicast groups that are directly connected to the switch and that were learned through MLD.

	Command or Action	Purpose
Step 10	**show ipv6 mld groups summary** **Example:** Switch # **show ipv6 mld groups summary**	Displays the number of (*, G) and (S, G) membership reports present in the MLD cache.
Step 11	**show ipv6 mld interface** [*type number*] **Example:** Switch # **show ipv6 mld interface GigabitEthernet 1/0/1**	Displays multicast-related information about an interface.
Step 12	**debug ipv6 mld** [*group-name* | *group-address* | *interface-type*] **Example:** Switch # **debug ipv6 mld**	Enables debugging on MLD protocol activity.
Step 13	**debug ipv6 mld explicit** [*group-name* | *group-address* **Example:** Switch # **debug ipv6 mld explicit**	Displays information related to the explicit tracking of hosts.
Step 14	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Implementing MLD Group Limits

Per-interface and global MLD limits operate independently of each other. Both per-interface and global MLD limits can be configured on the same switch. The number of MLD limits, globally or per interface, is not configured by default; the limits must be configured by the user. A membership report that exceeds either the per-interface or the global state limit is ignored.

Configuring Explicit Tracking of Receivers to Track Host Behavior

The explicit tracking feature allows a switch to track the behavior of the hosts within its IPv6 network and enables the fast leave mechanism to be used with MLD version 2 host reports.

Beginning in privileged EXEC mode, follow these steps:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *type number* **Example:** Switch(config)# **interface GigabitEthernet 1/0/1**	Specifies an interface type and number, and places the switch in interface configuration mode.

	Command or Action	**Purpose**
Step 3	**ipv6 mld explicit-tracking** *access-list-name* **Example:** Switch(config-if)# **ipv6 mld explicit-tracking list1**	Enables explicit tracking of hosts.
Step 4	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Resetting the MLD Traffic Counters

Beginning in privileged EXEC mode, follow these steps.

Procedure

	Command or Action	**Purpose**
Step 1	**clear ipv6 mld traffic** **Example:** Switch # **clear ipv6 mld traffic**	Resets all MLD traffic counters.
Step 2	**show ipv6 mld traffic** **Example:** Switch # **show ipv6 mld traffic**	Displays the MLD traffic counters.
Step 3	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Clearing the MLD Interface Counters

Beginning in privileged EXEC mode, follow these steps.

Procedure

	Command or Action	**Purpose**
Step 1	**clear ipv6 mld counters** *interface-type* **Example:** Switch # **clear ipv6 mld counters Ethernet1/0**	Clears the MLD interface counters.
Step 2	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Configuring PIM

This section explains how to configure PIM.

Configuring PIM-SM and Displaying PIM-SM Information for a Group Range

Beginning in privileged EXEC mode, follow these steps:

Procedure

	Command or Action	**Purpose**
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	ipv6 pim **rp-address** *ipv6-address*[*group-access-list*] **Example:** Switch (config) # **ipv6 pim rp-address 2001:DB8::01:800:200E:8C6C acc-grp-1**	Configures the address of a PIM RP for a particular group range.
Step 3	**exit** **Example:** Switch (config) # **exit**	Exits global configuration mode, and returns the switch to privileged EXEC mode.
Step 4	**show ipv6 pim interface** [state-on] [**state-off**] [*type-number*] **Example:** Switch # **show ipv6 pim interface**	Displays information about interfaces configured for PIM.
Step 5	**show ipv6 pim group-map** [*group-name* | group-address] | [*group-range* | *group-mask*] [**info-source {bsr** | **default** | **embedded-rp** | **static**}] **Example:** Switch # **show ipv6 pim group-map**	Displays an IPv6 multicast group mapping table.
Step 6	**show ipv6 pim neighbor** [**detail**] [*interface-type interface-number* | **count**] **Example:** Switch # **show ipv6 pim neighbor**	Displays the PIM neighbors discovered by the Cisco IOS software.
Step 7	**show ipv6 pim range-list** [**config**] [*rp-address* | *rp-name*] **Example:** Switch # **show ipv6 pim range-list**	Displays information about IPv6 multicast range lists.
Step 8	**show ipv6 pim tunnel** [*interface-type interface-number*] **Example:** Switch # **show ipv6 pim tunnel**	Displays information about the PIM register encapsulation and de-encapsulation tunnels on an interface.

	Command or Action	Purpose
Step 9	**debug ipv6 pim** [*group-name* | *group-address* | **interface** *interface-type* | **bsr** | **group** | **mvpn** | **neighbor**] **Example:** Switch # **debug ipv6 pim**	Enables debugging on PIM protocol activity.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Configuring PIM Options

Beginning in privileged EXEC mode, follow these steps:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ipv6 pim spt-threshold infinity** [**group-list** *access-list-name*] **Example:** Switch (config) # **ipv6 pim spt-threshold infinity group-list acc-grp-1**	Configures when a PIM leaf switch joins the SPT for the specified groups.
Step 3	**ipv6 pim accept-register** {**list** *access-list* | **route-map** *map-name*} **Example:** Switch (config) # **ipv6 pim accept-register route-map reg-filter**	Accepts or rejects registers at the RP.
Step 4	**interface** *type number* **Example:** Switch (config) # **interface GigabitEthernet 1/0/1**	Specifies an interface type and number, and places the switch in interface configuration mode.
Step 5	**ipv6 pim dr-priority** *value* **Example:** Switch (config-if) # **ipv6 pim dr-priority 3**	Configures the DR priority on a PIM switch.
Step 6	**ipv6 pim hello-interval** *seconds* **Example:** Switch (config-if) # **ipv6 pim hello-interval 45**	Configures the frequency of PIM hello messages on an interface.

	Command or Action	**Purpose**
Step 7	**ipv6 pim join-prune-interval** *seconds* **Example:** `Switch (config-if) # ipv6 pim join-prune-interval 75`	Configures periodic join and prune announcement intervals for a specified interface.
Step 8	**exit** **Example:** `Switch (config-if) # exit`	Enter this command twice to exit interface configuration mode and enter privileged EXEC mode.
Step 9	**ipv6 pim join-prune statistic** [*interface-type*] **Example:** `Switch (config-if) # show ipv6 pim join-prune statistic`	Displays the average join-prune aggregation for the most recently aggregated packets for each interface.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Resetting the PIM Traffic Counters

If PIM malfunctions or in order to verify that the expected number of PIM packets are received and sent, the user can clear PIM traffic counters. Once the traffic counters are cleared, the user can enter the show ipv6 pim traffic command to verify that PIM is functioning correctly and that PIM packets are being received and sent correctly.

Beginning in privileged EXEC mode, follow these steps:

Procedure

	Command or Action	**Purpose**
Step 1	**clear ipv6 pim traffic** **Example:** `Switch # clear ipv6 pim traffic`	Resets the PIM traffic counters.
Step 2	**show ipv6 pim traffic** **Example:** `Switch # show ipv6 pim traffic`	Displays the PIM traffic counters.
Step 3	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Clearing the PIM Topology Table to Reset the MRIB Connection

No configuration is necessary to use the MRIB. However, users may in certain situations want to clear the PIM topology table in order to reset the MRIB connection and verify MRIB information.

Beginning in privileged EXEC mode, follow these steps:

Procedure

	Command or Action	Purpose
Step 1	**clear ipv6 pim topology** [*group-name* | *group-address*] **Example:** Switch # **clear ipv6 pim topology FF04::10**	Clears the PIM topology table.
Step 2	**show ipv6 mrib client** [**filter**] [**name** {*client-name* | *client-name* **:** **client-id**}] **Example:** Switch # **show ipv6 mrib client**	Displays multicast-related information about an interface.
Step 3	**show ipv6 mrib route** {**link-local** | **summary** | [*sourceaddress-or-name* | *****] [*groupname-or-address*[*prefix-length*]]] **Example:** Switch # **show ipv6 mrib route**	Displays the MRIB route information.
Step 4	**show ipv6 pim topology** [*groupname-or-address* [*sourceaddress-or-name*] | **link-local** | **route-count** [**detail**]] **Example:** Switch # **show ipv6 pim topology**	Displays PIM topology table information for a specific group or all groups.
Step 5	**debug ipv6 mrib client** **Example:** Switch # **debug ipv6 mrib client**	Enables debugging on MRIB client management activity.
Step 6	**debug ipv6 mrib io** **Example:** Switch # **debug ipv6 mrib io**	Enables debugging on MRIB I/O events.
Step 7	**debug ipv6 mrib proxy** **Example:** Switch # **debug ipv6 mrib proxy**	Enables debugging on MRIB proxy activity between the switch processor and line cards on distributed switch platforms.
Step 8	**debug ipv6 mrib route** [*group-name* | *group-address*] **Example:** Switch # **debug ipv6 mrib route**	Displays information about MRIB routing entry-related activity.

	Command or Action	Purpose
Step 9	**debug ipv6 mrib table** **Example:** `Switch # debug ipv6 mrib table`	Enables debugging on MRIB table management activity.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Configuring PIM IPv6 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 MLD traffic.

PIM IPv6 Stub Routing Configuration Guidelines

- Before configuring PIM stub routing, you must have IPv6 multicast routing configured on both the stub router and the central router. You must also have PIM mode (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. For more information, see the EIGRPv6 Stub Routing, on page 145 section.
- Only directly connected multicast (MLD) 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.

Default IPv6 PIM Routing Configuration

This table displays the default IPv6 PIM routing configuration for the Switch.

Table 21: Default Multicast Routing Configuration

Feature	Default Setting
Multicast routing	Disabled on all interfaces.
PIM version	Version 2.
PIM mode	No mode is defined.
PIM stub routing	None configured.
PIM RP address	None configured.
PIM domain border	Disabled.
PIM multicast boundary	None.
Candidate BSRs	Disabled.

Feature	Default Setting
Candidate RPs	Disabled.
Shortest-path tree threshold rate	0 kb/s.
PIM router query message interval	30 seconds.

Enabling IPV6 PIM Stub Routing

Before you begin

PIM stub routing is disabled in IPv6 by default. Beginning in privileged EXEC mode, follow these steps to enable PIM stub routing on an interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 multicast pim-passive-enable**
4. **interface** *interface-id*
5. **ipv6 pim**
6. **ipv6 pim** {**bsr**} | {**dr-priority** | *value*} | {**hello-interval** | *seconds*} | {**join-prune-interval** | *seconds*} | {**passive**}
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** Example: Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 3	**ipv6 multicast pim-passive-enable** Example: Switch(config-if)# **ipv6 multicast pim-passive-enable**	Enables IPv6 Multicast PIM routing on the switch.

	Command or Action	**Purpose**
Step 4	**interface** *interface-id* **Example:** `Switch(config)# `**`interface`** **`gigabitethernet 9/0/6`**	Specifies the interface on which you want to enable PIM stub routing, and enters 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. You will also need to enable IP PIM sparse mode on the interface, and join the interface as a statically connected member to an MLD static group. • An SVI—A VLAN interface created by using the **interface vlan** *vlan-id* global configuration command. You will also need to enable IP PIM sparse mode on the VLAN, join the VLAN as a statically connected member to an MLD static group, and then enable MLD snooping on the VLAN, the MLD static group, and physical interface. These interfaces must have IPv6 addresses assigned to them.
Step 5	**ipv6 pim** **Example:** `Switch(config-if)# `**`ipv6 pim`**	Enables the PIM on the interface.
Step 6	**ipv6 pim** {**bsr**}	{**dr-priority**	*value*}	{**hello-interval**	*seconds*}	{**join-prune-interval**	*seconds*}	{**passive**} **Example:** `Switch(config-if)# `**`ipv6 pim`** `bsr|dr-priority|hello-interval|join-prune-interval|passive`	Configures the various PIM stub features on the interface. Enter **bsr** to configure BSR on a PIM switch Enter **dr-priority** to configure the DR priority on a PIM switch. Enter **hello-interval** to configure the frequency of PIM hello messages on an interface. Enter **join-prune-interval** to configure periodic join and prune announcement intervals for a specified interface. Enter **passive** to configure the PIM in the passive mode.
Step 7	**end** **Example:** `Switch(config-if)# `**`end`**	Returns to privileged EXEC mode.

Monitoring IPv6 PIM Stub Routing

Table 22: PIM Stub Configuration show Commands

Command	Purpose
show ipv6 pim interface Switch# **show ipv6 pim interface**	Displays the PIM stub that is enabled on each interface.
show ipv6 mld groups Switch# **show ipv6 mld groups**	Displays the interested clients that have joined the specific multicast source group.
show ipv6 mroute Switch# **show ipv6 mroute**	Verifies that the multicast stream forwards from the source to the interested clients.

Configuring a BSR

The tasks included here are described below.

Configuring a BSR and Verifying BSR Information

Beginning in privileged EXEC mode, follow these steps:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ipv6 pim bsr candidate bsr** *ipv6-address*[*hash-mask-length*] [**priority** *priority-value*] **Example:** Switch (config) # **ipv6 pim bsr candidate bsr 2001:DB8:3000:3000::42 124 priority 10**	Configures a switch to be a candidate BSR.
Step 3	**interface** *type number* **Example:** Switch (config) # **interface GigabitEthernet 1/0/1**	Specifies an interface type and number, and places the switch in interface configuration mode.
Step 4	**ipv6 pim bsr border** **Example:** Switch (config-if) # **ipv6 pim bsr border**	Specifies an interface type and number, and places the switch in interface configuration mode.

	Command or Action	Purpose
Step 5	**exit** **Example:** Switch (config-if) # **exit**	Enter this command twice to exit interface configuration mode and enter privileged EXEC mode.
Step 6	**show ipv6 pim** bsr {**election** | **rp-cache** | **candidate-rp**} **Example:** Switch (config-if) # **show ipv6 pim bsr election**	Displays information related to PIM BSR protocol processing.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Sending PIM RP Advertisements to the BSR

Beginning in privileged EXEC mode, follow these steps:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ipv6 pim bsr candidate rp** *ipv6-address* [**group-list** *access-list-name*] [**priority** *priority-value*] [**interval** seconds] **Example:** Switch(config) # **ipv6 pim bsr candidate rp 2001:DB8:3000:3000::42 priority 0**	Sends PIM RP advertisements to the BSR.
Step 3	**interface** *type number* **Example:** Switch(config) # **interface GigabitEthernet 1/0/1**	Specifies an interface type and number, and places the switch in interface configuration mode.
Step 4	**ipv6 pim bsr border** **Example:** Switch(config-if) # **ipv6 pim bsr border**	Configures a border for all BSMs of any scope on a specified interface.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Configuring BSR for Use Within Scoped Zones

Beginning in privileged EXEC mode, follow these steps:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ipv6 pim bsr candidate rp** *ipv6-address* [*hash-mask-length*] [**priority** priority-value] **Example:** Switch(config) # **ipv6 pim bsr candidate bsr 2001:DB8:1:1:4**	Configures a switch to be a candidate BSR.
Step 3	**ipv6 pim bsr candidate rp** *ipv6-address* [**group-list** *access-list-name*] [**priority** priority-value] [**interval** seconds] **Example:** Switch(config) # **ipv6 pim bsr candidate rp 2001:DB8:1:1:1 group-list list scope 6**	Configures the candidate RP to send PIM RP advertisements to the BSR.
Step 4	**interface** *type number* **Example:** Switch(config-if) # **interface GigabitEthernet 1/0/1**	Specifies an interface type and number, and places the switch in interface configuration mode.
Step 5	**ipv6 multicast boundary scope** *scope-value* **Example:** Switch(config-if) # **ipv6 multicast boundary scope 6**	Configures a multicast boundary on the interface for a specified scope.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Configuring BSR Switches to Announce Scope-to-RP Mappings

IPv6 BSR switches can be statically configured to announce scope-to-RP mappings directly instead of learning them from candidate-RP messages. A user might want to configure a BSR switch to announce scope-to-RP mappings so that an RP that does not support BSR is imported into the BSR. Enabling this feature also allows an RP positioned outside the enterprise's BSR domain to be learned by the known remote RP on the local candidate BSR switch.

Beginning in privileged EXEC mode, follow these steps:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal**	Enter global configuration mode.

	Command or Action	Purpose
Step 2	**ipv6 pim bsr announced rp** *ipv6-address* [**group-list** *access-list-name*] [**priority** *priority-value*] **Example:** Switch(config)# **ipv6 pim bsr announced rp 2001:DB8:3000:3000::42 priority 0**	Announces scope-to-RP mappings directly from the BSR for the specified candidate RP.
Step 3	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Configuring SSM Mapping

When the SSM mapping feature is enabled, DNS-based SSM mapping is automatically enabled, which means that the switch will look up the source of a multicast MLD version 1 report from a DNS server.

You can use either DNS-based or static SSM mapping, depending on your switch configuration. If you choose to use static SSM mapping, you can configure multiple static SSM mappings. If multiple static SSM mappings are configured, the source addresses of all matching access lists will be used.

Note   To use DNS-based SSM mapping, the switch needs to find at least one correctly configured DNS server, to which the switch may be directly attached.

Beginning in privileged EXEC mode, follow these steps:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ipv6 mld ssm-map enable** **Example:** Switch(config) # **ipv6 mld ssm-map enable**	Enables the SSM mapping feature for groups in the configured SSM range.
Step 3	**no ipv6 mld ssm-map query dns** **Example:** Switch(config) # **no ipv6 mld ssm-map query dns**	Disables DNS-based SSM mapping.
Step 4	**ipv6 mld ssm-map static** *access-list source-address* **Example:** Switch(config-if) # **ipv6 mld ssm-map static SSM_MAP_ACL_2 2001:DB8:1::1**	Configures static SSM mappings.

	Command or Action	**Purpose**
Step 5	**exit** **Example:** Switch(config-if) # **exit**	Exits global configuration mode, and returns the switch to privileged EXEC mode.
Step 6	**show ipv6 mld ssm-map** [*source-address*] **Example:** Switch(config-if) # **show ipv6 mld ssm-map**	Displays SSM mapping information.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Configuring Static Mroutes

Static multicast routes (mroutes) in IPv6 can be implemented as an extension of IPv6 static routes. You can configure your switch to use a static route for unicast routing only, to use a static multicast route for multicast RPF selection only, or to use a static route for both unicast routing and multicast RPF selection.

Beginning in privileged EXEC mode, follow these steps:

Procedure

	Command or Action	**Purpose**
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ipv6 route** {*ipv6-prefix / prefix-length ipv6-address* | *interface-type interface-number ipv6-address*]} [*administrative-distance*] [*administrative-multicast-distance* | *unicast* | *multicast*] [**tag** *tag*] **Example:** Switch (config) # **ipv6 route 2001:DB8::/64 6::6 100**	Establishes static IPv6 routes. The example shows a static route used for both unicast routing and multicast RPF selection.
Step 3	**exit** **Example:** Switch # **exit**	Exits global configuration mode, and returns the switch to privileged EXEC mode.
Step 4	**show ipv6 mroute** [link-local | [*group-name* | *group-address* [*source-address* | *source-name*]] **[summary]** [**count**] **Example:** Switch # **show ipv6 mroute ff07::1**	Displays the contents of the IPv6 multicast routing table.

	Command or Action	Purpose
Step 5	**show ipv6 mroute** [**link-local**	*group-name*	*group-address*] **active** [*kbps*] **Example:** Switch (config-if) # **show ipv6 mroute active**	Displays the active multicast streams on the switch.
Step 6	**show ipv6 rpf** [*ipv6-prefix*] **Example:** Switch (config-if) # **show ipv6 rpf 2001::1:1:2**	Checks RPF information for a given unicast host address and prefix.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

Using MFIB in IPv6 Multicast

Multicast forwarding is automatically enabled when IPv6 multicast routing is enabled.

Verifying MFIB Operation in IPv6 Multicast

Beginning in privileged EXEC mode, follow these steps:

Procedure

	Command or Action	Purpose
Step 1	**show ipv6 mfib** [**link-local**	**verbose**	*group-address-name*	*ipv6-prefix / prefix-length*	*source-address-name*	**active**	**count**	**interface**	**status**	**summary**] **Example:** Switch # **show ipv6 mfib**	Displays the forwarding entries and interfaces in the IPv6 MFIB.
Step 2	**show ipv6 mfib** [**link-local**	*group-name*	*group-address*] **active** [*kbps*] **Example:** Switch # **show ipv6 mfib active**	Displays the rate at which active sources are sending to multicast groups.
Step 3	**show ipv6 mfib** [all	linkscope	group-name	group-address [source-name	source-address]] **count** **Example:** Switch # **show ipv6 mfib ff07::1**	Displays the contents of the IPv6 multicast routing table.
Step 4	**show ipv6 mfib interface** **Example:**	Displays information about IPv6 multicast-enabled interfaces and their forwarding status.

	Command or Action	Purpose
	Switch # **show ipv6 mfib interface**	
Step 5	**show ipv6 mfib status** **Example:** Switch # **show ipv6 mfib status**	Displays general MFIB configuration and operational status.
Step 6	**show ipv6 mfib summary** **Example:** Switch # **show ipv6 mfib summary**	Displays summary information about the number of IPv6 MFIB entries and interfaces.
Step 7	**debug ipv6 mfib** [*group-name* | *group-address*] [**adjacency** | **db** | **fs** | **init** | **interface** | **mrib** [**detail**] | **nat** | **pak** | **platform** | **ppr** | **ps** | **signal** | **table**] **Example:** Switch # **debug ipv6 mfib FF04::10 pak**	Enables debugging output on the IPv6 MFIB.

Resetting MFIB Traffic Counters

Beginning in privileged EXEC mode, follow these steps:

Procedure

	Command or Action	Purpose
Step 1	**clear ipv6 mfib counters** [*group-name* | **group-address** [*source-address* | *source-name*]] **Example:** Switch # **clear ipv6 mfib counters FF04::10**	Resets all active MFIB traffic counters.

Configuring IPv6 ACL

Information About Configuring IPv6 ACLs

You can filter IP version 6 (IPv6) traffic by creating IPv6 access control lists (ACLs) and applying them to interfaces similarly to the way that you create and apply IP version 4(IPv4) named ACLs. You can also create and apply input router ACLs to filter Layer 3 management traffic.

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 {default }** global configuration command.

Understanding IPv6 ACLs

A switch image supports two types of IPv6 ACLs:

• IPv6 router ACLs - Supported on inbound or outbound traffic on Layer 3 interfaces, which can be routed ports, switch virtual interfaces (SVIs), or Layer 3 EtherChannels. Applied to only IPv6 packets that are routed.

• IPv6 port ACLs - Supported on inbound traffic on Layer 2 interfaces only. Applied to all IPv6 packets entering the interface.

Note If you configure unsupported IPv6 ACLs, an error message appears and the configuration does not take affect.

The switch does not support VLAN ACLs (VLAN maps) for IPv6 traffic.

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 If any port ACL (IPv4, IPv6, or MAC) is applied to an interface, that port ACL is used to filter packets, and any router ACLs attached to the SVI of the port VLAN are ignored.

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 TCAM space, packets associated with the ACL label are forwarded to the CPU, and the ACLs are applied in software.

• 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.

IPv6 ACL Limitations

With IPv4, you can configure standard and extended numbered IP ACLs, named IP ACLs, and MAC ACLs. IPv6 supports only named ACLs.

The switch supports most Cisco IOS-supported IPv6 ACLs with some exceptions:

• IPv6 source and destination addresses-ACL matching is supported only on prefixes from /0 to /64 and host addresses (/128) that are in the extended universal identifier (EUI)-64 format. The switch supports only these host addresses with no loss of information:

-aggregatable global unicast addresses

-link local addresses

• 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).

• This release supports only port ACLs and router ACLs for IPv6; it does not support VLAN ACLs (VLAN maps).

• 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.

- Output router ACLs and input port ACLs for IPv6 are supported only on . Switches support only control plane (incoming) IPv6 ACLs.

- When configuring an ACL, there is no restriction on keywords entered in the ACL, regardless of whether or not they are supported on the platform. When you apply the ACL to an interface that requires hardware forwarding (physical ports or SVIs), the switch checks to determine whether or not the ACL can be supported on the interface. If not, attaching the ACL is rejected.

- If an ACL is applied 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 ACL that is currently attached to the interface.

Configuring IPv6 ACLs

To filter IPv6 traffic, you perform these steps:

Before you begin

Before configuring IPv6 ACLs, you must select one of the dual IPv4 and IPv6 SDM templates.

Procedure

	Command or Action	Purpose
Step 1	Create an IPv6 ACL, and enter IPv6 access list configuration mode.	
Step 2	Configure the IPv6 ACL to block (deny) or pass (permit) traffic.	
Step 3	Apply the IPv6 ACL to an interface. For router ACLs, you must also configure an IPv6 address on the Layer 3 interface to which the ACL is applied.	

Default IPv6 ACL Configuration

There are no IPv6 ACLs configured or applied.

Interaction with Other Features and Switches

- If an IPv6 router ACL is configured to deny a packet, the packet is not routed. A copy of the packet is sent 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 or switch stack, and you can apply both IPv4 and IPv6 ACLs to the same interface. Each ACL must have a unique name; an error message appears if you try to use a name that is already configured.

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, packets are dropped to the CPU, and the ACLs are applied in software. When the hardware is full a message is printed to the console indicating the ACL has been unloaded and the packets will be dropped on the interface.

> **Note** Only packets of the same type as the ACL that could not be added (ipv4, ipv6, MAC) will be dropped on the interface.

Creating IPv6 ACL

Beginning in privileged EXEC mode, follow these steps to create an IPv6 ACL:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ipv6 access-list** *acl_name* **Example:** `ipv6 access-list access-list-name`	Use a name to define an IPv6 access list and enter IPv6 access-list configuration mode.
Step 3	**{deny|permit} protocol** **Example:** `{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]`	Enter deny or permit to specify whether to deny or permit the packet if conditions are matched. These are the conditions: • For protocol, enter the 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. • The source-ipv6-prefix/prefix-length or destination-ipv6-prefix/ prefix-length is the 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 (see RFC 2373). • Enter any as an abbreviation for the IPv6 prefix ::/0. • For host source-ipv6-address or destination-ipv6-address, enter the source or destination IPv6 host address for which to set deny or

Command or Action	Purpose
	permit conditions, specified in hexadecimal using 16-bit values between colons.
	• (Optional) For operator, specify an operand that compares the source or destination ports of the specified protocol. Operands are 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.
	• (Optional) The port-number is a decimal number from 0 to 65535 or the name of a TCP or UDP port. You can use TCP port names only when filtering TCP. You can use UDP port names only when filtering UDP.
	• (Optional) Enter dscp value to 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) Enter fragments to check noninitial fragments. This keyword is visible only if the protocol is ipv6.
	• (Optional) Enter log to cause an logging message to be sent to the console about the packet that matches the entry. Enter log-input to include the input interface in the log entry. Logging is supported only for router ACLs.
	• (Optional) Enter routing to specify that IPv6 packets be routed.
	• (Optional) Enter sequence value to specify the sequence number for the access list statement. The acceptable range is from 1 to 4294967295
	• (Optional) Enter time-range name to specify the time range that applies to the deny or permit statement.
Step 4	**{deny|permit} tcp** **Example:** `{deny	permit} tcp` `{source-ipv6-prefix/prefix-length	any |` `hostsource-ipv6-address}` `[operator` `[port-number]]{destination-ipv6-prefix/prefix-length` `	any |hostdestination-ipv6-address}`	(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 3, with these additional optional parameters: • ack—Acknowledgment bit set.

Command or Action	Purpose
`[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]`	• 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}—Matches only packets that are not on a given port number. • psh—Push function bit set. • range {port	protocol}—Matches only packets in the port number range. • rst—Reset bit set. • syn—Synchronize bit set. • urg—Urgent pointer bit set.
Step 5 **{deny|permit} udp** **Example:** `{deny	permit} udp {source-ipv6-prefix/prefix-length	any	hostsource-ipv6-address} [operator [port-number]]{destination-ipv6-prefix/prefix-length	any	hostdestination-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 6 **{deny|permit} icmp** **Example:** `{deny	permit} icmp {source-ipv6-prefix/prefix-length	any	hostsource-ipv6-address} [operator [port-number]] {destination-ipv6-prefix/prefix-length	any	hostdestination-ipv6-address} [operator [port-number]][icmp-type [icmp-code] |icmp-message] [dscpvalue] [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 or see command reference for this release.

	Command or Action	Purpose
Step 7	**end** **Example:** Device(config)# **end**	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.
Step 8	**show ipv6 access-list** **Example:** show ipv6 access-list	Verify the access list configuration.
Step 9	**copy running-config startup-config** **Example:** copy running-config startup-config	(Optional) Save your entries in the configuration file.

Applying an IPv6 to an Interface

This section describes how to apply IPv6 ACLs to network interfaces. You can apply an IPv6 ACL to outbound or inbound traffic on layer 2 and Layer 3 interfaces. You can apply IPv6 ACLs only to inbound management traffic on Layer 3 interfaces.

Beginning in privileged EXEC mode, follow these steps to control access to an interface:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Device# **configure terminal**	Enters global configuration mode.
Step 2	**interface interface_id** **Example:** Switch# interface interface-id	Identifies a Layer 2 interface (for port ACLs) or Layer 3 Switch Virtual interface (for router ACLs) on which to apply an access list, and enters interface configuration mode.
Step 3	**no switchport** **Example:** Switch# no switchport	Changes the interface from Layer 2 mode (the default) to Layer 3 mode (only if applying a router ACL).
Step 4	**ipv6 address** *ipv6_address* **Example:** Switch# ipv6 address ipv6-address	Configures 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 has already been configured with an explicit IPv6 address.
Step 5	**ipv6 traffic-filter** *acl_name* **Example:** Switch# ipv6 traffic-filter access-list-name {in | out}	Applies the access list to incoming or outgoing traffic on the interface.

	Command or Action	**Purpose**
Step 6	**end** **Example:** Device(config)# **end**	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.
Step 7	**show running-config interface tenGigabitEthernet 1/0/3** **Example:** Switch# show running-config interface tenGigabitEthernet 1/0/3 .... Building configuration .... Current configuration : 98 bytes ! interface TenGigabitEthernet1/0/3 switchport mode trunk ipv6 traffic-filter MyFilter out end	Shows the configuration summary.
Step 8	**copy running-config startup-config** **Example:** copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Displaying 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 privileged EXEC commands.

Procedure

	Command or Action	**Purpose**
Step 1	**show access-list** **Example:** Switch# show access-lists	Displays all access lists configured on the switch
Step 2	**show ipv6 access-list** *acl_name* **Example:** Switch# show ipv6 access-list *[access-list-name]*	Displays all configured IPv6 access list or the access list specified by name.

Configuration Examples for IPv6 ACL

Example: Creating IPv6 ACL

This example configures the IPv6 access list named CISCO. The first deny entry in the list denies all packets that have a destination TCP port number greater than 5000. The second deny entry denies packets that have a source UDP port number less than 5000. The second deny also logs all matches to the console. The first permit entry in the list permits all ICMP packets. The second permit entry in the list permits all other traffic. The second permit entry is necessary because an implicit deny -all condition is at the end of each IPv6 access list.

✎

Note   Logging is supported only on Layer 3 interfaces.

```
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
```

Example: Applying IPv6 ACLs

This example shows how to apply the access list Cisco to outbound traffic on a Layer 3 interface.

```
Switch(config-if)# no switchport
Switch(config-if)# ipv6 address 2001::/64 eui-64
Switch(config-if)# ipv6 traffic-filter CISCO out
```

Example: Displaying IPv6 ACLs

This is an example of the output from the **show access-lists** privileged EXEC command. The output shows all access lists that are configured on the switch or switch stack.

```
Switch #show access-lists
Extended IP access list hello
10 permit ip any any
IPv6 access list ipv6
permit ipv6 any any sequence 10
```

This is an example of the output from the show ipv6 access-lists privileged EXEC command. The output shows only IPv6 access lists configured on the switch or switch stack.

```
Switch# show ipv6 access-list
IPv6 access list inbound
permit tcp any any eq bgp (8 matches) sequence 10
permit tcp any any eq telnet (15 matches) sequence 20
permit udp any any sequence 30

IPv6 access list outbound
deny udp any any sequence 10
deny tcp any any eq telnet sequence 20
```

PART IV

Layer 2/3

Configuring Spanning Tree Protocol

Restrictions for STP

- An attempt to configure a switch as the root switch fails if the value necessary to be the root switch is less than 1.

- If your network consists of switches that support 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.

Information About Spanning Tree Protocol

Spanning Tree Protocol

Spanning Tree Protocol (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 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   By default, the switch sends keepalive messages (to ensure the connection is up) only on interfaces that do not have small form-factor pluggable (SFP) modules. You can change the default for an interface by entering the [**no**] **keepalive** interface configuration command with no keywords.

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. In a switch stack, all switches use the same bridge ID for a given spanning-tree instance.

- 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). See the figure following the bullets.

  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 the following figure.

- 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.

- Only one outgoing port on the stack root switch is selected as the root port. The remaining switches in the stack become its designated switches (Switch 2 and Switch 3) as shown in the following figure.

- 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.

Figure 10: Spanning-Tree Port States in a Switch Stack

One stack member is elected as the stack root switch. The stack root switch contains the outgoing root port
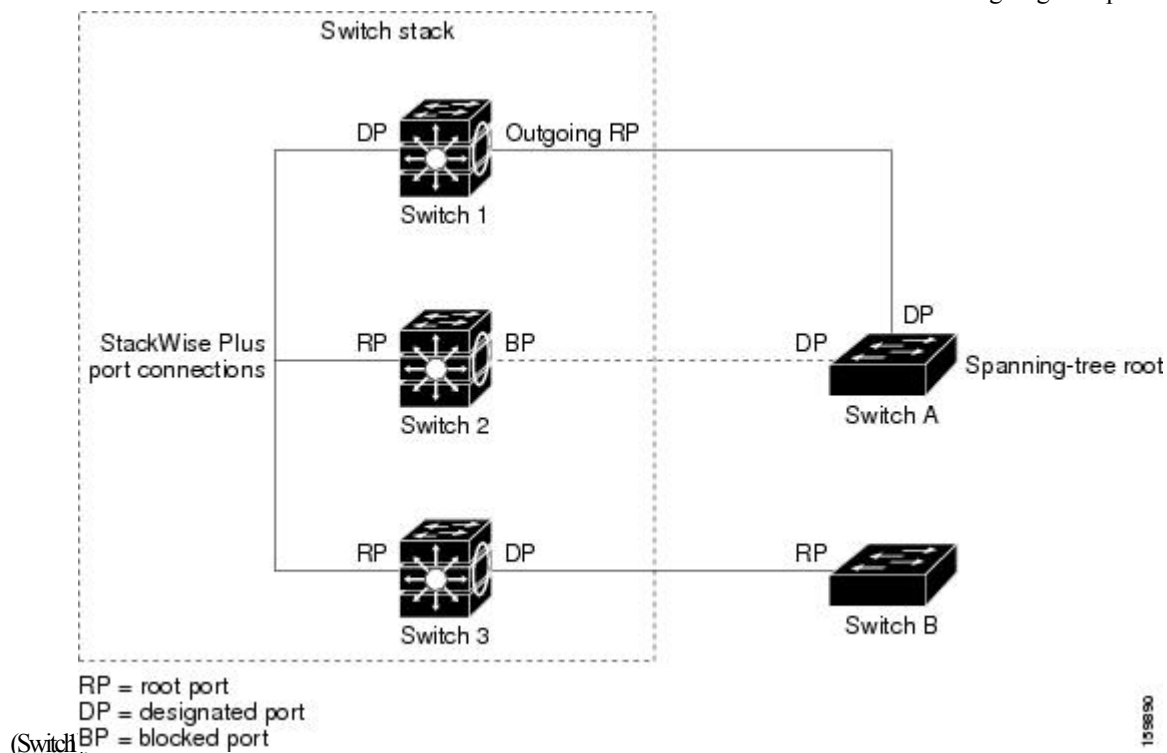
(Switch)

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, Device 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 ID 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.

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 23: Device Priority Value and Extended System ID

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. Because the switch stack appears as a single switch to the rest of the network, all switches in the stack use the same bridge ID for a given spanning tree. If the stack's active switch fails, the stack members recalculate their bridge IDs of all running spanning trees based on the new MAC address of the new stack's active switch.

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.

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 the table.

Port Priority Versus Path Cost

If a loop occurs, spanning tree uses 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.

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.

If your switch is a member of a switch stack, you must assign lower cost values to interfaces that you want selected first and higher cost values that you want selected last instead of adjusting its port priority. For details, see Related Topics.

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 11: Spanning-Tree Interface States
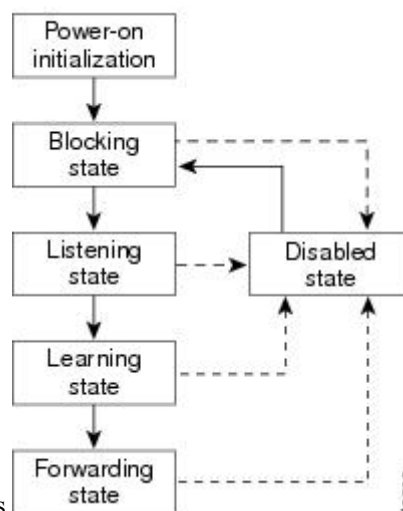
An interface moves through the states.

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 for 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.

Figure 12: Spanning-Tree Topology

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
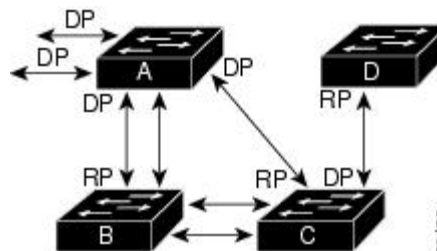
RP = Root Port
DP = Designated Port

to form a new topology with the ideal switch as the root.

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

Figure 13: 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. 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
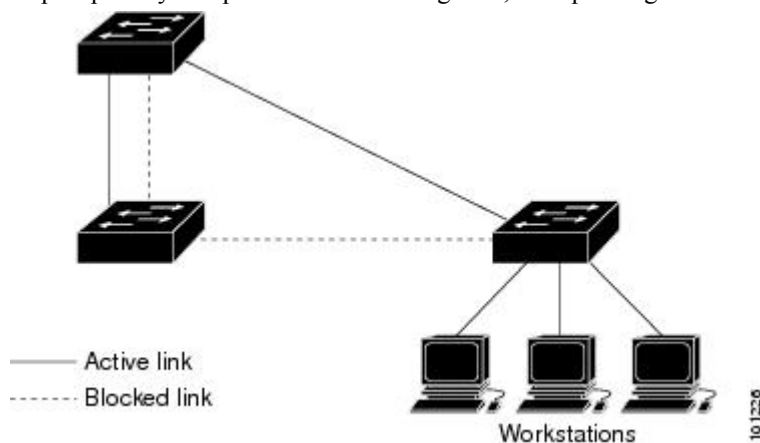
highest value.

You can also create redundant links between switches by using EtherChannel groups.

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 in the stack receives but does not forward packets destined for addresses between 0x0180C2000000 and 0x0180C200000F.

If spanning tree is enabled, the CPU on the switch or on each switch in the stack receives packets destined for 0x0180C2000000 and 0x0180C2000010. If spanning tree is disabled, the switch or each switch in the stack 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.

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 Multiple Spanning Tree Protocol (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. In a switch stack, the cross-stack rapid transition (CSRT) feature performs the same function as RSTP. You cannot run MSTP without RSTP or CSRT.

Supported Spanning-Tree Instances

In PVST+ or Rapid PVST+ mode, the switch or switch stack supports up to 128 spanning-tree instances.

In MSTP mode, the switch or switch stack supports up to 65 MST instances. The number of VLANs that can be mapped to a particular MST instance is unlimited.

Spanning-Tree Interoperability and Backward Compatibility

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.

All stack members run the same version of spanning tree (all PVST+, all Rapid PVST+, or all MSTP).

Table 24: PVST+, MSTP, and Rapid-PVST+ Interoperability and Compatibility

	PVST+	MSTP	Rapid PVST+
PVST+	Yes	Yes (with restrictions)	Yes (reverts to PVST+)

	PVST+	MSTP	Rapid PVST+
MSTP	Yes (with restrictions)	Yes	Yes (reverts to PVST+)
Rapid PVST+	Yes (reverts to PVST+)	Yes (reverts to PVST+)	Yes

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 and Inter-Switch Link (ISL) trunk 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. To use the fallback bridging feature, you must have the IP services feature set enabled on your switch.

Spanning Tree and Device Stacks

When the device stack is operating in PVST+ or Rapid PVST+ mode:

- A device stack appears as a single spanning-tree node to the rest of the network, and all stack members use the same bridge ID for a given spanning tree. The bridge ID is derived from the MAC address of the active stack.

- When a new device joins the stack, it sets its bridge ID to the active stack bridge ID. If the newly added device has the lowest ID and if the root path cost is the same among all stack members, the newly added device becomes the stack root.

- When a stack member leaves the stack, spanning-tree reconvergence occurs within the stack (and possibly outside the stack). The remaining stack member with the lowest stack port ID becomes the stack root.

- If the active stack fails or leaves the stack, the stack members elect a new active stack, and all stack members change their bridge IDs of the spanning trees to the new active stack bridge ID.

- If the device stack is the spanning-tree root and the active stack fails or leaves the stack, the stack members elect a new active stack, and a spanning-tree reconvergence occurs.

- If a neighboring device external to the device stack fails or is powered down, normal spanning-tree processing occurs. Spanning-tree reconvergence might occur as a result of losing a device in the active topology.

- If a new device external to the device stack is added to the network, normal spanning-tree processing occurs. Spanning-tree reconvergence might occur as a result of adding a device in the network.

Default Spanning-Tree Configuration

Table 25: Default Spanning-Tree Configuration

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

How to Configure Spanning-Tree Features

Changing the Spanning-Tree Mode

The switch supports three spanning-tree modes: per-VLAN spanning tree plus (PVST+), Rapid PVST+, or multiple spanning tree protocol (MSTP). By default, the switch runs the PVST+ protocol.

If you want to enable a mode that is different from the default mode, this procedure is required.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mode** {**pvst** | **mst** | **rapid-pvst**}
4. **interface** *interface-id*
5. **spanning-tree link-type point-to-point**
6. **end**
7. **clear spanning-tree detected-protocols**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**spanning-tree mode** {**pvst**	**mst**	**rapid-pvst**} **Example:** Switch(config)# **spanning-tree mode pvst**	Configures a spanning-tree mode. All stack members run the same version of spanning tree. • Select **pvst** to enable PVST+ (the default setting). • Select **mst** to enable MSTP (and RSTP). • Select **rapid-pvst** to enable rapid PVST+.
Step 4	**interface** *interface-id* **Example:** Switch(config)# **interface GigabitEthernet1/0/1**	(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. The VLAN ID range is 1 to 4094. The port-channel range is 1 to 48.

	Command or Action	**Purpose**
Step 5	**spanning-tree link-type point-to-point** **Example:** Switch(config-if)# **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.
Step 6	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 7	**clear spanning-tree detected-protocols** **Example:** Switch# **clear spanning-tree detected-protocols**	(Recommended for Rapid PVST+ mode only) If any port on the switch is connected to a port on a legacy IEEE 802.1D switch, this command restarts the protocol migration process on the entire switch. This step is optional if the designated switch detects that this switch is running rapid PVST+.

Disabling Spanning Tree

Spanning tree is enabled by default on VLAN 1 and on all newly created VLANs up to the spanning-tree limit. 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.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no spanning-tree vlan** *vlan-id*
4. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**no spanning-tree vlan** *vlan-id* **Example:** Switch(config)# **no spanning-tree vlan 300**	For *vlan-id*, the range is 1 to 4094.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Configuring the Root Switch

To configure a switch as 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.

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.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree vlan** *vlan-id* **root primary** [**diameter** *net-diameter*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**spanning-tree vlan** *vlan-id* **root primary** [**diameter** *net-diameter* **Example:** Switch(config)# **spanning-tree vlan 20-24 root primary diameter 4**	Configures a switch to become the root for the specified VLAN. • For *vlan-id*, 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 4094. • (Optional) For **diameter** *net-diameter*, specify the maximum number of switches between any two end stations. The range is 2 to 7.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

What to do next

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.

Configuring a Secondary Root Device

When you configure a switch as the secondary root, the switch priority is modified from the default value (32768) to 28672. With this priority, the switch is 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.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree vlan** *vlan-id* **root secondary** [**diameter** *net-diameter*
4. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**spanning-tree vlan** *vlan-id* **root secondary** [**diameter** *net-diameter* **Example:** Switch(config)# **spanning-tree vlan 20-24 root secondary diameter 4**	Configures a switch to become the secondary root for the specified VLAN. • For *vlan-id*, 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 4094. • (Optional) For **diameter** *net-diameter*, specify the maximum number of switches between any two end stations. The range is 2 to 7. Use the same network diameter value that you used when configuring the primary root switch.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Configuring Port Priority

✎ **Note**   If your switch is a member of a switch stack, you must use the **spanning-tree** [**vlan** *vlan-id*] **cost** *cost* interface configuration command instead of the **spanning-tree** [**vlan** *vlan-id*] **port-priority** *priority* interface configuration command to select an interface to put in the forwarding state. Assign lower cost values to interfaces that you want selected first and higher cost values that you want selected last.

This procedure is optional.

SUMMARY STEPS

1.   enable

2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree port-priority** *priority*
5. **spanning-tree vlan** *vlan-id* **port-priority** *priority*
6. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** Example: Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* Example: Switch(config)# **interface gigabitethernet1/0/2**	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*).
Step 4	**spanning-tree port-priority** *priority* Example: Switch(config-if)# **spanning-tree port-priority 0**	Configures the port priority for an interface. For *priority*, the range is 0 to 240, in increments of 16; the default is 128. Valid values are 0, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 176, 192, 208, 224, and 240. All other values are rejected. The lower the number, the higher the priority.
Step 5	**spanning-tree vlan** *vlan-id* **port-priority** *priority* Example: Switch(config-if)# **spanning-tree vlan 20-25 port-priority 0**	Configures the port priority for a VLAN. • For *vlan-id*, 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 4094. • For *priority*, the range is 0 to 240, in increments of 16; the default is 128. Valid values are 0, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 176, 192, 208, 224, and 240. All other values are rejected. The lower the number, the higher the priority.
Step 6	**end** Example:	Returns to privileged EXEC mode.

Command or Action	Purpose
Switch(config-if)# **end**	

Configuring Path Cost

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree cost** *cost*
5. **spanning-tree vlan** *vlan-id* **cost** *cost*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	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*).
Step 4	**spanning-tree cost** *cost* **Example:** Switch(config-if)# **spanning-tree cost 250**	Configures the cost for an interface. 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. For *cost*, the range is 1 to 200000000; the default value is derived from the media speed of the interface.
Step 5	**spanning-tree vlan** *vlan-id* **cost** *cost* **Example:**	Configures the cost for a VLAN.

	Command or Action	**Purpose**
	Switch(config-if)# **spanning-tree vlan 10,12-15,20 cost 300**	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.
		• For *vlan-id*, 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 4094.
		• For *cost*, the range is 1 to 200000000; the default value is derived from the media speed of the interface.
Step 6	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.

The **show spanning-tree interface** *interface-id* privileged EXEC command displays information only for ports that are in a link-up operative state. Otherwise, you can use the **show running-config** privileged EXEC command to confirm the configuration.

Configuring the Device Priority of a VLAN

You can configure the switch priority and make it more likely that a standalone switch or a switch in the stack will be chosen as the root switch.

Note Exercise care when using this command. 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.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree vlan** *vlan-id* **priority** *priority*
4. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
	`Switch> enable`	
Step 2	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 3	**spanning-tree vlan** *vlan-id* **priority** *priority* **Example:** `Switch(config)# spanning-tree vlan 20 priority 8192`	Configures the switch priority of a VLAN. • For *vlan-id*, 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 4094. • For *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. Valid priority values are 4096, 8192, 12288, 16384, 20480, 24576, 28672, 32768, 36864, 40960, 45056, 49152, 53248, 57344, and 61440. All other values are rejected.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Configuring the Hello Time

The hello time is the time interval between configuration messages generated and sent by the root switch.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **spanning-tree vlan** *vlan-id* **hello-time** *seconds*
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
	Switch> **enable**	
Step 2	**spanning-tree vlan** *vlan-id* **hello-time** *seconds* **Example:** Switch(config)# **spanning-tree vlan 20-24 hello-time 3**	Configures the hello time of a VLAN. The hello time is the time interval between configuration messages generated and sent by the root switch. These messages mean that the switch is alive. • For *vlan-id*, 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 4094. • For *seconds*, the range is 1 to 10; the default is 2.
Step 3	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.

Configuring the Forwarding-Delay Time for a VLAN

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree vlan** *vlan-id* **forward-time** *seconds*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.

	Command or Action	**Purpose**
Step 3	**spanning-tree vlan** *vlan-id* **forward-time** *seconds* **Example:** Switch(config)# **spanning-tree vlan 20,25** **forward-time 18**	Configures the forward time of a VLAN. The forwarding delay is the number of seconds an interface waits before changing from its spanning-tree learning and listening states to the forwarding state. • For *vlan-id*, 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 4094. • For *seconds*, the range is 4 to 30; the default is 15.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Configuring the Maximum-Aging Time for a VLAN

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree vlan** *vlan-id* **max-age** *seconds*
4. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**spanning-tree vlan** *vlan-id* **max-age** *seconds* **Example:**	Configures the maximum-aging time of a VLAN. The maximum-aging time is the number of seconds a switch waits without receiving spanning-tree configuration messages before attempting a reconfiguration.

	Command or Action	Purpose
	`Switch(config)# `**`spanning-tree vlan 20 max-age 30`**	• For *vlan-id*, 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 4094. • For *seconds*, the range is 6 to 40; the default is 20.
Step 4	**end** Example: `Switch(config-if)# `**`end`**	Returns to privileged EXEC mode.

Configuring the Transmit Hold-Count

You can configure the BPDU burst size by changing the transmit hold count value.

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.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree transmit hold-count** *value*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** Example: `Switch> `**`enable`**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** Example: `Switch# `**`configure terminal`**	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**spanning-tree transmit hold-count** *value* **Example:** `Switch(config)# ` **`spanning-tree transmit hold-count`** **` 6`**	Configures the number of BPDUs that can be sent before pausing for 1 second. For *value*, the range is 1 to 20; the default is 6.
Step 4	**end** **Example:** `Switch(config)# ` **`end`**	Returns to privileged EXEC mode.

Monitoring Spanning-Tree Status

Table 26: Commands for Displaying Spanning-Tree Status

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 vlan *vlan-id*	Displays spanning-tree information for the specified VLAN.
show spanning-tree interface *interface-id*	Displays spanning-tree information for the specified interface.
show spanning-tree interface *interface-id* **portfast**	Displays spanning-tree portfast information for the specified interface.
show spanning-tree summary [**totals**]	Displays a summary of interface states or displays the total lines of the STP state section.

To clear spanning-tree counters, use the **clear spanning-tree** [**interface** *interface-id*] privileged EXEC command.

Additional References for Spanning-Tree Protocol

Related Documents

Related Topic	Document Title
Layer 2 commands	*Catalyst 2960-XR Switch Layer 2 Command Reference*

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for STP

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Configuring Multiple Spanning-Tree Protocol

Prerequisites for MSTP

- For two or more switches to be in the same multiple spanning tree (MST) region, they must have the same VLAN-to-instance map, the same configuration revision number, and the same name.

- For two or more stacked 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.

- 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. You can achieve load-balancing across a switch stack by manually configuring the path cost.

- For load-balancing between a per-VLAN spanning tree plus (PVST+) and an MST cloud or between a rapid-PVST+ and an MST cloud to work, all MST boundary ports must be forwarding. MST boundary ports are forwarding when the root of the internal spanning tree (IST) of the MST cloud is the root of the common spanning tree (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.

Restrictions for MSTP

- The switch stack 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.)

- All stack members must run the same version of spanning tree (all PVST+, Rapid PVST+, or MSTP).

- VLAN Trunking Protocol (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 Simple Network Management Protocol (SNMP) support.

- 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.

- A region can have one member or multiple members with the same MST configuration; each member must be capable of processing rapid spanning tree protocol (RSTP) Bridge Protocol Data Units (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.

- After configuring a 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.

Table 27: PVST+, MSTP, and Rapid PVST+ Interoperability and 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

Information About MSTP

MSTP Configuration

MSTP, which uses RSTP for rapid convergence, enables multiple VLANs to be grouped into and 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, enables load balancing, and reduces the number of spanning-tree instances required to support a large number of VLANs. It improves the fault tolerance of the network because a failure in one instance (forwarding path) does not affect other instances (forwarding paths).

Note   The multiple spanning-tree (MST) implementation is based on the IEEE 802.1s standard.

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 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 PVST+ and rapid per-VLAN spanning-tree plus (Rapid PVST+).

A switch stack appears as a single spanning-tree node to the rest of the network, and all stack members use the same switch ID.

MSTP Configuration Guidelines

- When you enable MST by using the **spanning-tree mode mst** global configuration command, RSTP is automatically enabled.

- For configuration guidelines about UplinkFast, BackboneFast, and cross-stack UplinkFast, see the relevant sections in the Related Topics section.

- When the switch is in MST mode, it uses the long path-cost calculation method (32 bits) to compute the path cost values. With the long path-cost calculation method, the following path cost values are supported:

Speed	Path Cost Value
10 Mb/s	2,000,000
100 Mb/s	200,000
1 Gb/s	20,000
10 Gb/s	2,000
100 Gb/s	200

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.

When you configure a switch as the root, you 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 switches 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. For more information, select "Bridge ID, Switch Priority, and Extended System ID" link in Related Topics.

If your network consists of switches that support 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.

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.

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 specifying the MST region configuration on it. You can map VLANs to an MST instance, specify the region name, and set the revision number. For instructions and an example, select the "Specifying the MST Region Configuration and Enabling MSTP" link in Related Topics.

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 4094. 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 4094.

  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.

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). 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.

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 28: Prestandard and Standard 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

Illustration of MST Regions

This figure displays 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 14: MST Regions, CIST Regional Root, and CST Root
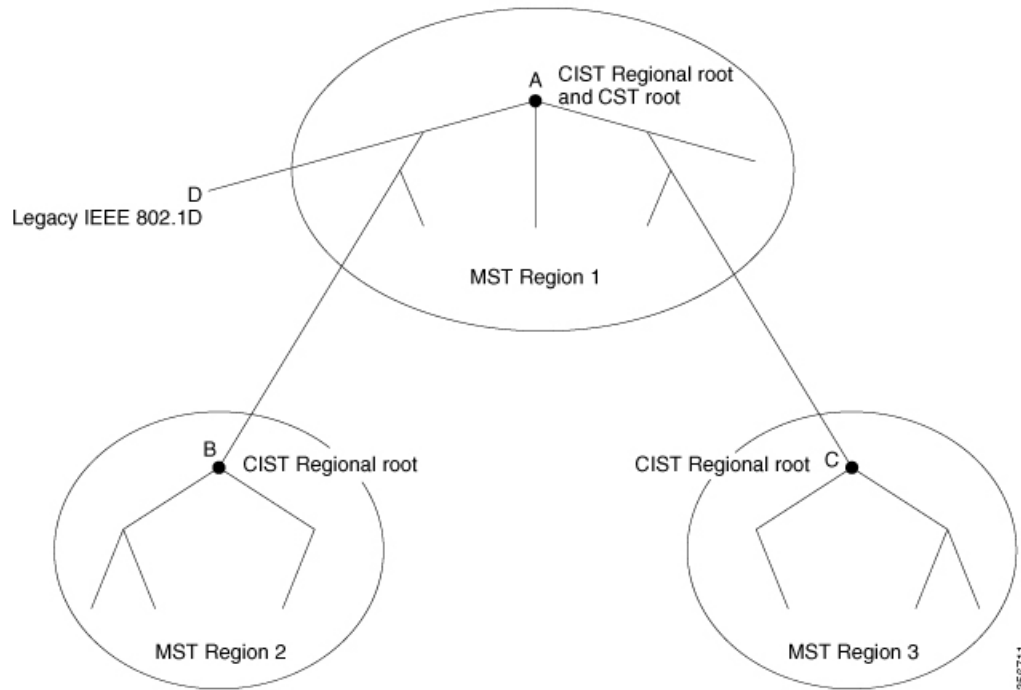
Hop Count

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)

- external (coming from another region)

When a message is internal, the CIST part is received by the CIST, and each MST instance receives its respective M-record.

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.

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 a port receiving both internal and external messages.

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 Cisco 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 boundary roles currently exist:

- 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 thus forwarding). The MSTI ports now have a special *primary* 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 15: Standard and Prestandard Switch Interoperation

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.
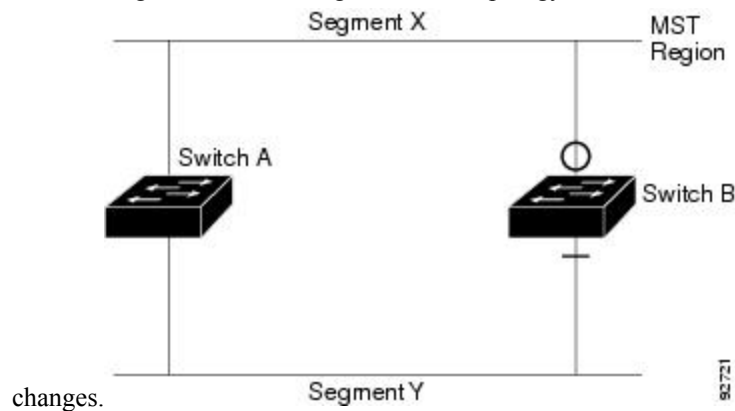
| **Note** | We recommend that you minimize the interaction between standard and prestandard MST implementations. |

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 the discarding state because disrupting connectivity in case of inconsistency is preferable to opening a bridging loop.

Figure 16: Detecting Unidirectional Link Failure

This figure 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
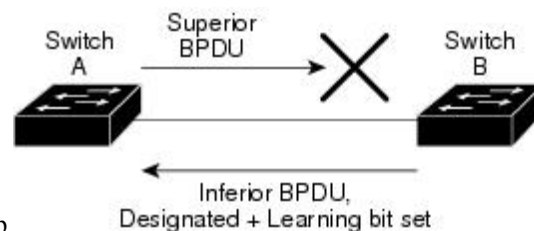
blocking) its port, which prevents the bridging loop.

MSTP and Device Stacks

A device stack appears as a single spanning-tree node to the rest of the network, and all stack members use the same bridge ID for a given spanning tree. The bridge ID is derived from the MAC address of the active stack.

If a device that does not support MSTP is added to a device stack that does support MSTP or the reverse, the device is put into a version mismatch state. If possible, the device is automatically upgraded or downgraded to the same version of software that is running on the device stack.

When a new device joins the stack, it sets its device ID to the device ID. If the newly added device has the lowest ID and if the root path cost is the same among all stack members, the newly added device becomes the stack root. A topology change occurs if the newly added device contains a better root port for the device stack or a better designated port for the LAN connected to the stack. The newly added device causes a topology change in the network if another device connected to the newly added device changes its root port or designated ports.

When a stack member leaves the stack, spanning-tree reconvergence occurs within the stack (and possibly outside the stack). The remaining stack member with the lowest stack port ID becomes the stack root.

If the active stack fails or leaves the stack, the stack members elect a new active stack, and all stack members change the device IDs of the spanning trees to the new active device ID.

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 Overview

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. The RSTP then 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 29: Port State Comparison

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.

Figure 17: Proposal and Agreement Handshaking for Rapid Convergence

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.

In a switch stack, the cross-stack rapid transition (CSRT) feature ensures that a stack member receives acknowledgments from all stack members during the proposal-agreement handshaking before moving the port to the forwarding state. CSRT is automatically enabled when the switch is in MST mode.

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.

DP = designated port
RP = root port
F = forwarding
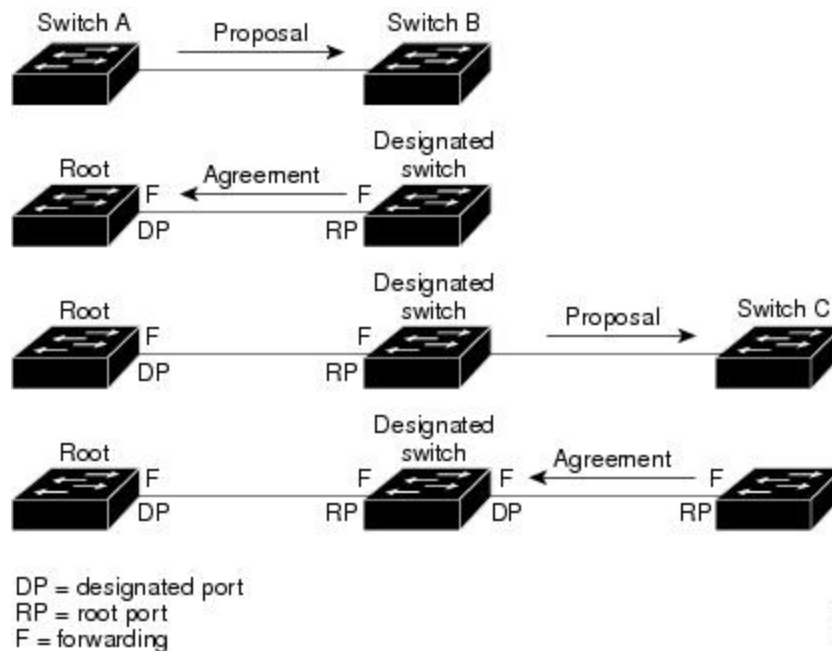
Synchronization of Port Roles

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.

Figure 18: Sequence of Events During Rapid Convergence

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.
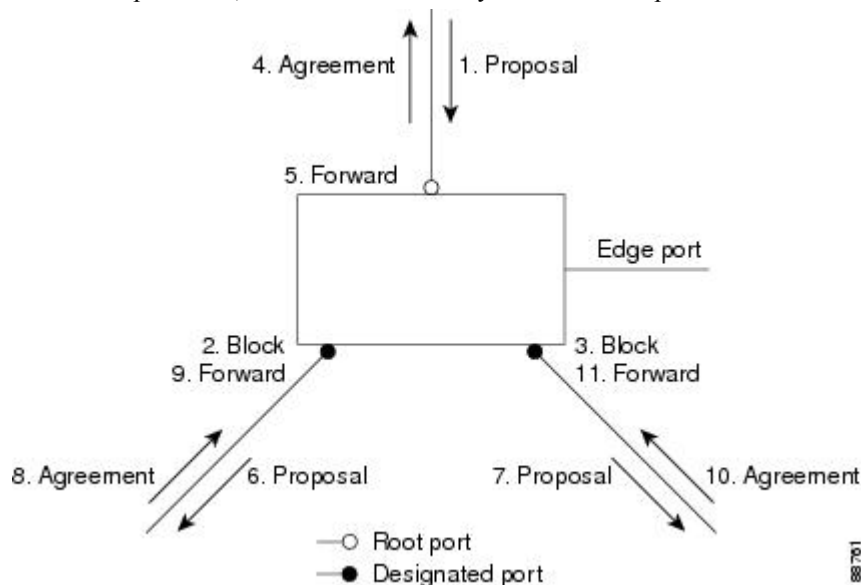
Bridge Protocol Data Unit Format and Processing

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 30: RSTP BPDU Flags

Bit	Function
0	Topology change (TC)
1	Proposal
2–3:	Port role:
00	Unknown
01	Alternate port
10	Root port
11	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 (such as a higher switch ID or a higher path cost 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.

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.

Default MSTP Configuration

Table 31: Default MSTP Configuration

Feature	Default Setting
Spanning-tree mode	MSTP
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 Mb/s: 20000 100 Mb/s: 20000 10 Mb/s: 20000 1000 Mb/s: 20000 100 Mb/s: 20000 10 Mb/s: 20000
Hello time	3 seconds

Feature	Default Setting
Forward-delay time	20 seconds
Maximum-aging time	20 seconds
Maximum hop count	20 hops

About MST-to-PVST+ Interoperability (PVST+ Simulation)

The PVST+ simulation feature enables seamless interoperability between MST and Rapid PVST+. You can enable or disable this per port, or globally. PVST+ simulation is enabled by default.

However, you may want to control the connection between MST and Rapid PVST+ to protect against accidentally connecting an MST-enabled port to a Rapid PVST+-enabled port. Because Rapid PVST+ is the default STP mode, you may encounter many Rapid PVST+-enabled connections.

Disabling this feature causes the switch to stop the MST region from interacting with PVST+ regions. The MST-enabled port moves to a PVST peer inconsistent (blocking) state once it detects it is connected to a Rapid PVST+-enabled port. This port remains in the inconsistent state until the port stops receiving Shared Spanning Tree Protocol (SSTP) BPDUs, and then the port resumes the normal STP transition process.

You can for instance, disable PVST+ simulation, to prevent an incorrectly configured switch from connecting to a network where the STP mode is not MSTP (the default mode is PVST+).

Observe these guidelines when you configure MST switches (in the same region) to interact with PVST+ switches:

- Configure the root for all VLANs inside the MST region as shown in this example:

```
Switch# show spanning-tree mst interface gigabitethernet 1/1
GigabitEthernet1/1 of MST00 is root forwarding
Edge port: no              (trunk) port guard : none    (default)
Link type: point-to-point (auto)  bpdu filter: disable (default)
Boundary : boundary       (PVST)  bpdu guard : disable (default)
Bpdus sent 10, received 310

Instance Role Sts Cost  Prio.Nbr    Vlans mapped
-------- ---- --- --------- -------- ------------------------------
0        Root FWD 20000 128.1       1-2,4-2999,4000-4094
3        Boun FWD 20000 128.1       3,3000-3999
```

The ports that belong to the MST switch at the boundary simulate PVST+ and send PVST+ BPDUs for all the VLANs.

If you enable loop guard on the PVST+ switches, the ports might change to a loop-inconsistent state when the MST switches change their configuration. To correct the loop-inconsistent state, you must disable and re-enable loop guard on that PVST+ switch.

- Do not locate the root for some or all of the VLANs inside the PVST+ side of the MST switch because when the MST switch at the boundary receives PVST+ BPDUs for all or some of the VLANs on its designated ports, root guard sets the port to the blocking state.

- When you connect a PVST+ switch to two different MST regions, the topology change from the PVST+ switch does not pass beyond the first MST region. In such a case, the topology changes are propagated only in the instance to which the VLAN is mapped. The topology change stays local to the first MST region, and the Cisco Access Manager (CAM) entries in the other region are not flushed. To make the

topology change visible throughout other MST regions, you can map that VLAN to IST or connect the PVST+ switch to the two regions through access links.

- When you disable the PVST+ simulation, note that the PVST+ peer inconsistency can also occur while the port is already in other states of inconsistency. For example, the root bridge for all STP instances must all be in either the MST region or the Rapid PVST+ side. If the root bridge for all STP instances are not on one side or the other, the software moves the port into a PVST + simulation-inconsistent state.

> **Note** We recommend that you put the root bridge for all STP instances in the MST region.

About Detecting Unidirectional Link Failure

The dispute mechanism that detects unidirectional link failures is included in the IEEE 802.1D-2004 RSTP and IEEE 802.1Q-2005 MSTP standard, and requires no user configuration.

The switch checks the consistency of the port role and state in the BPDUs it receives, to detect unidirectional link failures that could cause bridging loops. When a designated port detects a conflict, it keeps its role, but reverts to a discarding (blocking) state because disrupting connectivity in case of inconsistency is preferable to opening a bridging loop.

For example, in the figure below, Switch A is the root bridge and Switch B is the designated port. BPDUs from Switch A are lost on the link leading to switch B.

Figure 19: Detecting Unidirectional Link Failure
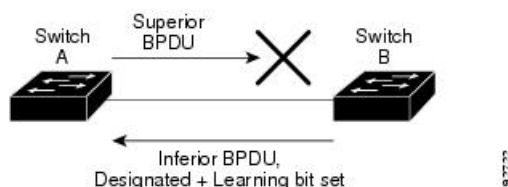
Since Rapid PVST+ (802.1w) and MST BPDUs include the role and state of the sending port, Switch A detects (from the inferior BPDU), that switch B does not react to the superior BPDUs it sends, because switch B has the role of a designated port and not the root bridge. As a result, switch A blocks (or keeps blocking) its port, thus preventing the bridging loop.

Note these guidelines and limitations relating to the dispute mechanism:

- It works only on switches running RSTP or MST (the dispute mechanism requires reading the role and state of the port initiating BPDUs).

- It may result in loss of connectivity. For example, in the figure below, Bridge A cannot transmit on the port it elected as a root port. As a result of this situation, there is loss of connectivity (r1 and r2 are designated, a1 is root and a2 is alternate. There is only a one way connectivity between A and R).

Figure 20: Loss of Connectivity
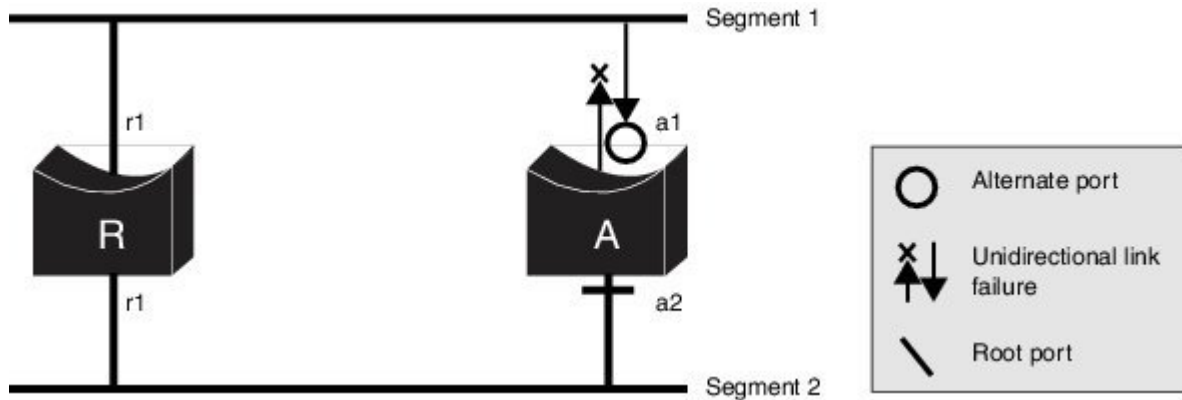
• It may cause permanent bridging loops on shared segments. For example, in the figure below, suppose that bridge R has the best priority, and that port b1 cannot receive any traffic from the shared segment 1 and sends inferior designated information on segment 1. Both r1 and a1 can detect this inconsistency. However, with the current dispute mechanism, only r1 will revert to discarding while the root port a1 opens a permanent loop. However, this problem does not occur in Layer 2 switched networks that are connected by point-to-point links.

Figure 21: Bridging Loops on Shared Segments
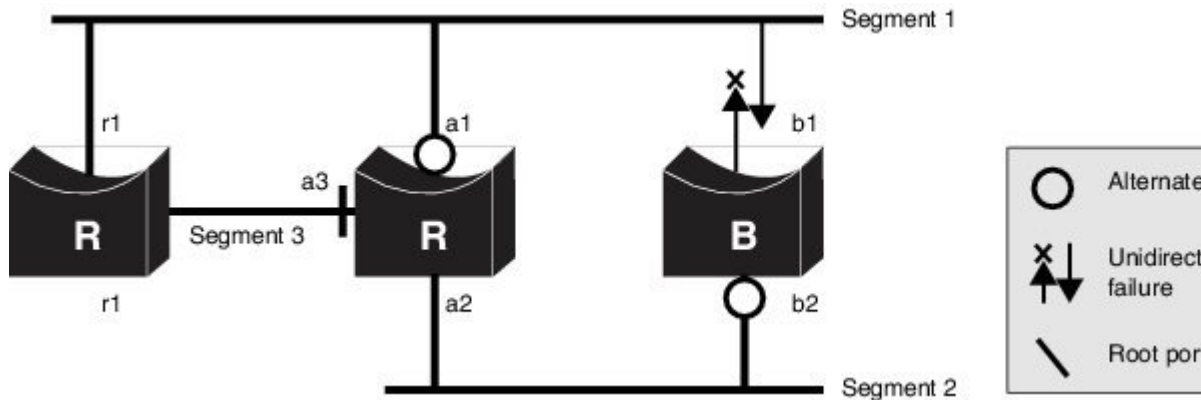
How to Configure MSTP Features

Specifying the 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.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst configuration**
4. **instance** *instance-id* **vlan** *vlan-range*
5. **name** *name*
6. **revision** *version*
7. **show pending**
8. **exit**
9. **spanning-tree mode mst**
10. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**spanning-tree mst configuration** **Example:** Switch(config)# **spanning-tree mst configuration**	Enters MST configuration mode.
Step 4	**instance** *instance-id* **vlan** *vlan-range* **Example:** Switch(config-mst)# **instance 1 vlan 10-20**	Maps VLANs to an MST instance. • For *instance-id*, the range is 0 to 4094. • For **vlan** *vlan-range*, the range is 1 to 4094. 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.

	Command or Action	**Purpose**
Step 5	**name** *name* Example: Switch(config-mst)# **name region1**	Specifies the configuration name. The *name* string has a maximum length of 32 characters and is case sensitive.
Step 6	**revision** *version* Example: Switch(config-mst)# **revision 1**	Specifies the configuration revision number. The range is 0 to 65535.
Step 7	**show pending** Example: Switch(config-mst)# **show pending**	Verifies your configuration by displaying the pending configuration.
Step 8	**exit** Example: Switch(config-mst)# **exit**	Applies all changes, and returns to global configuration mode.
Step 9	**spanning-tree mode mst** Example: Switch(config)# **spanning-tree mode mst**	Enables MSTP. RSTP is also enabled. 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.
Step 10	**end** Example: Switch(config)# **end**	Returns to privileged EXEC mode.

Configuring the Root Switch

This procedure is optional.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the switch. For instructions, see Related Topics.

You must also know the specified MST instance ID. Step 2 in the example uses 0 as the instance ID because that was the instance ID set up by the instructions listed under Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst** *instance-id* **root primary**
4. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**spanning-tree mst** *instance-id* **root primary** **Example:** Switch(config)# **spanning-tree mst 0 root primary**	Configures a switch as the root switch. • For *instance-id*, you can specify 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 4094.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Configuring a 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.

This procedure is optional.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the switch. For instructions, see Related Topics.

You must also know the specified MST instance ID. This example uses 0 as the instance ID because that was the instance ID set up by the instructions listed under Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst** *instance-id* **root secondary**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**spanning-tree mst** *instance-id* **root secondary** **Example:** Switch(config)# **spanning-tree mst 0 root secondary**	Configures a switch as the secondary root switch. • For *instance-id*, you can specify 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 4094.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Configuring 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.

> ✏️ **Note** If the switch is a member of a switch stack, you must use the **spanning-tree mst** [*instance-id*] **cost** *cost* interface configuration command instead of the **spanning-tree mst** [*instance-id*] **port-priority** *priority* interface configuration command to select a port to put in the forwarding state. Assign lower cost values to ports that you want selected first and higher cost values to ports that you want selected last. For more information, see the path costs topic listed under Related Topics.

This procedure is optional.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the switch. For instructions, see Related Topics.

You must also know the specified MST instance ID and the interface used. This example uses 0 as the instance ID and GigabitEthernet1/0/1 as the interface because that was the instance ID and interface set up by the instructions listed under Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree mst** *instance-id* **port-priority** *priority*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface GigabitEthernet1/0/1**	Specifies an interface to configure, and enters interface configuration mode.
Step 4	**spanning-tree mst** *instance-id* **port-priority** *priority* **Example:**	Configures port priority. • For *instance-id*, you can specify a single instance, a range of instances separated by a hyphen, or a series

	Command or Action	Purpose
	`Switch(config-if)# **spanning-tree mst 0 port-priority 64**`	of instances separated by a comma. The range is 0 to 4094.
		• For *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.
Step 5	**end** **Example:** `Switch(config-if)# **end**`	Returns to privileged EXEC mode.

The **show spanning-tree mst interface** *interface-id* privileged EXEC command displays information only if the port is in a link-up operative state. Otherwise, you can use the **show running-config interface** privileged EXEC command to confirm the configuration.

Configuring 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.

This procedure is optional.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the switch. For instructions, see Related Topics.

You must also know the specified MST instance ID and the interface used. This example uses 0 as the instance ID and GigabitEthernet1/0/1 as the interface because that was the instance ID and interface set up by the instructions listed under Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree mst** *instance-id* **cost** *cost*
5. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	Specifies an interface to configure, and enters interface configuration mode. Valid interfaces include physical ports and port-channel logical interfaces. The port-channel range is 1 to 48.
Step 4	**spanning-tree mst** *instance-id* **cost** *cost* **Example:** Switch(config-if)# **spanning-tree mst 0 cost 17031970**	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. • For *instance-id*, you can specify 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 4094. • For *cost*, the range is 1 to 200000000; the default value is derived from the media speed of the interface.
Step 5	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.

The **show spanning-tree mst interface** *interface-id* privileged EXEC command displays information only for ports that are in a link-up operative state. Otherwise, you can use the **show running-config** privileged EXEC command to confirm the configuration.

Configuring the Switch Priority

Changing the priority of a switch makes it more likely to be chosen as the root switch whether it is a standalone switch or a switch in the stack.

Note Exercise care when using this command. For normal network configurations, 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 specify a switch as the root or secondary root switch. You should modify the switch priority only in circumstances where these commands do not work.

This procedure is optional.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the switch. For instructions, see Related Topics.

You must also know the specified MST instance ID used. This example uses 0 as the instance ID because that was the instance ID set up by the instructions listed under Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst** *instance-id* **priority** *priority*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**spanning-tree mst** *instance-id* **priority** *priority* **Example:** Switch(config)# **spanning-tree mst 0 priority 40960**	Configures the switch priority. • For *instance-id*, you can specify 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 4094. • For *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.

	Command or Action	**Purpose**
		Priority values are 0, 4096, 8192, 12288, 16384, 20480, 24576, 28672, 32768, 36864, 40960, 45056, 49152, 53248, 57344, and 61440. These are the only acceptable values.
Step 4	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.

Configuring the Hello Time

The hello time is the time interval between configuration messages generated and sent by the root switch.

This procedure is optional.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the switch. For instructions, see Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst hello-time** *seconds*
4. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**spanning-tree mst hello-time** *seconds* **Example:** Switch(config)# **spanning-tree mst hello-time 4**	Configures the hello time for all MST instances. The hello time is the time interval between configuration messages generated and sent by the root switch. These messages indicate that the switch is alive.

	Command or Action	Purpose
		For *seconds*, the range is 1 to 10; the default is 3.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Configuring the Forwarding-Delay Time

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the switch. For instructions, see Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst forward-time** *seconds*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**spanning-tree mst forward-time** *seconds* **Example:** Switch(config)# **spanning-tree mst forward-time 25**	Configures the forward time for all MST instances. The forwarding delay is the number of seconds a port waits before changing from its spanning-tree learning and listening states to the forwarding state. For *seconds*, the range is 4 to 30; the default is 20.
Step 4	**end** **Example:**	Returns to privileged EXEC mode.

Command or Action	Purpose
Switch(config)# **end**	

Configuring the Maximum-Aging Time

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the switch. For instructions, see Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst max-age** *seconds*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**spanning-tree mst max-age** *seconds* **Example:** Switch(config)# **spanning-tree mst max-age 40**	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. For *seconds*, the range is 6 to 40; the default is 20.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Configuring the Maximum-Hop Count

This procedure is optional.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the switch. For instructions, see Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst max-hops** *hop-count*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**spanning-tree mst max-hops** *hop-count* **Example:** Switch(config)# **spanning-tree mst max-hops 25**	Specifies the number of hops in a region before the BPDU is discarded, and the information held for a port is aged. For *hop-count*, the range is 1 to 255; the default is 20.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Specifying the 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.

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.

This procedure is optional.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the switch. For instructions, see Related Topics.

You must also know the specified MST instance ID and the interface used. This example uses 0 as the instance ID and GigabitEthernet1/0/1 as the interface because that was the instance ID and interface set up by the instructions listed under Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree link-type point-to-point**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface GigabitEthernet1/0/1**	Specifies an interface to configure, and enters interface configuration mode. Valid interfaces include physical ports, VLANs, and port-channel logical interfaces. The VLAN ID range is 1 to 4094. The port-channel range is 1 to 48.
Step 4	**spanning-tree link-type point-to-point** **Example:** Switch(config-if)# **spanning-tree link-type point-to-point**	Specifies that the link type of a port is point-to-point.

	Command or Action	Purpose
Step 5	end **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.

Designating the 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.

This procedure is optional.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the switch. For instructions, see Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree mst pre-standard**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	interface *interface-id* **Example:** Switch(config)# **interface GigabitEthernet1/0/1**	Specifies an interface to configure, and enters interface configuration mode. Valid interfaces include physical ports.

	Command or Action	Purpose
Step 4	**spanning-tree mst pre-standard** **Example:** `Switch(config-if)# ` **`spanning-tree mst pre-standard`**	Specifies that the port can send only prestandard BPDUs.
Step 5	**end** **Example:** `Switch(config-if)# ` **`end`**	Returns to privileged EXEC mode.

Restarting the Protocol Migration Process

This procedure restarts the protocol migration process and forces renegotiation with neighboring switches. It reverts the switch to MST mode. It is needed when the switch no longer receives IEEE 802.1D BPDUs after it has been receiving them.

Follow these steps to restart the protocol migration process (force the renegotiation with neighboring switches) on the switch.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the switch. For instructions, see Related Topics.

If you want to use the interface version of the command, you must also know the MST interface used. This example uses GigabitEthernet1/0/1 as the interface because that was the interface set up by the instructions listed under Related Topics.

SUMMARY STEPS

1. **enable**
2. Enter one of the following commands:
   - **clear spanning-tree detected-protocols**
   - **clear spanning-tree detected-protocols interface** *interface-id*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Switch> ` **`enable`**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	Enter one of the following commands: • **clear spanning-tree detected-protocols**	The switch reverts to the MSTP mode, and the protocol migration process restarts.

Command or Action	Purpose
• **clear spanning-tree detected-protocols interface** *interface-id* **Example:** Switch# **clear spanning-tree detected-protocols** or Switch# **clear spanning-tree detected-protocols interface GigabitEthernet1/0/1**	

What to do next

This procedure may need to be repeated if the switch receives more legacy IEEE 802.1D configuration BPDUs (BPDUs with the protocol version set to 0).

Configuring PVST+ Simulation

PVST+ simulation is enabled by default. This means that all ports automatically interoperate with a connected device that is running in Rapid PVST+ mode. If you disabled the feature and want to re-configure it, refer to the following tasks.

To enable PVST+ simulation globally, perform this task:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst simulate pvst global**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**spanning-tree mst simulate pvst global** **Example:**	Enables PVST+ simulation globally. To prevent the switch from automatically interoperating with a connecting switch that is running Rapid PVST+, enter the **no** version of the command.

	Command or Action	Purpose
	Switch(config)# **spanning-tree mst simulate pvst global**	
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Enabling PVST+ Simulation on a Port

To enable PVST+ simulation on a port, perform this task:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree mst simulate pvst**
5. **end**
6. **show spanning-tree summary**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gi1/0/1**	Selects a port to configure.
Step 4	**spanning-tree mst simulate pvst** **Example:**	Enables PVST+ simulation on the specified interface. To prevent a specified interface from automatically interoperating with a connecting switch that is not running

	Command or Action	Purpose
	Switch(config-if)# **spanning-tree mst simulate pvst**	MST, enter the **spanning-tree mst simulate pvst disable** command.
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**show spanning-tree summary** **Example:** Switch# **show spanning-tree summary**	Verifies the configuration.

Examples

Examples: PVST+ Simulation

This example shows how to prevent the switch from automatically interoperating with a connecting switch that is running Rapid PVST+:

```
Switch# configure terminal
Switch(config)# no spanning-tree mst simulate pvst global
```

This example shows how to prevent a port from automatically interoperating with a connecting device that is running Rapid PVST+:

```
Switch(config)# interface1/0/1
Switch(config-if)# spanning-tree mst simulate pvst disable
```

The following sample output shows the system message you receive when a SSTP BPDU is received on a port and PVST+ simulation is disabled:

```
Message
SPANTREE_PVST_PEER_BLOCK: PVST BPDU detected on port %s [port number].

Severity
Critical

Explanation
A PVST+ peer was detected on the specified interface on the switch. PVST+
 simulation feature is disabled, as a result of which the interface was
moved to the spanning tree
Blocking state.

Action
```

Identify the PVST+ switch from the network which might be configured incorrectly.

The following sample output shows the system message you receive when peer inconsistency on the interface is cleared:

```
Message
SPANTREE_PVST_PEER_UNBLOCK: Unblocking port %s [port number].

Severity
Critical

Explanation
The interface specified in the error message has been restored to normal
 spanning tree state.

Action
None.
```

This example shows the spanning tree status when port **1/0/1** has been configured to disable PVST+ simulation and is currently in the peer type inconsistent state:

```
Switch# show spanning-tree
VLAN0010
  Spanning tree enabled protocol mstp
  Root ID Priority  32778
         Address   0002.172c.f400
         This bridge is the root
         Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
  Bridge ID Priority 32778 (priority 32768 sys-id-ext 10)
          Address  0002.172c.f400
         Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
         Aging Time 300
Interface        Role Sts Cost     Prio.Nbr Type
--------------- ---- --- --------- -------- -------------------------
Gi1/0/1          Desg BKN*4         128.270 P2p *PVST_Peer_Inc
```

This example shows the spanning tree summary when PVST+ simulation is enabled in the MSTP mode:

```
Switch# show spanning-tree summary
Switch is in mst mode (IEEE Standard)
Root bridge for: MST0
EtherChannel misconfig guard is enabled
Extended system ID is enabled
Portfast Default is disabled
PortFast BPDU Guard Default is disabled
Portfast BPDU Filter Default is disabled
Loopguard Default is disabled
UplinkFast is disabled
BackboneFast is disabled
Pathcost method used is long
PVST Simulation Default is enabled
Name                   Blocking Listening Learning Forwarding STP Active
---------------------- -------- --------- -------- ---------- ----------
MST0                          2         0        0          0          2
```

```
---------------------- -------- --------- -------- ---------- ----------
1 mst                         2         0        0          0          2
```

This example shows the spanning tree summary when PVST+ simulation is disabled in any STP mode:

```
Switch# show spanning-tree summary
Switch is in mst mode (IEEE Standard)
Root bridge for: MST0
EtherChannel misconfig guard is enabled
Extended system ID is enabled
Portfast Default is disabled
PortFast BPDU Guard Default is disabled
Portfast BPDU Filter Default is disabled
Loopguard Default is disabled
UplinkFast is disabled
BackboneFast is disabled
Pathcost method used is long
PVST Simulation Default is disabled
Name                    Blocking Listening Learning Forwarding STP Active
---------------------- -------- --------- -------- ---------- ----------
MST0                          2         0        0          0          2
---------------------- -------- --------- -------- ---------- ----------
1 mst                         2         0        0          0          2
```

This example shows the spanning tree summary when the switch is not in MSTP mode, that is, the switch is in PVST or Rapid-PVST mode. The output string displays the current STP mode:

```
Switch# show spanning-tree summary
Switch is in rapid-pvst mode
Root bridge for: VLAN0001, VLAN2001-VLAN2002
EtherChannel misconfig guard is enabled
Extended system ID is enabled
Portfast Default is disabled
PortFast BPDU Guard Default is disabled
Portfast BPDU Filter Default is disabled
Loopguard Default is disabled
UplinkFast is disabled
BackboneFast is disabled
Pathcost method used is short
PVST Simulation Default is enabled but inactive in rapid-pvst mode
Name                    Blocking Listening Learning Forwarding STP Active
---------------------- -------- --------- -------- ---------- ----------
VLAN0001                      2         0        0          0          2
VLAN2001                      2         0        0          0          2
VLAN2002                      2         0        0          0          2
---------------------- -------- --------- -------- ---------- ----------
3 vlans                       6         0        0          0          6
```

This example shows the interface details when PVST+ simulation is globally enabled, or the default configuration:

```
Switch# show spanning-tree interface1/0/1 detail
```

```
Port 269 (GigabitEthernet1/0/1) of VLAN0002 is forwarding
   Port path cost 4, Port priority 128, Port Identifier 128.297.
   Designated root has priority 32769, address 0013.5f20.01c0
   Designated bridge has priority 32769, address 0013.5f20.01c0
   Designated port id is 128.297, designated path cost 0
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   Link type is point-to-point by default
   PVST Simulation is enabled by default
   BPDU: sent 132, received 1
```

This example shows the interface details when PVST+ simulation is globally disabled:

```
Switch# show spanning-tree interface1/0/1 detail
Port 269 (GigabitEthernet1/0/1) of VLAN0002 is forwarding
   Port path cost 4, Port priority 128, Port Identifier 128.297.
   Designated root has priority 32769, address 0013.5f20.01c0
   Designated bridge has priority 32769, address 0013.5f20.01c0
   Designated port id is 128.297, designated path cost 0
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   Link type is point-to-point by default
   PVST Simulation is disabled by default
   BPDU: sent 132, received 1
```

This example shows the interface details when PVST+ simulation is explicitly enabled on the port:

```
Switch# show spanning-tree interface1/0/1 detail
Port 269 (GigabitEthernet1/0/1) of VLAN0002 is forwarding
   Port path cost 4, Port priority 128, Port Identifier 128.297.
   Designated root has priority 32769, address 0013.5f20.01c0
   Designated bridge has priority 32769, address 0013.5f20.01c0
   Designated port id is 128.297, designated path cost 0
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   Link type is point-to-point by default
   PVST Simulation is enabled
   BPDU: sent 132, received 1
```

This example shows the interface details when the PVST+ simulation feature is disabled and a PVST Peer inconsistency has been detected on the port:

```
Switch# show spanning-tree interface1/0/1 detail
Port 269 (GigabitEthernet1/0/1) of VLAN0002 is broken (PVST Peer Inconsistent)
   Port path cost 4, Port priority 128, Port Identifier 128.297.
   Designated root has priority 32769, address 0013.5f20.01c0
   Designated bridge has priority 32769, address 0013.5f20.01c0
   Designated port id is 128.297, designated path cost 0
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   Link type is point-to-point by default
   PVST Simulation is disabled
   BPDU: sent 132, received 1
```

Examples: Detecting Unidirectional Link Failure

This example shows the spanning tree status when port **1/0/1 detail** has been configured to disable PVST+ simulation and the port is currently in the peer type inconsistent state:

```
Switch# show spanning-tree
VLAN0010
```

```
  Spanning tree enabled protocol rstp
 Root ID    Priority 32778
            Address  0002.172c.f400
            This bridge is the root
            Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
 Bridge ID  Priority 32778 (priority 32768 sys-id-ext 10)
            Address  0002.172c.f400
            Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
            Aging Time 300

Interface        Role Sts Cost      Prio.Nbr Type
---------------- ---- --- --------- -------- ------------------------
Gi1/0/1          Desg BKN 4          128.270  P2p Dispute
```

This example shows the interface details when a dispute condition is detected:

```
Switch# show spanning-tree interface1/0/1 detail
Port 269 (GigabitEthernet1/0/1) of VLAN0002 is designated blocking (dispute)
   Port path cost 4, Port priority 128, Port Identifier 128.297.
   Designated root has priority 32769, address 0013.5f20.01c0
   Designated bridge has priority 32769, address 0013.5f20.01c0
   Designated port id is 128.297, designated path cost 0
   Timers: message age 0, forward delay 0, hold 0
   Number of transitions to forwarding state: 1
   Link type is point-to-point by default
   BPDU: sent 132, received 1
```

Additional References for MSTP

Related Documents

Related Topic	Document Title
Layer 2 commands	*Catalyst 2960-XR Switch Layer 2 Command Reference*

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for MSTP

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Configuring Optional Spanning-Tree Features

Restriction for Optional Spanning-Tree Features

- PortFast minimizes the time that interfaces must wait for spanning tree to converge, so 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.

Information About 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.

Figure 22: PortFast-Enabled Interfaces

You can use PortFast on interfaces connected to a single workstation or server to allow those devices to immediately connect to the network, rather than waiting for the spanning tree to

converge.
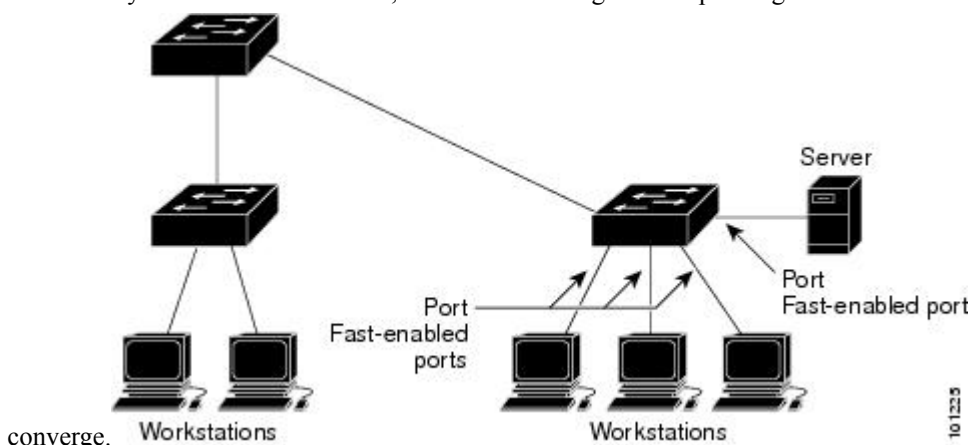
Interfaces connected to a single workstation or server should not receive bridge protocol data units (BPDUs). An interface with PortFast enabled goes through the normal cycle of spanning-tree status changes when the switch is restarted.

You can enable this feature by enabling it on either the interface or on all nontrunking ports.

BPDU Guard

The Bridge Protocol Data Unit (BPDU) guard feature can be globally enabled on the switch or can be enabled per port, but the feature operates with some differences.

When you enable BPDU guard at the global level on PortFast-enabled ports, 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 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.

When you enable BPDU guard at the interface level on any port without also enabling the PortFast feature, and 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.

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.

Enabling BPDU filtering on PortFast-enabled interfaces at the global level keeps those 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.

Enabling BPDU filtering on an interface without also enabling the PortFast feature keeps 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.

UplinkFast

Figure 23: Switches in a Hierarchical Network

Switches in hierarchical networks can be grouped into backbone switches, distribution switches, and access switches. This complex network has distribution switches and access switches that each have at least one redundant link that spanning tree blocks to prevent

loops.
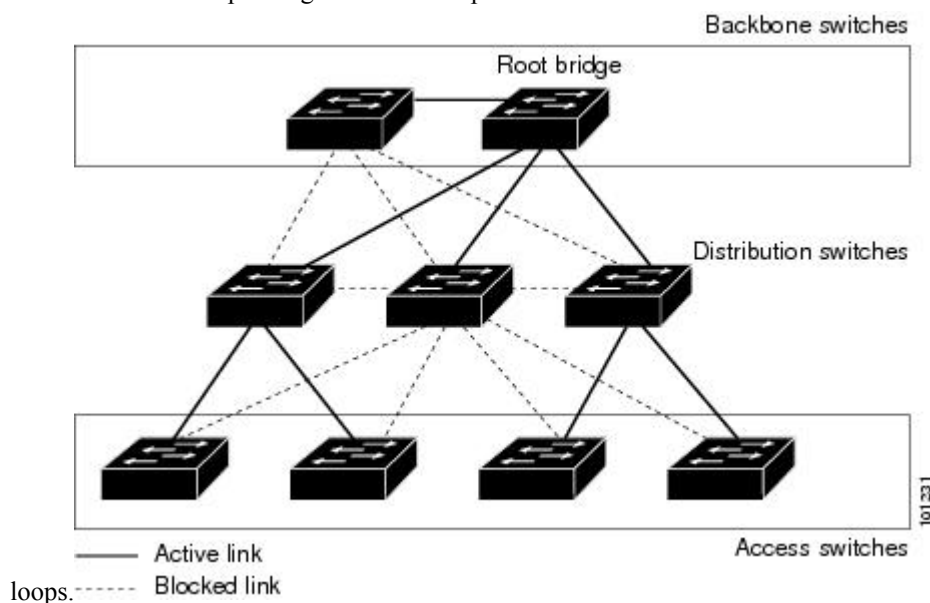
If a switch loses connectivity, it begins using the alternate paths as soon as the spanning tree selects a new root port. You can accelerate the choice of a new root port when a link or switch fails or when the spanning tree reconfigures itself by enabling UplinkFast. The root port transitions to the 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 24: UplinkFast Example Before Direct Link Failure

This topology has 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
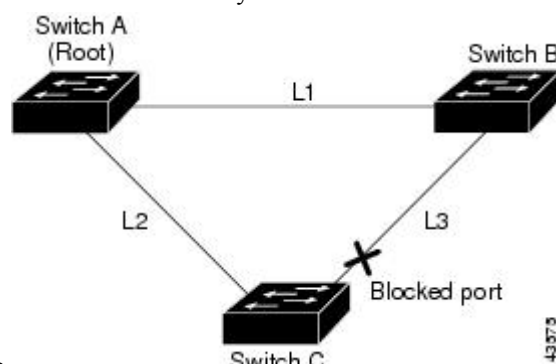
a blocking state.

Figure 25: UplinkFast Example After 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. This change takes approximately 1 to
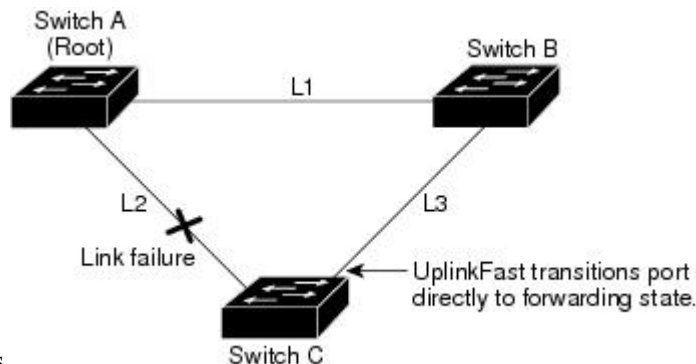
5 seconds.

Cross-Stack UplinkFast

Cross-Stack UplinkFast (CSUF) provides a fast spanning-tree transition (fast convergence in less than 1 second under normal network conditions) across a switch stack. During the fast transition, an alternate redundant link on the switch stack is placed in the forwarding state without causing temporary spanning-tree loops or loss of connectivity to the backbone. With this feature, you can have a redundant and resilient network in some configurations. CSUF is automatically enabled when you enable the UplinkFast feature.

CSUF might not provide a fast transition all the time; in these cases, the normal spanning-tree transition occurs, completing in 30 to 40 seconds. For more information, see Related Topics.

How Cross-Stack UplinkFast Works

Cross-Stack UplinkFast (CSUF) ensures that one link in the stack is elected as the path to the root.

Figure 26: Cross-Stack UplinkFast Topology

The stack-root port on Switch 1 provides the path to the root of the spanning tree. The alternate stack-root ports on Switches 2 and 3 can provide an alternate path to the spanning-tree root if the current stack-root switch fails or if its link to the spanning-tree root fails.

Link 1, the root link, is in the spanning-tree forwarding state. Links 2 and 3 are alternate redundant links that are in the spanning-tree blocking state. If Switch 1 fails, if its stack-root port fails, or if Link 1 fails, CSUF selects either the alternate stack-root port on Switch 2 or Switch 3 and puts it into the forwarding state in less than 1 second.
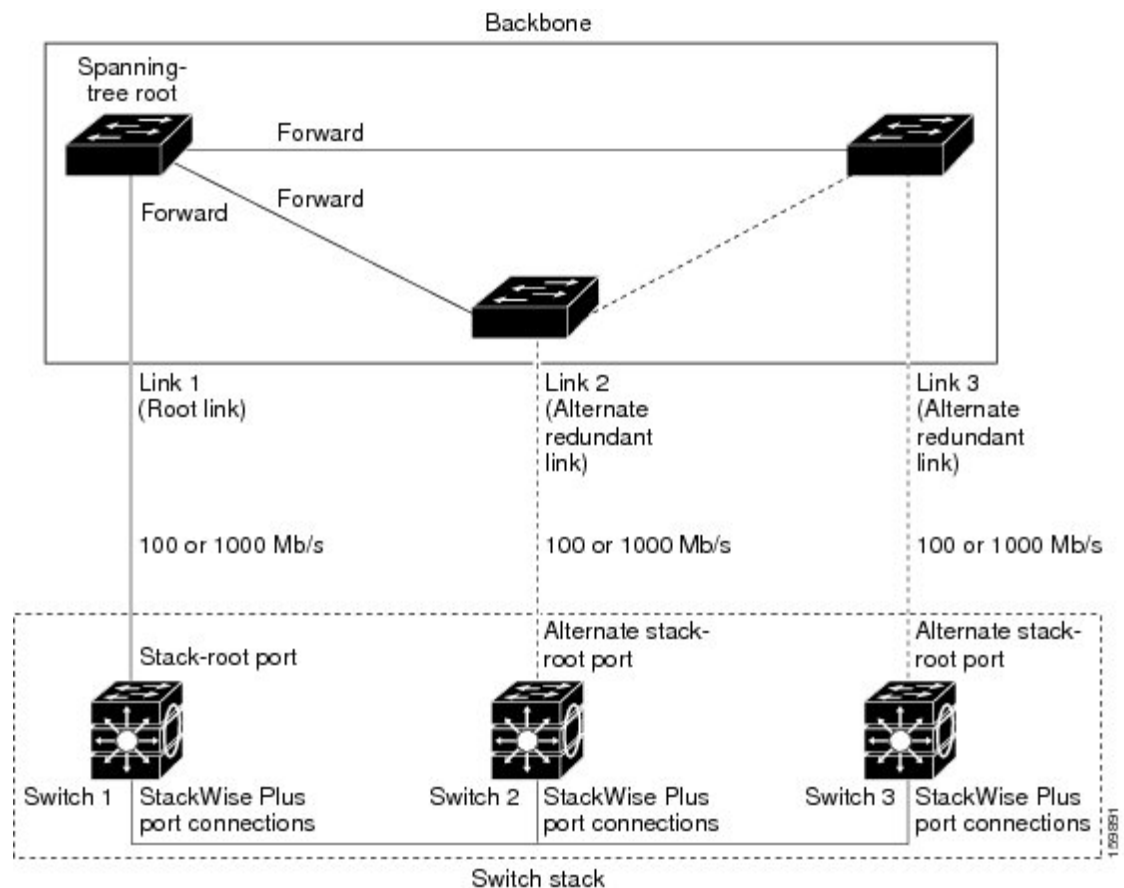
When certain link loss or spanning-tree events occur (described in the following topic), the Fast Uplink Transition Protocol uses the neighbor list to send fast-transition requests to stack members.

The switch sending the fast-transition request needs to do a fast transition to the forwarding state of a port that it has chosen as the root port, and it must obtain an acknowledgment from each stack switch before performing the fast transition.

Each switch in the stack decides if the sending switch is a better choice than itself to be the stack root of this spanning-tree instance by comparing the root, cost, and bridge ID. If the sending switch is the best choice as the stack root, each switch in the stack returns an acknowledgment; otherwise, it sends a fast-transition request. The sending switch then has not received acknowledgments from all stack switches.

When acknowledgments are received from all stack switches, the Fast Uplink Transition Protocol on the sending switch immediately transitions its alternate stack-root port to the forwarding state. If acknowledgments from all stack switches are not obtained by the sending switch, the normal spanning-tree transitions (blocking, listening, learning, and forwarding) take place, and the spanning-tree topology converges at its normal rate (2 * forward-delay time + max-age time).

The Fast Uplink Transition Protocol is implemented on a per-VLAN basis and affects only one spanning-tree instance at a time.

Events That Cause Fast Convergence

Depending on the network event or failure, the CSUF fast convergence might or might not occur.

Fast convergence (less than 1 second under normal network conditions) occurs under these circumstances:

- The stack-root port link fails.

  If two switches in the stack have alternate paths to the root, only one of the switches performs the fast transition.

- The failed link, which connects the stack root to the spanning-tree root, recovers.

- A network reconfiguration causes a new stack-root switch to be selected.

- A network reconfiguration causes a new port on the current stack-root switch to be chosen as the stack-root port.

> **Note** The fast transition might not occur if multiple events occur simultaneously. For example, if a stack member is powered off, and at the same time, the link connecting the stack root to the spanning-tree root comes back up, the normal spanning-tree convergence occurs.

Normal spanning-tree convergence (30 to 40 seconds) occurs under these conditions:

- The stack-root switch is powered off, or the software failed.

- The stack-root switch, which was powered off or failed, is powered on.

- A new switch, which might become the stack root, is added to the stack.

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 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 maximum aging time (default is 20 seconds).

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 to learn if any stack member has an alternate root to the root switch and waits for an RLQ reply from other switches in the network and in the stack. The switch sends the RLQ request on all alternate paths and waits for an RLQ reply from other switches in the network.

When a stack member receives an RLQ reply from a nonstack member on a blocked interface and the reply is destined for another nonstacked switch, it forwards the reply packet, regardless of the spanning-tree interface state.

When a stack member receives an RLQ reply from a nonstack member and the response is destined for the stack, the stack member forwards the reply so that all the other stack members receive it.

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 27: BackboneFast Example Before Indirect Link Failure

This is 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
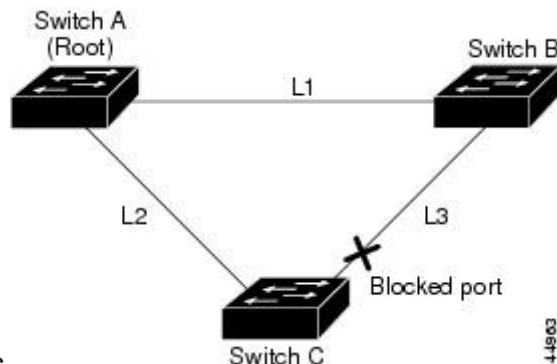
B is in the blocking state.

Figure 28: BackboneFast Example After Indirect Link Failure

If link L1 fails, 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. BackboneFast reconfigures the topology to account for the failure of link

L1.
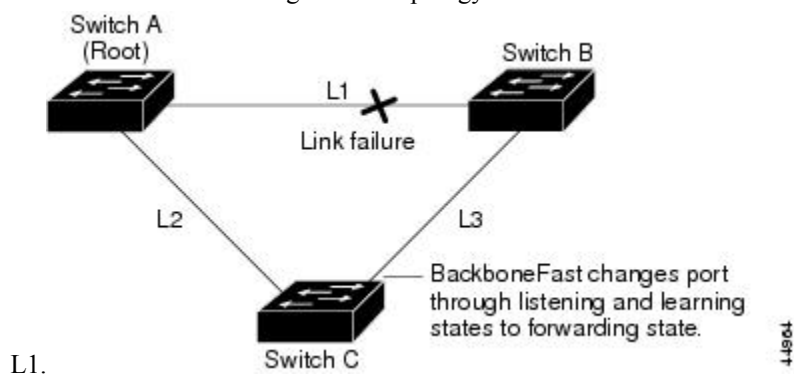
Figure 29: Adding a Switch in a Shared-Medium Topology

If a new switch is introduced into a shared-medium topology, 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
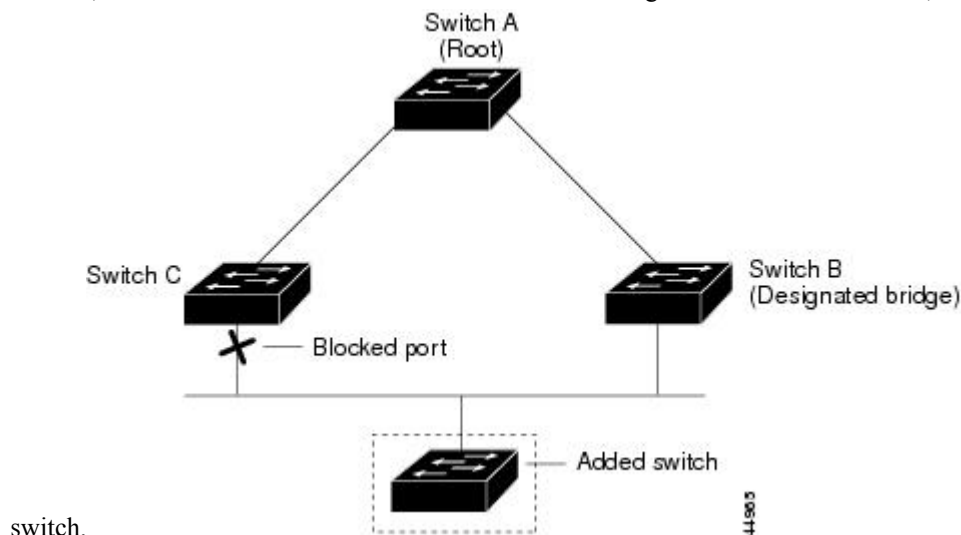
switch.

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.

Root Guard

Figure 30: Root Guard in a Service-Provider Network

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. 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.
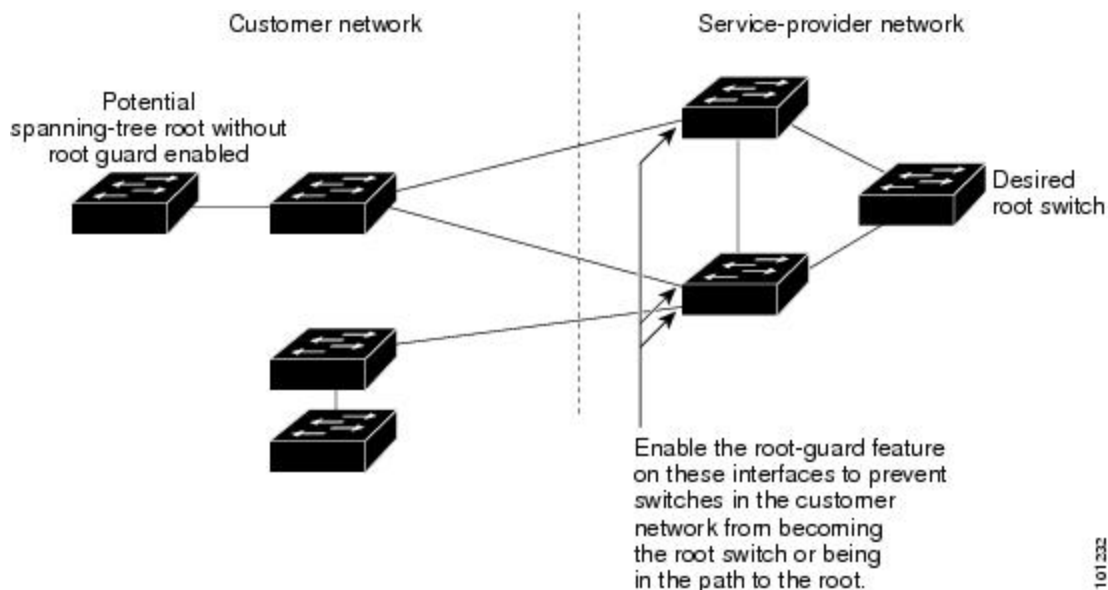
If a switch outside the SP network becomes the root switch, the interface is blocked (root-inconsistent state), and spanning tree selects a new root switch. The customer's switch does not become the root switch and is not in the path to the root.

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.

⚠️

Caution Misuse of the root guard feature can cause a loss of connectivity.

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.

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.

STP PortFast Port Types

You can configure a spanning tree port as an edge port, a network port, or a normal port. A port can be in only one of these states at a given time. The default spanning tree port type is normal. You can configure the port type either globally or per interface.

Depending on the type of device to which the interface is connected, you can configure a spanning tree port as one of these port types:

- A PortFast edge port—is connected to a Layer 2 host. This can be either an access port or an edge trunk port (**portfast edge trunk**). This type of port interface immediately transitions to the forwarding state, bypassing the listening and learning states. Use PortFast edge on Layer 2 access ports connected to a single workstation or server to allow those devices to connect to the network immediately, rather than waiting for spanning tree to converge.

  Even if the interface receives a bridge protocol data unit (BPDU), spanning tree does not place the port into the blocking state. Spanning tree sets the port's operating state to *non-port fast* even if the configured state remains *port fast edge* and starts participating in the topology change.

  > **Note** If you configure a port connected to a Layer 2 switch or bridge as an edge port, you might create a bridging loop.

- A PortFast network port—is connected only to a Layer 2 switch or bridge. Bridge Assurance is enabled only on PortFast network ports. For more information, refer to *Bridge Assurance*.

  > **Note** If you configure a port that is connected to a Layer 2 host as a spanning tree network port, the port will automatically move into the blocking state.

- A PortFast normal port—is the default type of spanning tree port.

  > **Note** Beginning with Cisco IOS Release 15.2(4)E, or IOS XE 3.8.0E, if you enter the **spanning-tree portfast** [trunk] command in the global or interface configuration mode, the system automatically saves it as **spanning-tree portfast edge** [trunk].

Bridge Assurance

You can use Bridge Assurance to help prevent looping conditions that are caused by unidirectional links (one-way traffic on a link or port), or a malfunction in a neighboring switch. Here a malfunction refers to a switch that is not able to run STP any more, while still forwarding traffic (a brain dead switch).

BPDUs are sent out on all operational network ports, including alternate and backup ports, for each hello time period. Bridge Assurance monitors the receipt of BPDUs on point-to-point links on all network ports. When a port does not receive BPDUs within the alloted hello time period, the port is put into a blocked state (the same as a port inconsistent state, which stops forwarding of frames). When the port resumes receipt of BPDUs, the port resumes normal spanning tree operations.

Note Only Rapid PVST+ and MST spanning tree protocols support Bridge Assurance. PVST+ does not support Bridge Assurance.

The following example shows how Bridge Assurance protects your network from bridging loops.

The following figure shows a network with normal STP topology.

Figure 31: Network with Normal STP Topology
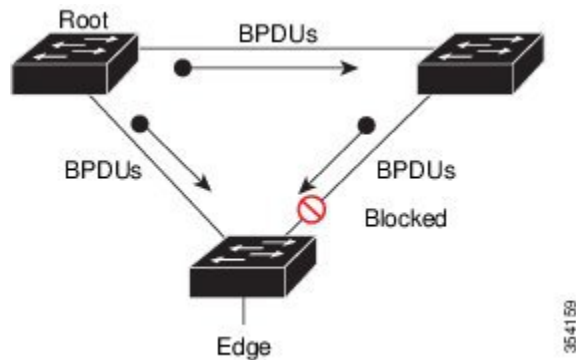
The following figure demonstrates a potential network problem when the device fails (brain dead) and Bridge Assurance is not enabled on the network.

Figure 32: Network Loop Due to a Malfunctioning Switch
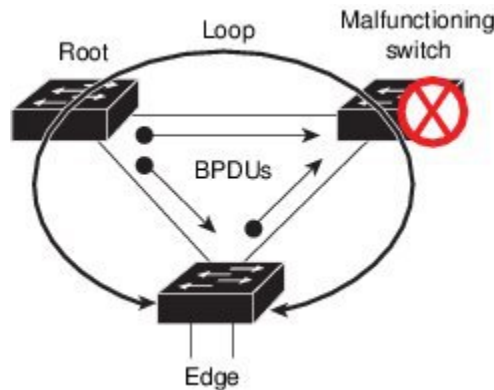
The following figure shows the network with Bridge Assurance enabled, and the STP topology progressing normally with bidirectional BDPUs issuing from every STP network port.

Figure 33: Network with STP Topology Running Bridge Assurance
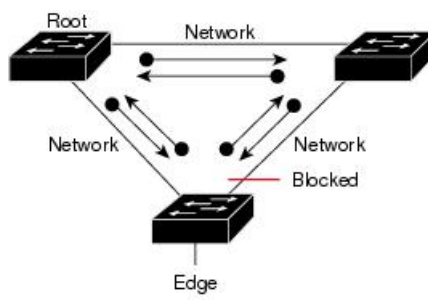
The following figure shows how the potential network problem shown in figure *Network Loop Due to a Malfunctioning Switch* does not occur when you have Bridge Assurance enabled on your network.

Figure 34: Network Problem Averted with Bridge Assurance Enabled
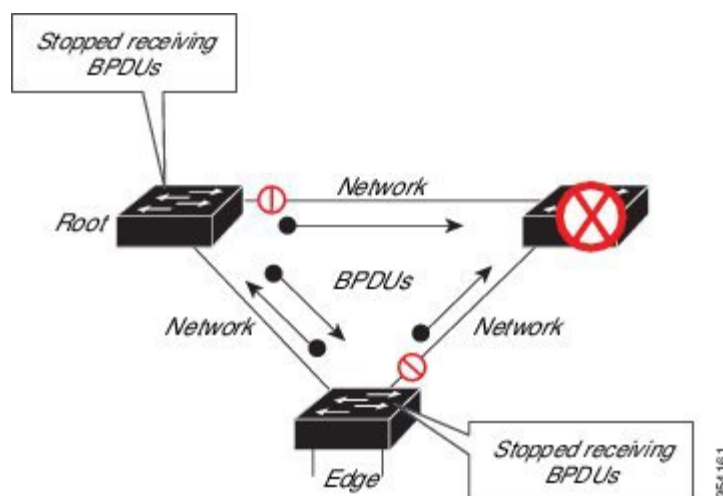
The system generates syslog messages when a port is block and unblocked. The following sample output shows the log that is generated for each of these states:

BRIDGE_ASSURANCE_BLOCK

```
Sep 17 09:48:16.249 PDT: %SPANTREE-2-BRIDGE_ASSURANCE_BLOCK: Bridge Assurance blocking port
 GigabitEthernet1/0/1 on VLAN0001.
```

BRIDGE_ASSURANCE_UNBLOCK

```
Sep 17 09:48:58.426 PDT: %SPANTREE-2-BRIDGE_ASSURANCE_UNBLOCK: Bridge Assurance unblocking
 port GigabitEthernet1/0/1 on VLAN0001.
```

Follow these guidelines when enabling Bridge Assurance:

- It can only be enabled or disabled globally.

- It applies to all operational network ports, including alternate and backup ports.

- Only Rapid PVST+ and MST spanning tree protocols support Bridge Assurance. PVST+ does not support Bridge Assurance.

- For Bridge Assurance to work properly, it must be supported and configured on both ends of a point-to-point link. If the device on one side of the link has Bridge Assurance enabled and the device on the other side does not, the connecting port is blocked and in a Bridge Assurance inconsistent state. We recommend that you enable Bridge Assurance throughout your network.

- To enable Bridge Assurance on a port, BPDU filtering and BPDU Guard must be disabled.

- You can enable Bridge Assurance in conjunction with Loop Guard.

- You can enable Bridge Assurance in conjunction with Root Guard. The latter is designed to provide a way to enforce the root bridge placement in the network.

How to Configure Optional Spanning-Tree Features

Enabling PortFast

An interface with the PortFast feature enabled is moved directly to the spanning-tree forwarding state without waiting for the standard forward-time delay.

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.

You can enable this feature if your switch is running PVST+, Rapid PVST+, or MSTP.

⚠️

Caution   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.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree portfast** [**trunk**]
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:**	Enables privileged EXEC mode. - Enter your password if prompted.

	Command or Action	Purpose
	Switch> **enable**	
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/2**	Specifies an interface to configure, and enters interface configuration mode.
Step 4	**spanning-tree portfast** [**trunk**] **Example:** Switch(config-if)# **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. 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. By default, PortFast is disabled on all interfaces.
Step 5	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.

What to do next

You can use the **spanning-tree portfast default** global configuration command to globally enable the PortFast feature on all nontrunking ports.

Enabling BPDU Guard

You can enable the BPDU guard feature if your switch is running PVST+, Rapid PVST+, or MSTP.

⚠️

Caution   Configure PortFast 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.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree portfast bpduguard default**
4. **interface** *interface-id*
5. **spanning-tree portfast**
6. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**spanning-tree portfast bpduguard default** **Example:** Switch(config)# **spanning-tree portfast bpduguard default**	Globally enables BPDU guard. By default, BPDU guard is disabled.
Step 4	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/2**	Specifies the interface connected to an end station, and enters interface configuration mode.
Step 5	**spanning-tree portfast** **Example:** Switch(config-if)# **spanning-tree portfast**	Enables the PortFast feature.
Step 6	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.

What to do next

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.

You also can use the **spanning-tree bpduguard enable** interface configuration command to enable BPDU guard on any port without also enabling the PortFast feature. When the port receives a BPDU, it is put it in the error-disabled state.

Enabling BPDU Filtering

You can also use the **spanning-tree bpdufilter enable** interface configuration command to enable BPDU filtering on any interface 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 if your switch is running PVST+, Rapid PVST+, or MSTP.

⚠

Caution   Configure PortFast 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.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree portfast bpdufilter default**
4. **interface** *interface-id*
5. **spanning-tree portfast**
6. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:**	Enters global configuration mode.

	Command or Action	**Purpose**
	Switch# **configure terminal**	
Step 3	**spanning-tree portfast bpdufilter default**	Globally enables BPDU filtering.
	Example:	By default, BPDU filtering is disabled.
	Switch(config)# **spanning-tree portfast bpdufilter default**	
Step 4	**interface** *interface-id*	Specifies the interface connected to an end station, and enters interface configuration mode.
	Example:	
	Switch(config)# **interface gigabitethernet1/0/2**	
Step 5	**spanning-tree portfast**	Enables the PortFast feature on the specified interface.
	Example:	
	Switch(config-if)# **spanning-tree portfast**	
Step 6	**end**	Returns to privileged EXEC mode.
	Example:	
	Switch(config-if)# **end**	

Enabling UplinkFast for Use with Redundant Links

> **Note**   When you enable UplinkFast, it affects all VLANs on the switch or switch stack. You cannot configure UplinkFast on an individual VLAN.

You can configure the UplinkFast or the Cross-Stack UplinkFast (CSUF) feature for Rapid PVST+ or for the MSTP, but the feature remains disabled (inactive) until you change the spanning-tree mode to PVST+.

This procedure is optional. Follow these steps to enable UplinkFast and CSUF.

Before you begin

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 using the **no spanning-tree vlan** *vlan-id* **priority** global configuration command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree uplinkfast** [**max-update-rate** *pkts-per-second*]

4. end

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**spanning-tree uplinkfast** [**max-update-rate** *pkts-per-second*] **Example:** Switch(config)# **spanning-tree uplinkfast max-update-rate 200**	Enables UplinkFast. (Optional) For *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. When you enter this command, CSUF also is enabled on all nonstack port interfaces.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

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.

When you enable the UplinkFast feature using these instructions, CSUF is automatically globally enabled on nonstack port interfaces.

Disabling UplinkFast

This procedure is optional.

Follow these steps to disable UplinkFast and Cross-Stack UplinkFast (CSUF).

Before you begin

UplinkFast must be enabled.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no spanning-tree uplinkfast**
4. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**no spanning-tree uplinkfast** **Example:** Switch(config)# **no spanning-tree uplinkfast**	Disables UplinkFast and CSUF on the switch and all of its VLANs.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

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.

When you disable the UplinkFast feature using these instructions, CSUF is automatically globally disabled on nonstack port interfaces.

Enabling BackboneFast

You can enable BackboneFast to detect indirect link failures and to start the spanning-tree reconfiguration sooner.

You can configure 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+.

This procedure is optional. Follow these steps to enable BackboneFast on the switch.

Before you begin

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.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree backbonefast**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**spanning-tree backbonefast** **Example:** Switch(config)# **spanning-tree backbonefast**	Enables BackboneFast.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Enabling EtherChannel Guard

You can enable EtherChannel guard to detect an EtherChannel misconfiguration if your switch is running PVST+, Rapid PVST+, or MSTP.

This procedure is optional.

Follow these steps to enable EtherChannel Guard on the switch.

SUMMARY STEPS

1. **enable**

2. **configure terminal**
3. **spanning-tree etherchannel guard misconfig**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**spanning-tree etherchannel guard misconfig** **Example:** Switch(config)# **spanning-tree etherchannel guard misconfig**	Enables EtherChannel guard.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

What to do next

You can use the **show interfaces status err-disabled** privileged EXEC command to show which switch ports are disabled because of an EtherChannel misconfiguration. On the remote device, you can enter the **show etherchannel summary** privileged EXEC command to verify the EtherChannel configuration.

After the configuration is corrected, enter the **shutdown** and **no shutdown** interface configuration commands on the port-channel interfaces that were misconfigured.

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.

You can enable this feature if your switch is running PVST+, Rapid PVST+, or MSTP.

This procedure is optional.

Follow these steps to enable root guard on the switch.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree guard root**
5. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/2**	Specifies an interface to configure, and enters interface configuration mode.
Step 4	**spanning-tree guard root** **Example:** Switch(config-if)# **spanning-tree guard root**	Enables root guard on the interface. By default, root guard is disabled on all interfaces.
Step 5	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.

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.

You can enable this feature if your switch is running PVST+, Rapid PVST+, or MSTP.

This procedure is optional. Follow these steps to enable loop guard on the switch.

SUMMARY STEPS

1. Enter one of the following commands:
   - **show spanning-tree active**
   - **show spanning-tree mst**
2. **configure terminal**
3. **spanning-tree loopguard default**
4. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	Enter one of the following commands: • **show spanning-tree active** • **show spanning-tree mst** **Example:** Switch# **show spanning-tree active** or Switch# **show spanning-tree mst**	Verifies which interfaces are alternate or root ports.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**spanning-tree loopguard default** **Example:** Switch(config)# **spanning-tree loopguard default**	Enables loop guard. By default, loop guard is disabled.

	Command or Action	**Purpose**
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Enabling PortFast Port Types

This section describes the different steps to enable Portfast Port types.

Configuring the Default Port State Globally

To configure the default PortFast state, perform this task:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree portfast** [**edge** | **network** | **normal**] **default**
4. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**spanning-tree portfast** [**edge**	**network**	**normal**] **default** **Example:** Switch(config)# **spanning-tree portfast default**	Configures the default state for all interfaces on the switch. You have these options: • (Optional) **edge**—Configures all interfaces as edge ports. This assumes all ports are connected to hosts/servers. • (Optional) **network**—Configures all interfaces as spanning tree network ports. This assumes all ports are connected to switches and bridges. Bridge Assurance is enabled on all network ports by default.

	Command or Action	Purpose
		• (Optional) **normal**—Configures all interfaces normal spanning tree ports. These ports can be connected to any type of device.
		• **default**—The default port type is normal.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Configuring PortFast Edge on a Specified Interface

Interfaces configured as edge ports immediately transition to the forwarding state, without passing through the blocking or learning states, on linkup.

> **Note** Because the purpose of this type of port is to minimize the time that access ports must wait for spanning tree to converge, it is most effective when used on access ports. If you enable PortFast edge on a port connecting to another switch, you risk creating a spanning tree loop.

To configure an edge port on a specified interface, perform this task:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id* | **port-channel** *port_channel_number*
4. **spanning-tree portfast edge** [**trunk**]
5. end
6. **show running interface** *interface-id* | **port-channel** *port_channel_number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.

	Command or Action	**Purpose**
Step 3	**interface** *interface-id*	**port-channel** *port_channel_number* **Example:** Switch(config)# **interface gigabitethernet 1/0/1 | port-channel** *port_channel_number*	Specifies an interface to configure.
Step 4	**spanning-tree portfast edge** [**trunk**] **Example:** Switch(config-if)# **spanning-tree portfast trunk**	Enables edge behavior on a Layer 2 access port connected to an end workstation or server. • (Optional) **trunk**—Enables edge behavior on a trunk port. Use this keyword if the link is a trunk. Use this command only on ports that are connected to end host devices that terminate VLANs and from which the port should never receive STP BPDUs. Such end host devices include workstations, servers, and ports on routers that are not configured to support bridging. • Use the **no** version of the command to disable PortFast edge.
Step 5	end **Example:** Switch(config-if)# **end**	Exits configuration mode.
Step 6	**show running interface** *interface-id*	**port-channel** *port_channel_number* **Example:** Switch# **show running interface gigabitethernet 1/0/1|** **port-channel** *port_channel_number*	Verifies the configuration.

Configuring a PortFast Network Port on a Specified Interface

Ports that are connected to Layer 2 switches and bridges can be configured as network ports.

✎

Note Bridge Assurance is enabled only on PortFast network ports. For more information, refer to *Bridge Assurance*.

To configure a port as a network port, perform this task.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id* | **port-channel** *port_channel_number*
4. **spanning-tree portfast network**
5. end

6. **show running interface** *interface-id* | **port-channel** *port_channel_number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* | **port-channel** *port_channel_number* **Example:** Switch(config)# **interface gigabitethernet 1/0/1|** **port-channel** *port_channel_number*	Specifies an interface to configure.
Step 4	**spanning-tree portfast network** **Example:** Switch(config-if)# **spanning-tree portfast network**	Enables edge behavior on a Layer 2 access port connected to an end workstation or server. • Configures the port as a network port. If you have enabled Bridge Assurance globally, it automatically runs on a spanning tree network port. • Use the **no** version of the command to disable PortFast.
Step 5	end **Example:** Switch(config-if)# **end**	Exits configuration mode.
Step 6	**show running interface** *interface-id* | **port-channel** *port_channel_number* **Example:** Switch# **show running interface gigabitethernet 1/0/1 | port-channel** *port_channel_number*	Verifies the configuration.

Enabling Bridge Assurance

To configure the Bridge Assurance, perform the steps given below:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree bridge assurance**
4. **end**
5. **show spanning-tree summary**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**spanning-tree bridge assurance** **Example:** Switch(config)# **spanning-tree bridge assurance**	Enables Bridge Assurance on all network ports on the switch. Bridge Assurance is enabled by default. Use the **no** version of the command to disable the feature. Disabling Bridge Assurance causes all configured network ports to behave as normal spanning tree ports.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show spanning-tree summary** **Example:** Switch# **show spanning-tree summary**	Displays spanning tree information and shows if Bridge Assurance is enabled.

Examples

Examples: Configuring PortFast Edge on a Specified Interface

This example shows how to enable edge behavior on GigabitEthernet interface **1/0/1**:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# spanning-tree portfast edge
Switch(config-if)# end
Switch#
```

This example shows how to verify the configuration:

```
Switch# show running-config interface gigabitethernet1/0/1
Building configuration...
Current configuration:
!
interface GigabitEthernet1/0/1
no ip address
switchport
switchport access vlan 200
switchport mode access
spanning-tree portfast edge
end
```

This example shows how you can display that port GigabitEthernet **1/0/1** is currently in the edge state:

```
Switch# show spanning-tree vlan 200
VLAN0200
Spanning tree enabled protocol rstp
Root ID Priority 2
Address 001b.2a68.5fc0
Cost 3
Port 125 (GigabitEthernet5/9)
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Bridge ID Priority 2 (priority 0 sys-id-ext 2)
Address 7010.5c9c.5200
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Aging Time 0 sec
Interface Role Sts Cost Prio.Nbr Type
------------------ ---- --- --------- -------- --------------------------------
Gi1/0/1 Desg FWD 4 128.1 P2p Edge
```

Examples: Configuring a PortFast Network Port on a Specified Interface

This example shows how to configure GigabitEthernet interface **1/0/1** as a network port:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# spanning-tree portfast network
Switch(config-if)# end
Switch#
```

This example shows how to verify the configuration:

```
Switch# show running-config interface gigabitethernet1/0/1
Building configuration...
Current configuration:
!
interface GigabitEthernet1/0/1
no ip address
switchport
switchport access vlan 200
switchport mode access
spanning-tree portfast network
end
```

This example shows the output for show spanning-tree vlan

```
Switch# show spanning-tree vlan
Sep 17 09:51:36.370 PDT: %SYS-5-CONFIG_I: Configured from console by console2

VLAN0002
  Spanning tree enabled protocol rstp
  Root ID    Priority    2
             Address     7010.5c9c.5200
             This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    2      (priority 0 sys-id-ext 2)
             Address     7010.5c9c.5200
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
             Aging Time   0   sec

Interface          Role Sts Cost      Prio.Nbr Type
------------------ ---- --- --------- -------- --------------------------------
Gi1/0/1            Desg FWD 4         128.1    P2p Edge
Po4                Desg FWD 3         128.480  P2p Network
Gi4/0/1            Desg FWD 4         128.169  P2p Edge
Gi4/0/47           Desg FWD 4         128.215  P2p Network

Switch#
```

Example: Configuring Bridge Assurance

This output shows port GigabitEthernet **1/0/1** has been configured as a network port and it is currently in the Bridge Assurance inconsistent state.

> ✎
>
> **Note**   The output shows the port type as network and *BA_Inc, indicating that the port is in an inconsistent state.

```
Switch# show spanning-tree
VLAN0010
Spanning tree enabled protocol rstp
Root ID Priority 32778
Address 0002.172c.f400
This bridge is the root
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Bridge ID Priority 32778 (priority 32768 sys-id-ext 10)
Address 0002.172c.f400
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Aging Time 300
Interface Role Sts   Cost    Prio.  Nbr    Type
--------------- ---- --- --------- -------- --------------------------------
Gi1/0/1    Desg BKN*4 128.270 Network, P2p *BA_Inc
```

The example shows the output for show spanning-tree summary.

```
Switch#sh spanning-tree summary
Switch is in rapid-pvst mode
Root bridge for: VLAN0001-VLAN0002, VLAN0128
EtherChannel misconfig guard          is enabled
Extended system ID                    is enabled
Portfast Default                      is network
Portfast Edge BPDU Guard Default      is disabled
Portfast Edge BPDU Filter Default     is disabled
```

```
Loopguard Default                       is enabled
PVST Simulation Default                 is enabled but inactive in rapid-pvst mode
Bridge Assurance                        is enabled
UplinkFast                              is disabled
BackboneFast                            is disabled
Configured Pathcost method used is short

Name                  Blocking Listening Learning Forwarding STP Active
--------------------- -------- --------- -------- ---------- ----------
VLAN0001                     0         0        0          5          5
VLAN0002                     0         0        0          4          4
VLAN0128                     0         0        0          4          4
--------------------- -------- --------- -------- ---------- ----------
3 vlans                      0         0        0         13         13

Switch#
```

Monitoring the Spanning-Tree Status

Table 32: Commands for Monitoring the Spanning-Tree Status

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.

Additional References for Optional Spanning Tree Features

Related Documents

Related Topic	Document Title
Layer 2 commands	Catalyst 2960-XR Switch Layer 2 Command Reference

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for Optional Spanning-Tree Features

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Configuring EtherChannels

Restrictions for EtherChannels

• Layer 3 EtherChannels are not supported if the switch is running the LAN Base feature set.

• All ports in an EtherChannel must be assigned to the same VLAN or they must be configured as trunk ports.

• When the ports in an EtherChannel are configured as trunk ports, all the ports must be configured with the same mode (either Inter-Switch Link [ISL] or IEEE 802.1Q).

• Port Aggregation Protocol (PAgP) can be enabled only in single-switch EtherChannel configurations; PAgP cannnot be enabled on cross-stack EtherChannels.

Information About EtherChannels

EtherChannel Overview

EtherChannel provides fault-tolerant high-speed links between switches, routers, and servers. You can use the EtherChannel 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.

An EtherChannel consists of individual Ethernet links bundled into a single logical link.

Figure 35: Typical EtherChannel Configuration
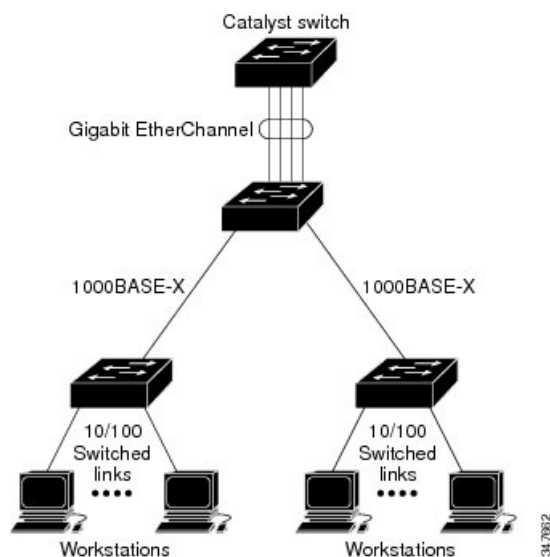
The EtherChannel provides full-duplex bandwidth up to 8 Gb/s (Gigabit EtherChannel) or 80 Gb/s (10-Gigabit EtherChannel) between your switch and another switch or host.

Each EtherChannel can consist of up to eight compatibly configured Ethernet ports.

The LAN Base feature set supports up to 24 EtherChannels.

The IP Lite feature set supports up to 48 EtherChannels.

EtherChannel Modes

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. If the remote port cannot negotiate an EtherChannel, the local port is put into an independent state and continues 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.

EtherChannel on Switches

You can create an EtherChannel on a switch, on a single switch in the stack, or on multiple switches in the stack (known as cross-stack EtherChannel).

Figure 36: Single-Switch EtherChannel
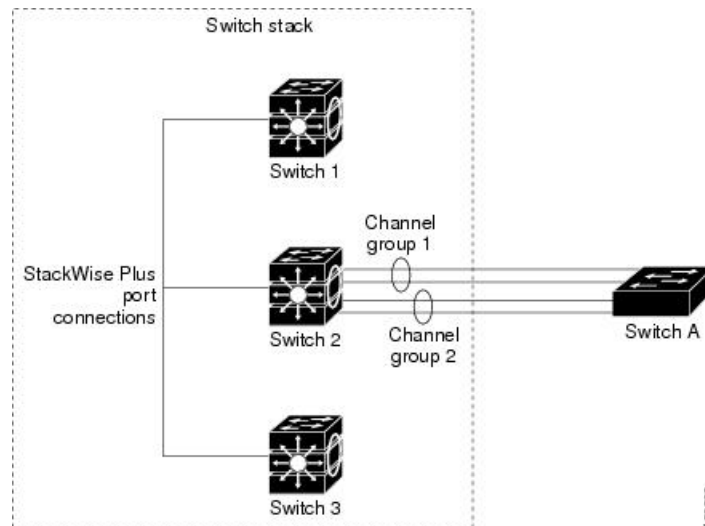
Figure 37: Cross-Stack EtherChannel
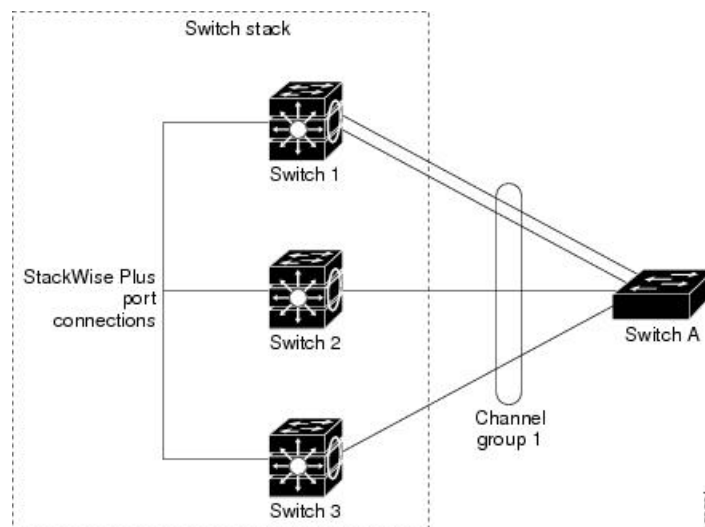
EtherChannel Link Failover

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.

Channel Groups and Port-Channel Interfaces

An EtherChannel comprises a channel group and a port-channel interface. The channel group binds physical ports to the port-channel interface. Configuration changes applied to the port-channel interface apply to all the physical ports bound together in the channel group.

Figure 38: Relationship of Physical Ports, Channel Group and Port-Channel Interface

The **channel-group** command binds the physical port and the port-channel interface together. Each EtherChannel has a port-channel logical interface numbered from 1 to 48. This port-channel interface number corresponds to the one specified with the **channel-group** interface configuration command.
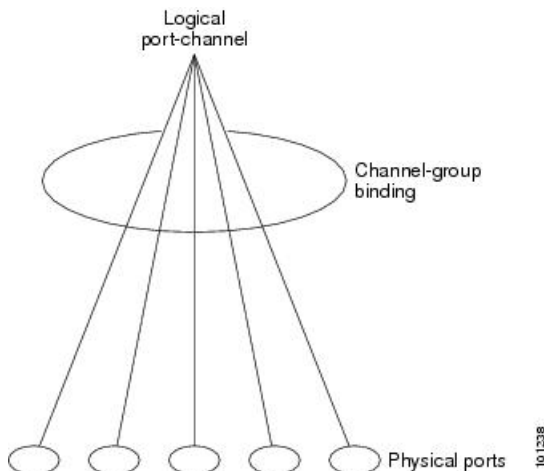
- With Layer 2 ports, use the **channel-group** interface configuration command to dynamically create the port-channel interface.

  You also can use the **interface port-channel** *port-channel-number* global configuration command to manually create the port-channel 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.

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 or switch stack learns the identity of partners capable of supporting PAgP and the capabilities of each port. It then dynamically groups similarly configured ports (on a single switch in the stack) 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

PAgP modes specify whether a port can send PAgP packets, which start PAgP negotiations, or only respond to PAgP packets received.

Table 33: 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. This mode is not supported when the EtherChannel members are from different switches in the switch stack (cross-stack EtherChannel).
desirable	Places a port into an active negotiating state, in which the port starts negotiations with other ports by sending PAgP packets. This mode is not supported when the EtherChannel members are from different switches in the switch stack (cross-stack EtherChannel).

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, based on trunk 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.

Silent Mode

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 specify **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, such as the Catalyst 1900 switch.

When the link partner of the switch is a physical learner, 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 physcial learner 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 its 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 or switch stack 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

LACP modes specify whether a port can send LACP packets or only receive LACP packets.

Table 34: 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, based on trunk 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 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. You can specify one of several different load-balancing modes, including load distribution based on MAC addresses, IP addresses, source addresses, 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.

MAC Address Forwarding

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.

IP Address Forwarding

With source-IP address-based forwarding, packets are distributed across the ports in the EtherChannel based on the source-IP address of the incoming packet. To provide load balancing, packets from different IP addresses use different ports in the channel, and packets from the same IP address use the same port in the channel.

With destination-IP address-based forwarding, packets are distributed across the ports in the EtherChannel based on the destination-IP address of the incoming packet. 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. 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 distributed across the ports in the EtherChannel 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.

Load-Balancing Advantages

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.

Figure 39: Load Distribution and Forwarding Methods

In the following figure, an EtherChannel of 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.
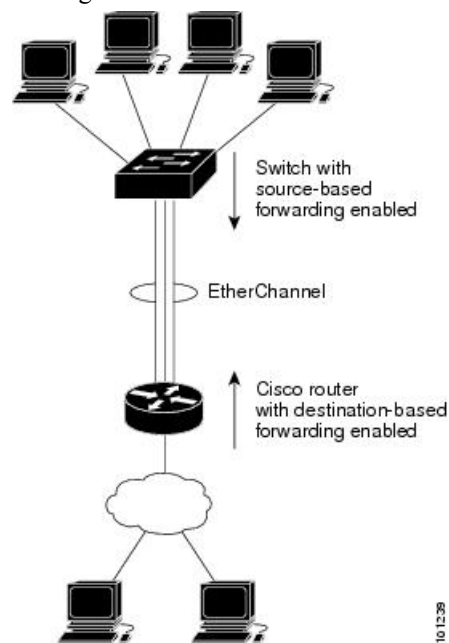
Use the option that provides the greatest variety in your configuration. For example, if the traffic on a channel is going only to a single MAC address, using the destination-MAC address always chooses the same link in the channel. Using source addresses or IP addresses might result in better load-balancing.

EtherChannel Load Deferral Overview

In an Instant Access system, the EtherChannel Load Deferral feature allows ports to be bundled into port channels, but prevents the assignment of group mask values to these ports. This prevents the traffic from being forwarded to new instant access stack members and reduce data loss following a stateful swtichover (SSO).

Cisco Catalyst Instant Access creates a single network touch point and a single point of configuration across distribution and access layer switches. Instant Access enables the merging of physical distribution and access layer switches into a single logical entity with a single point of configuration, management, and troubleshooting. The following illustration represents a sample network where an Instant Access system interacts with a switch (Catalyst 2960-X Series Switches) that is connected via a port channel to stacked clients (Member 1 and Member 2).

When the EtherChannel Load Deferral feature is configured and a new Instant Access client stack member comes up, ports of this newly-joined stack member is bundled into the port channel. In the transition period, the data path is not fully established on the distribution switch (Catalyst 6000 Series Switches), and traffic originating from the access layer switch (Catalyst 2960-X Series Switches) reaches the non-established ports and the traffic gets lost.

When load share deferral is enabled on a port channel, the assignment of a member port's load share is delayed for a period that is configured globally by the **port-channel load-defer** command. During the deferral period, the load share of a deferred member port is set to 0. In this state, the deferred port is capable of receiving data and control traffic, and of sending control traffic, but the port is prevented from sending data traffic to the virtual switching system (VSS). Upon expiration of the global deferral timer, the deferred member port exits the deferral state and the port assumes its normal configured load share.

Load share deferral is applied only if at least one member port of the port channel is currently active with a nonzero load share. If a port enabled for load share deferral is the first member bringing up the EtherChannel, the deferral feature does not apply and the port will forward traffic immediately.

This feature is enabled on a per port-channel basis; however, the load deferral timer is configured globally and not per port-channel. As a result, when a new port is bundled, the timer starts only if it is not already running. If some other ports are already deferred then the new port will be deferred only for the remaining amount of time.

The load deferral is stopped as soon as a member in one of the deferred port channels is unbundled. As a result, all the ports that were deferred is assigned a group-mask in the event of an unbundling during the deferral period.

Note When you try to enable this feature on a stack member switch, the following message is displayed:

```
Load share deferral is supported only on stand-alone stack.
```

EtherChannel and Switch Stacks

If a stack member that has ports participating in an EtherChannel fails or leaves the stack, the active switch removes the failed stack member switch ports from the EtherChannel. The remaining ports of the EtherChannel, if any, continue to provide connectivity.

When a switch is added to an existing stack, the new switch receives the running configuration from the active switch and updates itself with the EtherChannel-related stack configuration. The stack member also receives the operational information (the list of ports that are up and are members of a channel).

When two stacks merge that have EtherChannels configured between them, self-looped ports result. Spanning tree detects this condition and acts accordingly. Any PAgP or LACP configuration on a winning switch stack is not affected, but the PAgP or LACP configuration on the losing switch stack is lost after the stack reboots.

For a mixed stack containing one or more Catalyst 2960-S switches, we recommend that you configure no more than six EtherChannels on the stack.

Switch Stack and PAgP

With PAgP, if the active switch fails or leaves the stack, the standby switch becomes the new active switch. The new active switch synchronizes the configuration of the stack members to that of the active switch. The PAgP configuration is not affected after an active switch change unless the EtherChannel has ports residing on the old active switch.

Switch Stacks and LACP

With LACP, the system ID uses the stack MAC address from the active switch, and if the active switch changes, the LACP system ID can change. If the LACP system ID changes, the entire EtherChannel will flap, and there will be an STP reconvergence. Use the **stack-mac persistent timer** command to control whether or not the stack MAC address changes during a active switch failover.

Default EtherChannel Configuration

The default EtherChannel configuration is described in this table.

Table 35: Default EtherChannel Configuration

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 or stack 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 24 EtherChannels on the switch or switch stack.

- In a mixed switch stack that contains one or more Catalyst 2960-S switches, do not configure more than six EtherChannels on the switch stack.

- Configure a PAgP EtherChannel with up to eight Ethernet ports of the same type.

- Configure a LACP EtherChannel with 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.

- 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 or on different switches in the stack. Individual EtherChannel groups can run either PAgP or LACP, but they cannot interoperate.

- Do not configure a secure port as part of an EtherChannel or the reverse.

- 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 cross-stack EtherChannel configurations, ensure that all ports targeted for the EtherChannel are either configured for LACP or are manually configured to be in the channel group using the **channel-group** *channel-group-number* **mode on** interface configuration command. The PAgP protocol is not supported on cross- stack EtherChannels.

Layer 2 EtherChannel Configuration Guidelines

When configuring Layer 2 EtherChannels, follow these guidelines:

- Assign all ports in the EtherChannel to the same VLAN, or configure them as trunks. Ports with different native VLANs cannot form an EtherChannel.

- 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.

Layer 3 EtherChannel Configuration Guidelines

- For Layer 3 EtherChannels, assign the Layer 3 address to the port-channel logical interface, not to the physical ports in the channel.

Auto-LAG

The auto-LAG feature provides the ability to auto create EtherChannels on ports connected to a switch. By default, auto-LAG is disabled globally and is enabled on all port interfaces. The auto-LAG applies to a switch only when it is enabled globally.

On enabling auto-LAG globally, the following scenarios are possible:

- All port interfaces participate in creation of auto EtherChannels provided the partner port interfaces have EtherChannel configured on them. For more information, see the *"The supported auto-LAG configurations between the actor and partner devices"* table below.

- Ports that are already part of manual EtherChannels cannot participate in creation of auto EtherChannels.

- When auto-LAG is disabled on a port interface that is already a part of an auto created EtherChannel, the port interface will unbundle from the auto EtherChannel.

The following table shows the supported auto-LAG configurations between the actor and partner devices:

Table 36: The supported auto-LAG configurations between the actor and partner devices

Actor/Partner	Active	Passive	Auto
Active	Yes	Yes	Yes
Passive	Yes	No	Yes
Auto	Yes	Yes	Yes

On disabling auto-LAG globally, all auto created Etherchannels become manual EtherChannels.

You cannot add any configurations in an existing auto created EtherChannel. To add, you should first convert it into a manual EtherChannel by executing the **port-channel**<*channel-number*>**persistent**.

Note Auto-LAG uses the LACP protocol to create auto EtherChannel. Only one EtherChannel can be automatically created with the unique partner devices.

Auto-LAG Configuration Guidelines

Follow these guidelines when configuring the auto-LAG feature.

- When auto-LAG is enabled globally and on the port interface , and if you do not want the port interface to become a member of the auto EtherChannel, disable the auto-LAG on the port interface.

- A port interface will not bundle to an auto EtherChannel when it is already a member of a manual EtherChannel. To allow it to bundle with the auto EtherChannel, first unbundle the manual EtherChannel on the port interface.

- When auto-LAG is enabled and auto EtherChannel is created, you can create multiple EtherChannels manually with the same partner device. But by default, the port tries to create auto EtherChannel with the partner device.

- The auto-LAG is supported only on Layer 2 EtherChannel. It is not supported on Layer 3 interface and Layer 3 EtherChannel.

How to Configure EtherChannels

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.

If you enabled PAgP on a port in the **auto** or **desirable** mode, you must reconfigure it for either the **on** mode or the LACP mode before adding this port to a cross-stack EtherChannel. PAgP does not support cross-stack EtherChannels.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode** {**access** | **trunk**}
4. **switchport access vlan** *vlan-id*
5. **channel-group** *channel-group-number* **mode** {**auto** [**non-silent**] | **desirable** [**non-silent**] | **on** } | { **active** | **passive**}
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:**	Enters global configuration mode.

	Command or Action	Purpose
	Switch# **configure terminal**	
Step 2	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet2/0/1**	Specifies a physical port, and enters interface configuration mode. Valid interfaces are 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.
Step 3	**switchport mode** {**access** | **trunk**} **Example:** Switch(config-if)# **switchport mode access**	Assigns all ports as static-access ports in the same VLAN, or configure them as trunks. If you configure the port as a static-access port, assign it to only one VLAN. The range is 1 to 4094.
Step 4	**switchport access vlan** *vlan-id* **Example:** Switch(config-if)# **switchport access vlan 22**	(Optional) If you configure the port as a static-access port, assign it to only one VLAN. The range is 1 to 4094.
Step 5	**channel-group** *channel-group-number* **mode** {**auto** [**non-silent**] | **desirable** [**non-silent**] | **on** } | { **active** | **passive**} **Example:** Switch(config-if)# **channel-group 5 mode auto**	Assigns the port to a channel group, and specifies the PAgP or the LACP mode. For *channel-group-number*, the range is 1 to 48. 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.This keyword is not supported when EtherChannel members are from different switches in the switch stack. • **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. This keyword is not supported when EtherChannel members are from different switches in the switch stack. • **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.

	Command or Action	**Purpose**
		• **non-silent** −(Optional) If your switch is connected to a partner that is PAgP-capable, configures the switch port for nonsilent operation when the port is in the **auto** or **desirable** mode. If you do not specify **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.
Step 6	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Configuring Layer 3 EtherChannels

To configure Layer 3 EtherChannels, you create the port-channel logical interface and then put the Ethernet ports into the port channel as described in the next two sections.

Creating Port-Channel Logical Interfaces

When configuring Layer 3 EtherChannels, you should first manually create the port-channel logical interface by using the **interface port-channel** global configuration command. Then put the logical interface into the channel group by using the **channel-group** interface configuration command.

> **Note** To move an IP address from a physical port to an EtherChannel, you must delete the IP address from the physical port before configuring it on the port-channel interface.

Follow these steps to create a port-channel interface for a Layer 3 EtherChannel. This procedure is required.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface port-channel** *port-channel-number*
4. **no switchport**

5. **ip address** *ip-address mask*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface port-channel** *port-channel-number* **Example:** Switch(config)# **interface port-channel 5**	Specifies the port-channel logical interface, and enters interface configuration mode. For *port-channel-number*, the range is 1 to 48.
Step 4	**no switchport** **Example:** Switch(config-if)# **no switchport**	Puts the interface into Layer 3 mode.
Step 5	**ip address** *ip-address mask* **Example:** Switch(config-if)# **ip address ip address 172.10.20.10 255.255.255.0**	Assigns an IP address and subnet mask to the EtherChannel.
Step 6	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.

This example shows how to create the logical port channel 5 and assign 172.10.20.10 as its IP address:

```
Switch# configure terminal
Switch(config)# interface port-channel 5
Switch(config-if)# no switchport
Switch(config-if)# ip address 172.10.20.10 255.255.255.0
Switch(config-if)# end
```

Configuring the Physical Interfaces

Follow these steps to assign an Ethernet port to a Layer 3 EtherChannel. This procedure is required.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **no ip address**
5. **no switchport**
6. **channel-group** *channel-group-number* **mode** { **auto** [**non-silent**] | **desirable** [**non-silent**] | **on** } | { **active** | **passive** }
7. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet 1/0/2**	Specifies a physical port, and enters 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.
Step 4	**no ip address** **Example:** Switch(config-if)# **no ip address**	Ensures that there is no IP address assigned to the physical port.
Step 5	**no switchport** **Example:** Switch(config-if)# **no switchport**	Puts the port into Layer 3 mode.
Step 6	**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.

	Command or Action	Purpose
	Example: Switch(config-if)# **channel-group 5 mode auto**	For *channel-group-number*, the range is 1 to 48. This number must be that of a previously created port channel (logical port). 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. This keyword is not supported when EtherChannel members are from different switches in the switch stack. • **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. This keyword is not supported when EtherChannel members are from different switches in the switch stack. • **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, configures the switch port for nonsilent operation when the port is in the **auto** or **desirable** mode. If you do not specify **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.
Step 7	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.

Configuring EtherChannel Load-Balancing

You can configure EtherChannel load-balancing by using source-based or destination-based forwarding methods.

This task is optional.

SUMMARY STEPS

    1. **configure terminal**

    2. **port-channel load-balance** { **dst-ip** | **dst-mac** | **src-dst-ip** | **src-dst-mac** | **src-ip** | **src-mac** }

    3. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**port-channel load-balance** { **dst-ip** | **dst-mac** | **src-dst-ip** | **src-dst-mac** | **src-ip** | **src-mac** } **Example:** Switch(config)# **port-channel load-balance src-mac**	Configures an EtherChannel load-balancing method. The default is **src-mac**. Select one of these load-distribution methods: • **dst-ip**—Specifies 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.
Step 3	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Configuring Port Channel Load Deferral

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **port-channel load-defer** *seconds*
4. **interface** *type number*
5. **port-channel load-defer**
6. **end**
7. **show etherchannel** *channel-group* **port-channel**
8. **show platform pm group-masks**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# configure terminal	Enters global configuration mode.
Step 3	**port-channel load-defer** *seconds* **Example:** Switch(config)# port-channel load-defer 60	Configures the port load share deferral interval for all port channels. • *seconds*—The time interval during which load sharing is initially 0 for deferred port channels. The range is 1 to 1800 seconds; the default is 120 seconds
Step 4	**interface** *type number* **Example:** Switch(config)# interface port-channel 10	Configures a port channel interface and enters interface configuration mode.
Step 5	**port-channel load-defer** **Example:** Switch(config-if)# port-channel load-defer	Enables port load share deferral on the port channel.
Step 6	**end** **Example:** Switch(config-if)# end	Exits interface configuration mode and returns to privileged EXEC mode.
Step 7	**show etherchannel** *channel-group* **port-channel** **Example:** Switch# show etherchannel 1 port-channel	Displays port channel information.

	Command or Action	Purpose
Step 8	**show platform pm group-masks** **Example:** `Switch# show platform pm group-masks`	Display EtherChannel group masks information.

Example

The following is sample output from the **show etherchannel** *channel-group* **port-channel** command. If the *channel-group* argument is not specified; the command displays information about all channel groups are displayed.

```
Switch# show etherchannel 1 port-channel

  Port-channels in the group:
  --------------------------

Port-channel: Po1
------------

Age of the Port-channel   = 0d:00h:37m:08s
Logical slot/port   = 9/1          Number of ports = 0
GC                  = 0x00000000     HotStandBy port = null
Port state          = Port-channel Ag-Not-Inuse
Protocol            =    -
Port security       = Disabled
Load share deferral = Enabled   defer period = 120 sec   time left = 0 sec
```

The following is sample output from the **show platform pm group-masks** command. Deferred ports have the group mask of 0xFFFF, when the defer timer is running.

```
Switch# show platform pm group-masks

==================================================================
                   Etherchannel members and group masks table
Group #ports group frame-dist slot port mask interface index
------------------------------------------------------------------
 1    0      1      src-mac
 2    0      2      src-mac
 3    0      3      src-mac
 4    0      4      src-mac
 5    0      5      src-mac
 6    0      6      src-mac
 7    0      7      src-mac
 8    0      8      src-mac
 9    0      9      src-mac
10    3      10     src-mac
                             1    12   0000 Gi1/0/12  3
                             1    10   FFFF Gi1/0/10  6
                             1    11   FFFF Gi1/0/11  7
11    0      11     src-mac
12    0      12     src-mac
13    0      13     src-mac
14    0      14     src-mac
15    0      15     src-mac
```

Configuring the PAgP Learn Method and Priority

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **pagp learn-method physical-port**
4. **pagp port-priority** *priority*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet 1/0/2**	Specifies the port for transmission, and enters interface configuration mode.
Step 3	**pagp learn-method physical-port** **Example:** Switch(config-if)# **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. Selects **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**. The learning method must be configured the same at both ends of the link.
Step 4	**pagp port-priority** *priority* **Example:** Switch(config-if)# **pagp port-priority 200**	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.
Step 5	**end** **Example:**	Returns to privileged EXEC mode.

Command or Action	Purpose
`Switch(config-if)# end`	

Configuring 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.

Configuring the LACP System Priority

You can configure the system priority for all the EtherChannels that are enabled for LACP by using the **lacp system-priority** global configuration command. You cannot configure a system priority for each LACP-configured channel. By changing this value from the default, you can affect how the software selects active and standby links.

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).

Follow these steps to configure the LACP system priority. This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **lacp system-priority** *priority*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**lacp system-priority** *priority* **Example:** Switch(config)# **lacp system-priority 32000**	Configures the LACP system priority. The range is 1 to 65535. The default is 32768. The lower the value, the higher the system priority.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Configuring the LACP Port Priority

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).

Note   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.

Follow these steps to configure the LACP port priority. This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **lacp port-priority** *priority*

5. end

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet 1/0/2**	Specifies the port to be configured, and enters interface configuration mode.
Step 4	**lacp port-priority** *priority* **Example:** Switch(config-if)# **lacp port-priority 32000**	Configures the LACP port 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.
Step 5	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.

Configuring the Port Channel Min-Links Feature

You can specify the minimum number of active ports that must be in the link-up state and bundled in an EtherChannel for the port channel interface to transition to the link-up state.

Beginning in privileged EXEC mode, follow these steps to configure the minimum number of links that are required for a port channel. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface port-channel** *channel-number*
3. **port-channel min-links** *min-links-number*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface port-channel** *channel-number* Example: Switch(config)# **interface port-channel 2**	Enters interface configuration mode for a port-channel. For *channel-number*, the range is 1 to 128.
Step 3	**port-channel min-links** *min-links-number* Example: Switch(config-if)# **port-channel min-links 3**	Specifies the minimum number of member ports that must be in the link-up state and bundled in the EtherChannel for the port channel interface to transition to the link-up state. For *min-links-number* , the range is 2 to 8.
Step 4	**end** Example: Switch(config)# **end**	Returns to privileged EXEC mode.

Configuring LACP Fast Rate Timer

You can change the LACP timer rate to modify the duration of the LACP timeout. Use the **lacp rate** command to set the rate at which LACP control packets are received by an LACP-supported interface. You can change the timeout rate from the default rate (30 seconds) to the fast rate (1 second). This command is supported only on LACP-enabled interfaces.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** {**fastethernet** | **gigabitethernet** | **tengigabitethernet**} *slot/port*
4. **lacp rate** {**normal** | **fast**}
5. **end**
6. **show lacp internal**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** Example:	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	**Purpose**
	Switch> **enable**	
Step 2	**configure terminal**	Enters global configuration mode.
	Example:	
	Switch# **configure terminal**	
Step 3	**interface** {**fastethernet** | **gigabitethernet** | **tengigabitethernet**} *slot/port*	Configures an interface and enters interface configuration mode.
	Example:	
	Switch(config)# **interface gigabitEthernet 2/1**	
Step 4	**lacp rate** {**normal** | **fast**}	Configures the rate at which LACP control packets are received by an LACP-supported interface.
	Example:	• To reset the timeout rate to its default, use the **no lacp rate** command.
	Switch(config-if)# **lacp rate fast**	
Step 5	**end**	Returns to privileged EXEC mode.
	Example:	
	Switch(config)# **end**	
Step 6	**show lacp internal**	Verifies your configuration.
	Example:	
	Switch# **show lacp internal** Switch# **show lacp counters**	

Configuring Auto-LAG Globally

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. [**no**] **port-channel auto**
4. **end**
5. **show etherchannel auto**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	[**no**] **port-channel auto** **Example:** Switch(config)# **port-channel auto**	Enables the auto-LAG feature on a switch globally. Use the no form of this command to disable the auto-LAG feature on the switch globally. **Note** By default, the auto-LAG feature is enabled on the port.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show etherchannel auto** **Example:** Switch# **show etherchannel auto**	Displays that EtherChannel is created automatically.

Configuring Auto-LAG on a Port Interface

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. [**no**] **channel-group auto**
5. **end**
6. **show etherchannel auto**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
	Switch> **enable**	
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet 1/0/1**	Specifies the port interface to be enabled for auto-LAG, and enters interface configuration mode.
Step 4	[**no**] **channel-group auto** **Example:** Switch(config-if)# **channel-group auto**	(Optional) Enables auto-LAG feature on individual port interface. Use the no form of this command to disable the auto-LAG feature on individual port interface. **Note** By default, the auto-LAG feature is enabled on the port.
Step 5	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 6	**show etherchannel auto** **Example:** Switch# **show etherchannel auto**	Displays that EtherChannel is created automatically.

What to do next

Configuring Persistence with Auto-LAG

You use the persistence command to convert the auto created EtherChannel into a manual one and allow you to add configuration on the existing EtherChannel.

SUMMARY STEPS

1. **enable**
2. **port-channel** *channel-number* **persistent**
3. **show etherchannel summary**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Switch> **enable**	• Enter your password if prompted.
Step 2	**port-channel** *channel-number* **persistent** **Example:** Switch# **port-channel 1 persistent**	Converts the auto created EtherChannel into a manual one and allows you to add configuration on the EtherChannel.
Step 3	**show etherchannel summary** **Example:** Switch# **show etherchannel summary**	Displays the EtherChannel information.

Monitoring EtherChannel, PAgP, and LACP Status

You can display EtherChannel, PAgP, and LACP status using the commands listed in this table.

Table 37: Commands for Monitoring EtherChannel, PAgP, and LACP Status

Command	Description
clear lacp { *channel-group-number* **counters** | **counters** }	Clears LACP channel-group information and traffic counters.
clear pagp { *channel-group-number* **counters** | **counters** }	Clears PAgP channel-group information and traffic counters.
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** | **sys-id**}	Displays LACP information such as traffic information, the internal LACP configuration, and neighbor information.
show running-config	Verifies your configuration entries.
show etherchannel load-balance	Displays the load balance or frame distribution scheme among ports in the port channel.

Configuration Examples for Configuring EtherChannels

Configuring Layer 2 EtherChannels: Examples

This example shows how to configure an EtherChannel on a single switch in the stack. It assigns two ports as static-access ports in VLAN 10 to channel 5 with the PAgP mode **desirable**:

```
Switch# configure terminal
Switch(config)# interface range gigabitethernet2/0/1 -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 on a single switch in the stack. It assigns two ports as static-access ports in VLAN 10 to channel 5 with the LACP mode **active**:

```
Switch# configure terminal
Switch(config)# interface range gigabitethernet2/0/1 -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
```

This example shows how to configure a cross-stack EtherChannel. It uses LACP passive mode and assigns two ports on stack member 1 and one port on stack member 2 as static-access ports in VLAN 10 to channel 5:

```
Switch# configure terminal
Switch(config)# interface range gigabitethernet2/0/4 -5
Switch(config-if-range)# switchport mode access
Switch(config-if-range)# switchport access vlan 10
Switch(config-if-range)# channel-group 5 mode passive
Switch(config-if-range)# exit
Switch(config)# interface gigabitethernet3/0/3
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 10
Switch(config-if)# channel-group 5 mode passive
Switch(config-if)# exit
```

Configuring Port-Channel Logical Interfaces: Example

This example shows how to create the logical port channel 5 and assign 172.10.20.10 as its IP address:

```
Switch# configure terminal
Switch(config)# interface port-channel 5
Switch(config-if)# no switchport
Switch(config-if)# ip address 172.10.20.10 255.255.255.0
Switch(config-if)# end
```

Configuring EtherChannel Physical Interfaces: Examples

This example shows how to configure an EtherChannel. It assigns two ports to channel 5 with the LACP mode **active**:

```
Switch# configure terminal
Switch(config)# interface range gigabitethernet2/0/1 -2
Switch(config-if-range)# no ip address
Switch(config-if-range)# no switchport
Switch(config-if-range)# channel-group 5 mode active
Switch(config-if-range)# end
```

This example shows how to configure a cross-stack EtherChannel. It assigns two ports on stack member 2 and one port on stack member 3 to channel 7 using LACP active mode:

```
Switch# configure terminal
Switch(config)# interface range gigabitethernet2/0/4 -5
Switch(config-if-range)# no ip address
Switch(config-if-range)# no switchport
Switch(config-if-range)# channel-group 7 mode active
Switch(config-if-range)# exit
Switch(config)# interface gigabitethernet3/0/3
Switch(config-if)# no ip address
Switch(config-if)# no switchport
Switch(config-if)# channel-group 7 mode active
Switch(config-if)# exit
```

Example: Configuring Port Channel Load Deferral

```
Switch# configure terminal
Switch(config)# port-channel load-defer 60
Switch(config)# interface port-channel 10
Switch(config-if)# port-channel load-defer
Switch(config-if)# end
```

Configuring Auto LAG: Examples

This example shows how to configure Auto-LAG on a switch

```
switch> enable
switch# configure terminal
switch (config)# port-channel auto
switch (config-if)# end
switch#  show etherchannel auto
```

The following example shows the summary of EtherChannel that was created automatically.

```
switch# show etherchannel auto
Flags:  D - down        P - bundled in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3       S - Layer2
        U - in use       f - failed to allocate aggregator
        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port
        A - formed by Auto LAG


Number of channel-groups in use: 1
Number of aggregators:          1

Group  Port-channel  Protocol    Ports
------+------------+-----------+-----------------------------------------------
1      Po1(SUA)      LACP       Gi1/0/45(P) Gi2/0/21(P) Gi3/0/21(P)
```

The following example shows the summary of auto EtherChannel after executing the **port-channel** 1 **persistent** command.

```
switch# port-channel 1 persistent


switch# show etherchannel summary
Switch# show etherchannel summary
Flags:  D - down        P - bundled in port-channel
        I - stand-alone s - suspended
        H - Hot-standby (LACP only)
        R - Layer3       S - Layer2
        U - in use       f - failed to allocate aggregator
        M - not in use, minimum links not met
        u - unsuitable for bundling
        w - waiting to be aggregated
        d - default port
        A - formed by Auto LAG


Number of channel-groups in use: 1
Number of aggregators:          1

Group  Port-channel  Protocol    Ports
------+------------+-----------+-----------------------------------------------
1      Po1(SU)       LACP       Gi1/0/45(P) Gi2/0/21(P) Gi3/0/21(P)
```

Configuring LACP Port Channel Min-Links: Examples

This example shows how to configure LACP port-channel min-links:

```
switch > enable
switch# configure terminal
switch(config)# interface port-channel 5
switch(config-if)# port-channel min-links 3
switch#   show etherchannel 25 summary
switch# end
```

When the minimum links requirement is not met in standalone switches, the port-channel is flagged and assigned SM/SN or RM/RN state.

```
 switch# show etherchannel 5 summary

Flags: D - down P - bundled in port-channel
I - stand-alone s - suspended
H - Hot-standby (LACP only)
```

```
R - Layer3 S - Layer2
U - in use N- not in use, no aggregation
f - failed to allocate aggregator
M - not in use, no aggregation due to minimum links not met
m- not in use, port not aggregated due to minimum links not met
u - unsuitable for bundling
w - waiting to be aggregated
d - default port
Number of channel-groups in use: 6
Number of aggregators: 6


 Group   Port-channel   Protocol    Ports
------+------------+----------+------------------------------------------------
  6      Po25(RM)       LACP        Gi1/3/1(D) Gi1/3/2(D) Gi2/2/25(D) Gi2/2/26(W)
```

Example: Configuring LACP Fast Rate Timer

This example shows you how to configure the LACP rate:

```
switch> enable
switch# configure terminal
switch(config)# interface gigabitEthernet 2/1
switch(config-if)# lacp rate fast
switch(config-if)# exit
switch(config)# end
switch# show lacp internal
switch# show lacp counters
```

The following is sample output from the **show lacp internal** command:

```
switch# show lacp internal
Flags: S - Device is requesting Slow LACPDUs
F - Device is requesting Fast LACPDUs
A - Device is in Active mode P - Device is in Passive mode
Channel group 25
LACP port Admin Oper Port Port
Port Flags State Priority Key Key Number State
Te1/49 FA bndl 32768 0x19 0x19 0x32 0x3F
Te1/50 FA bndl 32768 0x19 0x19 0x33 0x3F
Te1/51 FA bndl 32768 0x19 0x19 0x34 0x3F
Te1/52 FA bndl 32768 0x19 0x19 0x35 0x3F
```

The following is sample output from the **show lacp counters** command:

```
switch# show lacp counters

LACPDUs Marker Marker Response LACPDUs
Port Sent Recv Sent Recv Sent Recv Pkts Err
------------------------------------------------------------------
Channel group: 24
Te1/1/27 2 2 0 0 0 0 0
Te2/1/25 2 2 0 0 0 0 0
```

Additional References for EtherChannels

Related Documents

Related Topic	Document Title
Layer 2 command reference	*Catalyst 2960-XR Switch Layer 2 Command Reference*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for EtherChannels

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Configuring Link-State Tracking

Restrictions for Configuring Link-State Tracking

- You can configure only two link-state groups per switch.

- An interface cannot be a member of more than one link-state group.

- An interface that is defined as an upstream interface in a link-state group cannot also be defined as a downstream interface in the link-state group.

- Do not enable link-state tracking on individual interfaces that will part of a downstream EtherChannel interface.

Understanding Link-State Tracking

Link-state tracking, also known as trunk failover, binds the link state of multiple interfaces. Link-state tracking can be with server NIC adapter teaming to provide redundancy in the network. When the server NIC adapters are configured in a primary or secondary relationship, and the link is lost on the primary interface, network connectivity is transparently changed to the secondary interface.

Note An interface can be an aggregation of ports (an EtherChannel) or a single physical port in either access or trunk mode .

The configuration in this figure ensures that the network traffic flow is balanced.

Figure 40: Typical Link-State Tracking Configuration
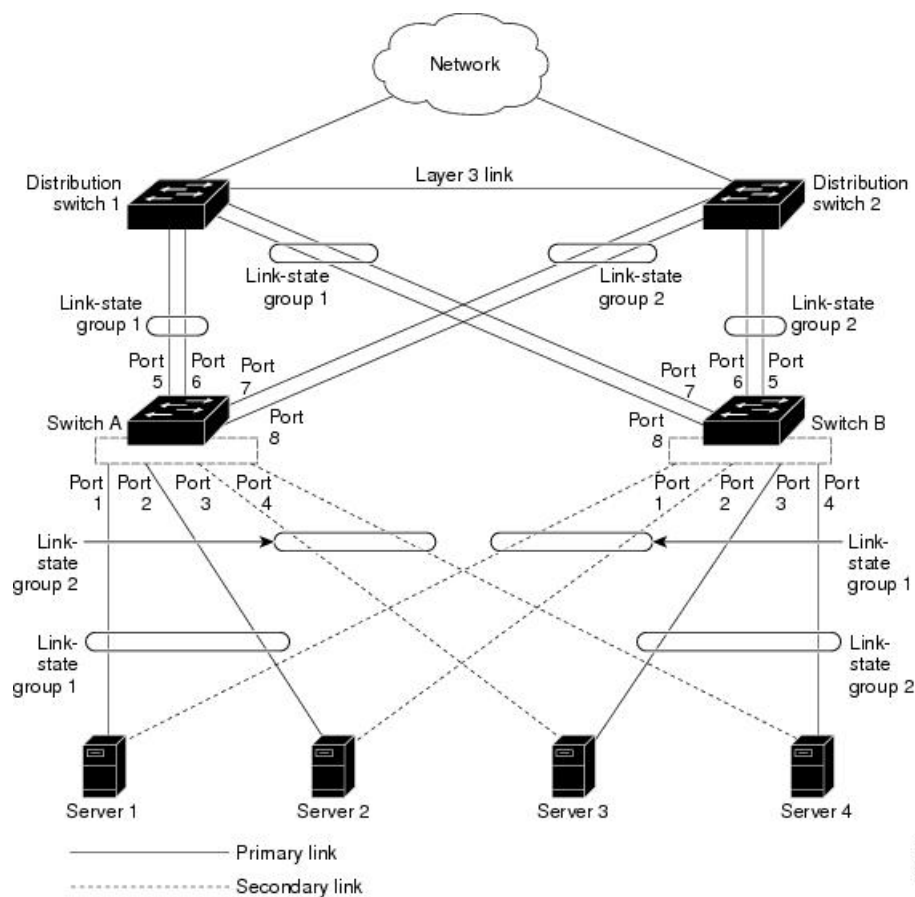
- For links to switches and other network devices

    - 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. For example, in the previous figure, 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.

How to Configure Link-State Tracking

To enable link-state tracking, create a link-state group and specify the interfaces that are assigned to the group. This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **link state track** *number*
3. **interface** *interface-id*
4. **link state group** [*number*]{**upstream** | **downstream**}
5. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**link state track** *number* **Example:** Switch(config)# **link state track 2**	Creates a link-state group and enables link-state tracking. The group number can be 1 or 2; the default is 1.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet2/0/1**	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) or routed ports. **Note** Do not enable link-state tracking on individual interfaces that will be part of an Etherchannel interface.
Step 4	**link state group** [*number*]{**upstream** | **downstream**} **Example:** Switch(config-if)# **link state group 2 upstream**	Specifies a link-state group and configures the interface as either an upstream or downstream interface in the group.
Step 5	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.

Monitoring Link-State Tracking

You can display link-state tracking status using the command in this table.

Table 38: Commands for Monitoring Link-State Tracking Status

Command	**Description**
show link state group [*number*] [**detail**]	Displays the link-state group information.

Configuring Link-State Tracking: Example

This example shows how to create the link-state group 1 and configure the interfaces in the link-state group.

```
Switch# configure terminal
Switch(config)# link state track 1
Switch(config-if)# interface range gigabitethernet1/0/21-22
Switch(config-if)# link state group 1 upstream
Switch(config-if)# interface gigabitethernet1/0/1
Switch(config-if)# link state group 1 downstream
Switch(config-if)# interface gigabitethernet1/0/3
Switch(config-if)# link state group 1 downstream
Switch(config-if)# interface gigabitethernet1/0/5
Switch(config-if)# link state group 1 downstream
Switch(config-if)# end
```

Additional References for Link-State Tracking

Related Documents

Related Topic	Document Title
Layer 2 command reference	*Catalyst 2960-XR Switch Layer 2 Command Reference*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for Link-State Tracking

Releases	Feature Information
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Configuring Flex Links and the MAC Address-Table Move Update Feature

Restrictions for Configuring Flex Links and MAC Address-Table Move Update

- Flex Links are supported only on Layer 2 ports and port channels.

- You can configure up to 16 backup links.

- You can configure only one Flex Links backup link for any active link, and it must be a different interface from the active interface.

- An interface can belong to only one Flex Links pair. An interface can be a backup link for only one active link. An active link cannot belong to another Flex Links 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 Flex Links, and you can configure a port channel and a physical interface as Flex Links, 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 Flex Links 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 Flex Links ports. A Flex Links 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.

Information About Flex Links and MAC Address-Table Move Update

Flex Links

Flex Links 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. Flex Links 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, Flex Links are not necessary because STP already provides link-level redundancy or backup.

You configure Flex Links on one Layer 2 interface (the active link) by assigning another Layer 2 interface as the Flex Links or backup link. On switches, the Flex Links can be on the same switch or on another switch in the stack. 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 Flex Links interfaces.

Flex Links Configuration

In the following figure, ports 1 and 2 on switch A are connected to uplink switches B and C. Because they are configured as Flex Links, 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 configure a preemption function, specifying the preferred port for forwarding traffic. For example, you can configure the Flex Links 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 **switchport backup interface preemption mode bandwidth** and **switchport backup interface preemption delay** interface configuration commands.

Figure 41: Flex Links Configuration Example
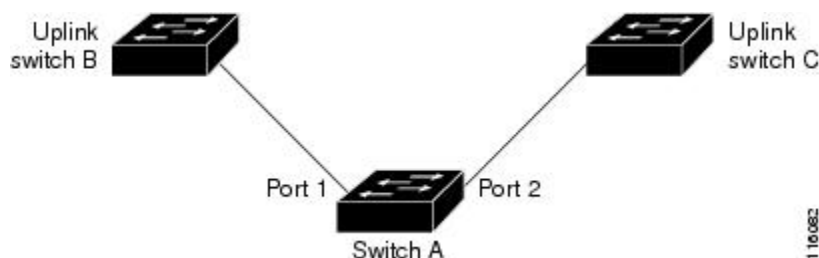
If a primary (forwarding) link goes down, a trap notifies the network management stations. If the standby link goes down, a trap notifies the users.

Flex Links are supported only on Layer 2 ports and port channels, not on VLANs or on Layer 3 ports.

VLAN Flex Links Load Balancing and Support

VLAN Flex Links load balancing allows users to configure a Flex Links pair so that both ports simultaneously forward the traffic for some mutually exclusive VLANs. For example, if Flex Links ports are configured for 1 to 100 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. In addition to providing the redundancy, this Flex Links pair can be used for load balancing. Flex Links VLAN load balancing does not impose any restrictions on uplink switches.

Figure 42: VLAN Flex Links Load-Balancing Configuration Example

The following figure displays a VLAN Flex Links load-balancing configuration.
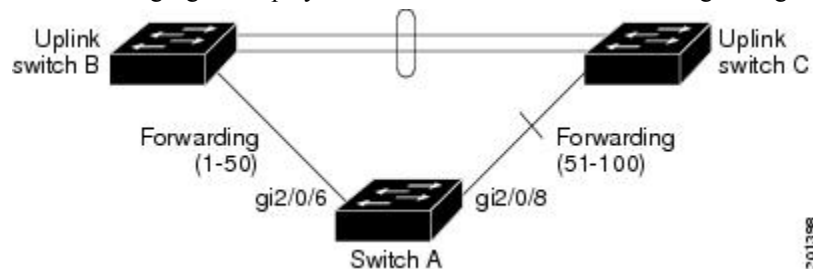
Multicast Fast Convergence with Flex Links Failover

Multicast fast convergence reduces the multicast traffic convergence time after a Flex Links failure. Multicast fast convergence is implemented by a combination of learning the backup link as an mrouter port, generating IGMP reports, and leaking IGMP reports.

Learning the Other Flex Links 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 Flex Links ports receiving queries. Flex Links 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 Flex Links port. The other Flex Links port is then learned as the mrouter port. After changeover, multicast traffic then flows through the other Flex Links port. To achieve faster convergence of traffic, both Flex Links ports are learned as mrouter ports whenever either Flex Links port is learned as the mrouter port. Both Flex Links ports are always part of multicast groups.

Although both Flex Links ports are part of the groups in normal operation mode, all traffic on the backup port is blocked. 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 Flex Links 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 Flex Links active link goes down. This can be achieved by leaking only IGMP report packets on the Flex Links 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 Flex Links 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.

Figure 43: MAC Address-Table Move Update Example

In the following figure, switch A is an access switch, and ports 1 and 2 on switch A are connected to uplink switches B and D through a Flex Links 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.
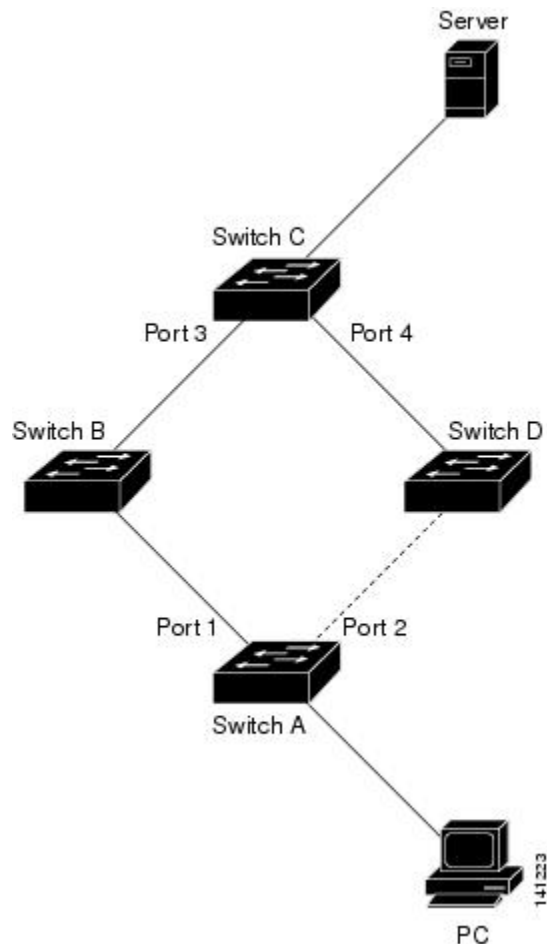
If the MAC address-table move update feature is not configured and port 1 goes down, port 2 starts forwarding traffic. However, for a short time, switch C keeps forwarding traffic from the server to the PC through port 3, and the PC does not get the traffic because port 1 is down. If switch C removes 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, 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 less than 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.

Flex Links VLAN Load Balancing Configuration Guidelines

- For Flex Links 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 Flex Links pair.

MAC Address-Table Move Update Configuration Guidelines

- 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 *get* the MAC address-table move updates.

Default Flex Links and MAC Address-Table Move Update Configuration

- Flex Links is not configured, and there are no backup interfaces defined.

- The preemption mode is off.

- The preemption delay is 35 seconds.

- The MAC address-table move update feature is not configured on the switch.

How to Configure Flex Links and the MAC Address-Table Move Update Feature

Configuring Flex Links

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport backup interface** *interface-id*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.

	Command or Action	**Purpose**
Step 2	**interface** *interface-id* **Example:** Switch(conf)# **interface gigabitethernet1/0/1**	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 48.
Step 3	**switchport backup interface** *interface-id* **Example:** Switch(conf-if)# **switchport backup interface gigabitethernet1/0/2**	Configures a physical Layer 2 interface (or port channel) as part of a Flex Links pair with the interface. When one link is forwarding traffic, the other interface is in standby mode.
Step 4	**end** **Example:** Switch(conf-if)# **end**	Returns to privileged EXEC mode.

Configuring a Preemption Scheme for a Pair of Flex Links

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport backup interface** *interface-id*
4. **switchport backup interface** *interface-id* **preemption mode** [**forced** | **bandwidth** | **off**]
5. **switchport backup interface** *interface-id* **preemption delay** *delay-time*
6. **end**
7. **show interface** [*interface-id*] **switchport backup**
8. **copy running-config startup config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode
Step 2	**interface** *interface-id* **Example:** Switch(conf)# **interface gigabitethernet1/0/1**	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 48.

	Command or Action	Purpose
Step 3	**switchport backup interface** *interface-id* **Example:** Switch(conf-if)# **switchport backup interface gigabitethernet1/0/2**	Configures a physical Layer 2 interface (or port channel) as part of a Flex Links pair with the interface. When one link is forwarding traffic, the other interface is in standby mode.
Step 4	**switchport backup interface** *interface-id* **preemption mode** [**forced** | **bandwidth** | **off**] **Example:** Switch(conf-if)# **switchport backup interface gigabitethernet1/0/2 preemption mode forced**	Configures a preemption mechanism and delay for a Flex Links interface pair. You can configure the preemption as: • **forced**—(Optional) The active interface always preempts the backup. • **bandwidth**—(Optional) The interface with the higher bandwidth always acts as the active interface. • **off**—(Optional) No preemption occurs from active to backup.
Step 5	**switchport backup interface** *interface-id* **preemption delay** *delay-time* **Example:** Switch(conf-if)# **switchport backup interface gigabitethernet1/0/2 preemption delay 50**	Configures the time delay until a port preempts another port. **Note** Setting a delay time only works with forced and bandwidth modes.
Step 6	**end** **Example:** Switch(conf-if)# **end**	Returns to privileged EXEC mode.
Step 7	**show interface** [*interface-id*] **switchport backup** **Example:** Switch# **show interface gigabitethernet1/0/2 switchport backup**	Verifies the configuration.
Step 8	**copy running-config startup config** **Example:** Switch# **copy running-config startup config**	(Optional) Saves your entries in the switch startup configuration file.

Configuring VLAN Load Balancing on Flex Links

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport backup interface** *interface-id* **prefer vlan** *vlan-range*
4. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** Switch (config)# **interface gigabitethernet2/0/6**	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 48.
Step 3	**switchport backup interface** *interface-id* **prefer vlan** *vlan-range* **Example:** Switch (config-if)# **switchport backup interface gigabitethernet2/0/8 prefer vlan 2**	Configures a physical Layer 2 interface (or port channel) as part of a Flex Links pair with the interface and specifies the VLANs carried on the interface. The VLAN ID range is 1 to 4094.
Step 4	**end** **Example:** Switch (config-if)# **end**	Returns to privileged EXEC mode.

Configuring MAC Address-Table Move Update

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. Use one of the following:

   - **switchport backup interface** *interface-id*
   - **switchport backup interface** *interface-id* **mmu primary vlan** *vlan-id*

4. **end**
5. **mac address-table move update transmit**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# `configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** Switch#`interface gigabitethernet1/0/1`	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 48.
Step 3	Use one of the following: **switchport backup interface** *interface-id* **switchport backup interface** *interface-id* **mmu primary vlan** *vlan-id* **Example:** Switch(config-if)# `switchport backup interface gigabitethernet0/2 mmu primary vlan 2`	Configures a physical Layer 2 interface (or port channel), as part of a Flex Links pair with the interface. The MAC address-table move update VLAN is the lowest VLAN ID on the interface. Configure 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.
Step 4	**end** **Example:** Switch(config-if)# `end`	Returns to global configuration mode.

Step 5	**mac address-table move update transmit**
	
	Example:
	
	Switch(config)#
	`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.

Step 6	**end** **Example:** Switch(config)# `end`	Returns to privileged EXEC mode.

Configuring a Switch to Obtain and Process MAC Address-Table Move Update Messages

SUMMARY STEPS

1. **configure terminal**
2. **mac address-table move update receive**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode
Step 2	**mac address-table move update receive** **Example:** Switch (config)# **mac address-table move update receive**	Enables the switch to obtain and processes the MAC address-table move updates.
Step 3	**end** **Example:** Switch (config)# **end**	Returns to privileged EXEC mode.

Monitoring Flex Links, Multicast Fast Convergence, and MAC Address-Table Move Update

Command	Purpose
show interface [*interface-id*] **switchport backup**	Displays the Flex Links backup interface configured for an interface or all the configured Flex Links and the state of each active and backup interface (up or standby mode).
show ip igmp profile address-table move update *profile-id*	Displays the specified IGMP profile or all the IGMP profiles defined on the switch.
show mac address-table move update	Displays the MAC address-table move update information on the switch.

Configuration Examples for Flex Links

Configuring Flex Links: Examples

This example shows how to verify the configuration after you configure an interface with a backup interface:

```
Switch# show interface switchport backup

Switch Backup Interface Pairs:
Active Interface Backup Interface State
------------------------------------------------------------------------
GigabitEthernet1/0/1 GigabitEthernet1/0/2 Active Up/Backup Standby
```

This example shows how to verify the configuration after you configure the preemption mode as forced for a backup interface pair:

```
Switch# show interface switchport backup detail

Switch Backup Interface Pairs:

Active Interface Backup Interface State
------------------------------------------------------------------------
GigabitEthernet1/0/211 GigabitEthernet1/0/2 Active Up/Backup Standby
Interface Pair : Gi1/0/1, Gi1/0/2
Preemption Mode : forced
Preemption Delay : 50 seconds
Bandwidth : 100000 Kbit (Gi1/0/1), 100000 Kbit (Gi1/0/2)
Mac Address Move Update Vlan : auto
```

Configuring VLAN Load Balancing on Flex Links: Examples

In the following example, VLANs 1 to 50, 60, and 100 to 120 are configured on the switch:

```
Switch(config)# interface gigabitethernet 2/0/6
Switch(config-if)# switchport backup interface gigabitethernet 2/0/8 prefer vlan 60,100-120
```

When both interfaces are up, Gi2/0/8 forwards traffic for VLANs 60 and 100 to 120 and Gi2/0/6 forwards traffic for VLANs 1 to 50.

```
Switch# show interfaces switchport backup

Switch Backup Interface Pairs:

Active Interface        Backup Interface        State
------------------------------------------------------------------------
GigabitEthernet2/0/6    GigabitEthernet2/0/8    Active Up/Backup Standby

Vlans Preferred on Active Interface: 1-50
Vlans Preferred on Backup Interface: 60, 100-120
```

When a Flex Links interface goes down (LINK_DOWN), VLANs preferred on this interface are moved to the peer interface of the Flex Links pair. In this example, if interface Gi2/0/6 goes down, Gi2/0/8 carries all VLANs of the Flex Links pair.

```
Switch# show interfaces switchport backup

Switch Backup Interface Pairs:

Active Interface        Backup Interface        State
------------------------------------------------------------------------
GigabitEthernet2/0/6    GigabitEthernet2/0/8    Active Down/Backup Up

Vlans Preferred on Active Interface: 1-50
Vlans Preferred on Backup Interface: 60, 100-120
```

When a Flex Links 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 Gi2/0/6 comes up, VLANs preferred on this interface are blocked on the peer interface Gi2/0/8 and forwarded on Gi2/0/6.

```
Switch# show interfaces switchport backup

Switch Backup Interface Pairs:

Active Interface        Backup Interface        State
------------------------------------------------------------------------
GigabitEthernet2/0/6    GigabitEthernet2/0/8    Active Up/Backup Standby

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
------------------------------------------------------------------------
FastEthernet1/0/3       FastEthernet1/0/4       Active Down/Backup Up

Vlans Preferred on Active Interface: 1-2,5-4094
Vlans Preferred on Backup Interface: 3-4
Preemption Mode  : off
Bandwidth : 10000 Kbit (Fa1/0/3), 100000 Kbit (Fa1/0/4)
Mac Address Move Update Vlan : auto
```

Configuring the MAC Address-Table Move Update: Examples

This example shows how to verify the configuration after you configure an access switch to send MAC address-table move updates:

```
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
```

Configuring Multicast Fast Convergence with Flex Links Failover: Examples

These are configuration examples for learning the other Flex Links port as the mrouter port when Flex Links is configured on GigabitEthernet1/0/11 and GigabitEthernet1/0/12, and output for the **show interfaces switchport backup** command:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface GigabitEthernet1/0/11
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport backup interface GigabitEthernet1/0/12
Switch(config-if)# exit
Switch(config)# interface GigabitEthernet1/0/12
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# end
Switch# show interfaces switchport backup detail
Switch Backup Interface Pairs:
Active Interface Backup Interface State
GigabitEthernet1/0/11 GigabitEthernet1/0/12 Active Up/Backup Standby
Preemption Mode : off
Multicast Fast Convergence : Off
Bandwidth : 100000 Kbit (Gi1/0/11), 100000 Kbit (Gi1/0/12)
Mac Address Move Update Vlan : auto
```

This output shows a querier for VLANs 1 and 401, with their queries reaching the switch through GigabitEthernet1/0/11:

```
Switch# show ip igmp snooping querier

Vlan  IP Address  IGMP Version  Port
----------------------------------------------------------
1    10.0.0.10   v2      Gi1/0/11
401  41.41.41.1  v2      Gi1/0/11
```

This example is output for the **show ip igmp snooping mrouter** command for VLANs 1 and 401:

```
Switch# show ip igmp snooping mrouter

Vlan    ports
----    -----
```

```
1     Gi1/0/11(dynamic), Gi1/0/12(dynamic)
401   Gi1/0/11(dynamic), Gi1/0/12(dynamic)
```

Similarly, both Flex Links ports are part of learned groups. In this example, GigabitEthernet2/0/11 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    Gi1/0/11, Gi1/0/12, Gi2/0/11
1   228.1.5.2   igmp   v2    Gi1/0/11, Gi1/0/12, Gi2/0/11
```

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 GigabitEthernet1/0/11, because the backup port GigabitEthernet1/0/12 is blocked. When the active link, GigabitEthernet1/0/11, goes down, the backup port, GigabitEthernet1/0/12, 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 Flex Links. This behavior changes when the user configures fast convergence using the **switchport backup interface gigabitEthernet 1/0/12 multicast fast-convergence** command. This example shows turning on this feature:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitEthernet 1/0/11
Switch(config-if)# switchport backup interface gigabitEthernet 1/0/12 multicast
fast-convergence
Switch(config-if)# exit
Switch# show interfaces switchport backup detail

Switch Backup Interface Pairs:
Active          Interface       Backup Interface State
---------------------------------------------------------------------
GigabitEthernet1/0/11  GigabitEthernet1/0/12  Active Up/Backup Standby
Preemption Mode : off
Multicast Fast Convergence : On
Bandwidth : 100000 Kbit (Gi1/0/11), 100000 Kbit (Gi1/0/12)
Mac Address Move Update Vlan : auto
```

This output shows a querier for VLAN 1 and 401 with their queries reaching the switch through GigabitEthernet1/0/11:

```
Switch# show ip igmp snooping querier

Vlan  IP Address  IGMP Version  Port
-------------------------------------------------------------
1    10.0.0.10   v2     Gi1/0/11
401  41.41.41.1  v2     Gi1/0/11
```

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    Gi1/0/11(dynamic), Gi1/0/12(dynamic)
401    Gi1/0/11(dynamic), Gi1/0/12(dynamic)
```

Similarly, both the Flex Links ports are a part of the learned groups. In this example, GigabitEthernet2/0/11 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   Gi1/0/11, Gi1/0/12, Gi2/0/11
1   228.1.5.2   igmp   v2   Gi1/0/11, Gi1/0/12, Gi2/0/11
```

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 GigabitEthernet1/0/11, it is also leaked to the backup port GigabitEthernet1/0/12. The upstream router learns the groups and starts forwarding multicast data, which is dropped at the ingress because GigabitEthernet1/0/12 is blocked. When the active link, GigabitEthernet1/0/11, goes down, the backup port, GigabitEthernet1/0/12, begins forwarding. You do not need to send any proxy reports as 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 very minimal.

Additional References for Flex Links and MAC Address-Table Move Update

Related Documents

Related Topic	Document Title
Layer 2 command reference	*Catalyst 2960-XR Switch Layer 2 Command Reference*
switchport backup interface command	*Catalyst 2960-XR Switch Interface and Hardware Component Command Reference*

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for Flex Links and MAC Address-Table Move Update

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

CHAPTER **21**

Configuring UniDirectional Link Detection

Restrictions for Configuring UDLD

The following are restrictions for configuring UniDirectional Link Detection (UDLD):

- A UDLD-capable port cannot detect a unidirectional link if it is connected to a UDLD-incapable port of another switch.
- When configuring the mode (normal or aggressive), make sure that the same mode is configured on both sides of the link.

⚠

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.

Information About 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.

Normal Mode

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.

Aggressive Mode

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

- Event-driven detection and echoing

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 operation. 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.

UDLD Reset Options

If an interface becomes disabled by UDLD, you can use one of the following options to reset UDLD:

- The **udld reset** interface configuration command.

- The **shutdown** interface configuration command followed by the **no shutdown** interface configuration command restarts the disabled port.

- The **no udld** {**aggressive** | **enable**} global configuration command followed by the **udld** {**aggressive** | **enable**} global configuration command reenables the disabled ports.

- The **no udld port** interface configuration command followed by the **udld port** [**aggressive**] interface configuration command reenables the disabled fiber-optic port.

- The **errdisable recovery cause udld** global configuration command enables the timer to automatically recover from the UDLD error-disabled state, and the **errdisable recovery interval** *interval* global configuration command specifies the time to recover from the UDLD error-disabled state.

Default UDLD Configuration

Table 39: Default UDLD Configuration

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

Feature	Default Setting
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.

SUMMARY STEPS

1. **configure terminal**
2. **udld** {**aggressive** | **enable** | **message time** *message-timer-interval*}
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**udld** {**aggressive**	**enable**	**message time** *message-timer-interval*} **Example:** Switch(config)# **udld enable message time 10**	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. • **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; the default value is 15.

	Command or Action	Purpose
		Note This command affects fiber-optic ports only. Use the **udld** interface configuration command to enable UDLD on other port types. Use the **no** form of this command, to disable UDLD.
Step 3	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Enabling UDLD on an Interface

Follow these steps either to enable UDLD in the aggressive or normal mode or to disable UDLD on a port.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **udld port** [**aggressive**]
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet 1/0/1**	Specifies the port to be enabled for UDLD, and enters interface configuration mode.
Step 3	**udld port** [**aggressive**] **Example:** Switch(config-if)# **udld port aggressive**	UDLD is disabled by default. • **udld port**—Enables UDLD in normal mode on the specified port. • **udld port aggressive**—(Optional) Enables UDLD in aggressive mode on the specified port.

	Command or Action	Purpose
		Note Use the **no udld port** interface configuration command to disable UDLD on a specified fiber-optic port.
Step 4	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.

Monitoring and Maintaining UDLD

Command	Purpose
show udld [*interface-id* | **neighbors**]	Displays the UDLD status for the specified port or for all ports.

Additional References for UDLD

Related Documents

Related Topic	Document Title
Layer 2 command reference	*Catalyst 2960-XR Switch Layer 2 Command Reference*

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for UDLD

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Configuring Resilient Ethernet Protocol

Overview of Resilient Ethernet Protocol

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.

A REP segment is a chain of ports connected to each other and configured with a segment ID. Each segment consists of standard (nonedge) segment ports and two user-configured edge ports. A device 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 Trunk Ethernet Flow Point (EFP) interfaces.

The following figure 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 the ports are operational (as in the segment on the left), a single port is blocked, as shown by the diagonal line. When there is a failure in the network, the blocked port returns to the forwarding state to minimize network disruption.

Figure 44: REP Open Segment
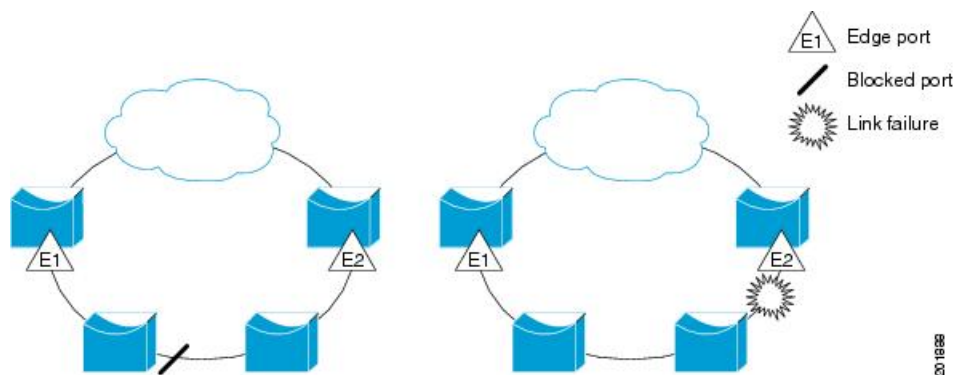
The segment shown in the figure above is an open segment; there is no connectivity between the two edge ports. The REP segment cannot cause a bridging loop, and you can safely connect the segment edges to any network. All the hosts connected to devices inside the segment have two possible connections to the rest of the network through the edge ports, but only one connection is accessible at any time. If a failure occurs on any segment or on any port on a REP segment, REP unblocks all the ports to ensure that connectivity is available through the other gateway.

The segment shown in the following figure is a ring segment, with both the edge ports located on the same device. With this configuration, you can create a redundant connection between any two devices in the segment.

Figure 45: REP Ring Segment
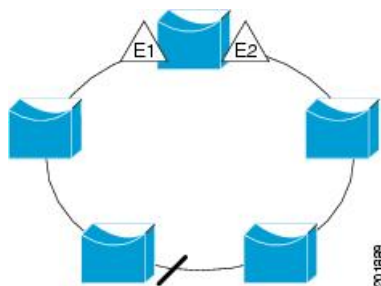
REP segments have the following characteristics:

- If all the ports in a 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, and cause a link failure, all the ports forward traffic on all the VLANs to ensure connectivity.

- In case of a link failure, alternate ports are unblocked as quickly as possible. When the failed link is 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, which is controlled by the primary edge port (any port in the segment).

In access ring-topologies, the neighboring switch might not support REP as shown in the following figure. In this scenario, you can configure the non-REP-facing ports (E1 and E2) as edge no-neighbor ports. These ports inherit all the 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 scenario, the STP topology change notice (TCN) that is sent is a multiple spanning-tree (MST) STP message.

Figure 46: Edge No-Neighbor Ports
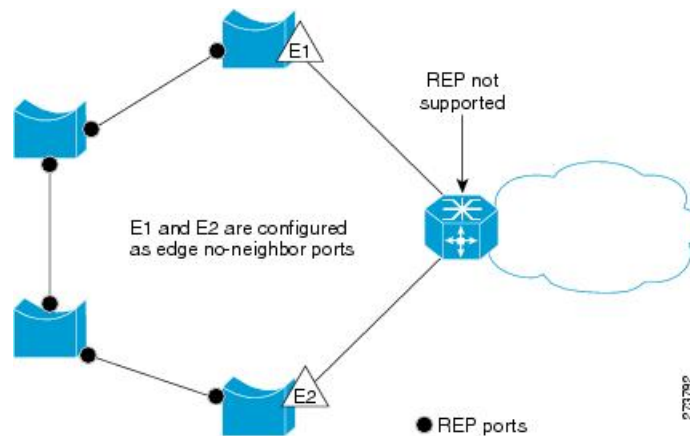
REP has these limitations:

- You must configure each segment port; an incorrect configuration might 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 function 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 the VLANs are blocked on an interface until the neighbor is detected. 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.

- A neighbor does not acknowledge a local port as a peer.

Each port creates an adjacency with its immediate neighbor. After the neighbor adjacencies are created, the ports negotiate with each other to determine the blocked port for the segment, which will function as the alternate port. All the other ports become unblocked. By default, REP packets are sent to a bridge protocol data unit-class MAC address. The packets can also be sent to a 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 the devices not running REP.

Fast Convergence

REP runs on a physical link basis and not on a per-VLAN basis. Only one hello message is required for all the VLANs, and this reduces the load on the protocol. We recommend that you create VLANs consistently on all the 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 entire 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 an administrative VLAN for the entire domain or for a particular segment.

The estimated convergence recovery time on fiber interfaces is between 50 ms and 200 ms for the local segment with 200 VLANs configured. 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; and another 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 the 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 **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.

- 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 Configure offset numbers on the primary edge port by identifying a port's downstream position from the primary (or secondary) edge port. Never enter an offset value of 1 because that is the offset number of the primary edge port.

The following figure shows neighbor offset numbers for a segment, where E1 is the primary edge port and E2 is the secondary edge port. The numbers inside the ring are numbers offset from the primary edge port; the numbers outside of the ring show the offset numbers from the secondary edge port. Note that you can identify all the 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.

Figure 47: Neighbor Offset Numbers in a Segment
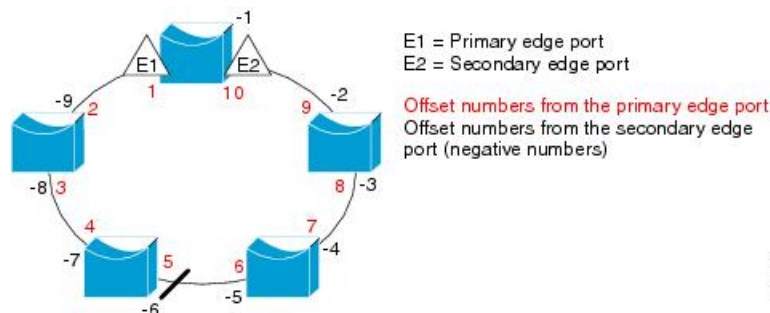
When the REP segment is complete, all the VLANs are blocked. When you configure VLAN load balancing, you must also configure triggers in one of two ways:

- 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 the interfaces in the segment about the preemption. When the secondary port receives the message, the message is sent to 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 the 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 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 the STP or the Flex Link 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 an 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. Since each segment always contains a blocked port, multiple segments means multiple blocked

ports and a potential loss of connectivity. After the segment is configured in both directions up to the location of the edge ports, configure the edge ports.

REP Ports

REP segments consist of Failed, Open, or Alternate ports:

- 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 the VLANs on the interface. Blocked-port negotiations occur, and when the segment settles, one blocked port remains in the alternate role and all the other ports become open ports.

- When a failure occurs in a link, all the ports move to the Failed state. When the Alternate port receives the failure notification, it changes to the Open state, forwarding all the 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 to 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.

How to Configure Resilient Ethernet Protocol

A segment is a collection of ports connected to one another in a chain and configured with a segment ID. To configure REP segments, 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 a segment, with one of them being the primary edge port and the other the secondary edge port by default. A segment should have only one primary edge port. If you configure two ports in a segment as primary edge ports, for example, ports on different switches, the REP selects one of them to serve as the segment's primary edge port. If required, you can configure the location to which segment topology change notices (STCNs) and VLAN load balancing are to be sent.

Default REP Configuration

REP is disabled on all the interfaces. When enabled, the interface is a regular segment port unless it is configured as an edge port.

When REP is enabled, the task of sending segment topology change notices (STCNs) is disabled, all the 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 the VLANs in 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 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 goes into a forwarding state for the data path to help maintain connectivity during configuration. In the **show rep interface** command output, the Port Role for this port is displayed as **Fail Logical Open**; the Port Role for the other failed port is displayed as **Fail No Ext Neighbor**. When the external neighbors for the failed ports are configured, the ports go through the alternate port transitions and eventually go to an open state, or remain as the alternate port, based on the alternate port selection mechanism.

- REP ports must be Layer 2 IEEE 802.1Q or Trunk ports.

- We recommend that you configure all the trunk ports in a segment with the same set of allowed VLANs.

- Be careful when configuring REP through a Telnet connection because REP blocks all the VLANs until another REP interface sends a message to unblock it. You might lose connectivity to the router if you enable REP in a Telnet session that accesses the router 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 an REP segment, be sure that the connection is at the segment edge. An STP connection that is not at the edge might cause a bridging loop because STP does not run on REP segments. All the STP BPDUs are dropped at REP interfaces.

- You must configure all the trunk ports in a segment with the same set of allowed VLANs. If this is not done, misconfiguration occurs.

- If REP is enabled on two ports on a switch, both the ports must be either regular segment ports or edge ports. 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 both be edge ports, 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 remain in a blocked state until they are safe to be unblocked. You must, therefore, be aware of the status of REP interfaces to avoid sudden connection losses.

- REP sends all the LSL PDUs in the untagged frames to the native VLAN. The BPA message sent to a Cisco multicast address is sent to the administration VLAN, which is VLAN 1 by default.

- You can configure the duration for which a REP interface remains up without receiving a hello from a neighbor. 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.

  - 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.

- REP ports cannot be configured as one of the following port types:

    - Switched Port Analyzer (SPAN) destination port

    - Tunnel port

    - Access port

- REP is supported on EtherChannels, but not on an individual port that belongs to an EtherChannel.

- There can be a maximum of 64 REP segments per switch.

Configuring REP Administrative VLAN

To avoid the delay created by link-failure messages, and VLAN-blocking notifications during load balancing, REP floods packets to a regular multicast address at the hardware flood layer (HFL). These messages are flooded to the whole network, and not just the REP segment. You can control the flooding of these messages by configuring an administrative VLAN.

Follow these guidelines when configuring the REP administrative VLAN:

- If you do not configure an administrative VLAN, the default is VLAN 1.

- You can configure one admin VLAN on the switch for all segments.

- The administrative VLAN cannot be the RSPAN VLAN.

To configure the REP administrative VLAN, follow these steps, beginning in privileged EXEC mode:

SUMMARY STEPS

1. **configure terminal**
2. **rep admin vlan** *vlan-id*
3. **end**
4. **show interface** [*interface-id*] **rep detail**
5. **copy running-config startup config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**rep admin vlan** *vlan-id* **Example:** Switch(config)# **rep admin vlan 2**	Specifies the administrative VLAN. The range is from 2 to 4094. To set the admin VLAN to 1, which is the default, enter the **no rep admin vlan** global configuration command.

	Command or Action	Purpose
Step 3	**end** **Example:** Switch(config)# **end**	Exits global configuration mode and returns to privileged EXEC mode.
Step 4	**show interface** [*interface-id*] **rep detail** **Example:** Switch# **show interface gigabitethernet1/1 rep detail**	(Optional) Verifies the configuration on a REP interface.
Step 5	**copy running-config startup config** **Example:** Switch# **copy running-config startup config**	(Optional) Saves your entries in the switch startup configuration file.

Configuring a REP Interface

To configure REP, enable REP on each segment interface and identify the segment ID. This task is mandatory, and must be done before other REP configurations. You must also configure a primary and secondary edge port on each segment. All the other steps are optional.

Follow these steps to enable and configure REP on an interface:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport mode trunk**
5. **rep segment** *segment-id* [**edge** [**no-neighbor**] [**primary**]] [**preferred**]
6. **rep stcn** {**interface** *interface id* | **segment** *id-list* | **stp**}
7. **rep block port** {**id** *port-id* | *neighbor-offset* | **preferred**} **vlan** {*vlan-list* | **all**}
8. **rep preempt delay** *seconds*
9. **rep lsl-age-timer** *value*
10. **end**
11. **show interface** [*interface-id*] **rep** [**detail**]
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.

Layer 2/3

Configuring a REP Interface

	Command or Action	**Purpose**
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch# **interface gigabitethernet1/1**	Specifies the interface, and enters interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface).
Step 4	**switchport mode trunk** **Example:** Switch# **switchport mode trunk**	Configures the interface as a Layer 2 trunk port.
Step 5	**rep segment** *segment-id* [**edge** [**no-neighbor**] [**primary**]] [**preferred**] **Example:** Switch# **rep segment 1 edge no-neighbor primary**	Enables REP on the interface and identifies a segment number. The segment ID range is from 1 to 1024. **Note** You must configure two edge ports, including one primary edge port, for each segment. These optional keywords are available: • (Optional) **edge**—Configures the port as an edge port. Each segment has only two edge ports. Entering the keyword **edge** without the keyword **primary** configures the port as the secondary edge port. • (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 the properties of an edge port, and you can configure the properties the same way you would for an 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 keyword **primary** on both the switches, the configuration is valid. 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.

Consolidated Platform Configuration Guide, Cisco IOS XE 15.2(2)E (Catalyst 2960-XR Switch)

368

	Command or Action	**Purpose**
		Note Configuring a port as preferred does not guarantee that it becomes the alternate port; it merely gives the port a slight edge over equal contenders. The alternate port is usually a previously failed port.
Step 6	**rep stcn** {**interface** *interface id* | **segment** *id-list* | **stp**} **Example:** Switch# **rep stcn segment 25-50**	(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 from 1 to 1024. • **stp**—Sends STCNs to STP networks. **Note** Spanning Tree (MST) mode is required on edge no-neighbor nodes when **rep stcn stp** command is configured for sending STCNs to STP networks.
Step 7	**rep block port** {**id** *port-id* | *neighbor-offset* | **preferred**} **vlan** {*vlan-list* | **all**} **Example:** Switch# **rep block port id 0009001818D68700 vlan 1-100**	(Optional) Configures VLAN load balancing on the primary edge port, identifies the REP alternate port in one of three ways (**id** *port-id*, *neighbor_offset*, **preferred**), and configures 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** *type number* **rep** [**detail**] privileged EXEC command. • *neighbor_offset*—Number to identify 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. Enter **-1** to identify the secondary edge port as the alternate port. **Note** Because you enter the **rep block port** command at the primary edge port (offset number 1), you cannot 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.

	Command or Action	Purpose
		• **vlan all**—Blocks all the VLANs.
		Note Enter this command only on the REP primary edge port.
Step 8	**rep preempt delay** *seconds* **Example:** Switch# **rep preempt delay 100**	(Optional) Configures a preempt time delay. • Use this command if you want VLAN load balancing to be automatically triggered after a link failure and recovery. • The time delay range is between15 to 300 seconds. The default is manual preemption with no time delay. **Note** Enter this command only on the REP primary edge port.
Step 9	**rep lsl-age-timer** *value* **Example:** Switch# **rep lsl-age-timer 2000**	(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** • EtherChannel port channel interfaces do not support LSL age-timer values that are less than 1000 ms. • Both the ports on the link should have the same LSL age configured in order to avoid link flaps.
Step 10	**end** **Example:** Switch(config)# **end**	Exits global configuration mode and returns to privileged EXEC mode.
Step 11	**show interface** [*interface-id*] **rep** [**detail**] **Example:** Switch(config)# **show interface gigabitethernet1/1 rep detail**	(Optional) Displays the REP interface configuration.
Step 12	**copy running-config startup-config** **Example:** Switch(config)# **copy running-config startup-config**	(Optional) Saves your entries in the router startup configuration file.

Setting Manual Preemption for VLAN Load Balancing

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 the other segment configurations have been completed before manually preempting VLAN load balancing. When you enter the **rep preempt delay segment** *segment-id* command, a confirmation message is displayed before the command is executed because preemption might cause network disruption.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **rep preempt segment** *segment-id*
4. **show rep topology segment** *segment-id*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** Example: Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 3	**rep preempt segment** *segment-id* Example: Switch# **rep preempt segment 100** The command will cause a momentary traffic disruption. Do you still want to continue? [confirm]	Manually triggers VLAN load balancing on the segment. You need to confirm the command before it is executed.
Step 4	**show rep topology segment** *segment-id* Example: Switch# **show rep topology segment 100**	(Optional) Displays REP topology information.
Step 5	**end** Example: Switch# **end**	Exits privileged EXEC mode.

Configuring SNMP Traps for REP

You can configure a router to send REP-specific traps to notify the Simple Network Management Protocol (SNMP) server of link-operational status changes and port role changes.

SUMMARY STEPS

1. **configure terminal**
2. **snmp mib rep trap-rate** *value*
3. **end**
4. **show running-config**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**snmp mib rep trap-rate** *value* **Example:** Switch(config)# **snmp mib rep trap-rate 500**	Enables the switch to send REP traps, and sets the number of traps sent per second. • Enter the number of traps sent per second. The range is from 0 to 1000. The default is 0 (no limit is imposed; a trap is sent at every occurrence).
Step 3	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 4	**show running-config** **Example:** Switch# **show running-config**	(Optional) Displays the running configuration, which can be used to verify the REP trap configuration.
Step 5	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the switch startup configuration file.

Monitoring Resilient Ethernet Protocol Configurations

You can display the rep interface and rep topology details using the commands in this topic.

SUMMARY STEPS

1. **show interface** [*interface-id*] **rep** [**detail**]
2. **show rep topology** [**segment** *segment-id*] [**archive**] [**detail**]

DETAILED STEPS

Step 1 **show interface** [*interface-id*] **rep** [**detail**]

Displays REP configuration and status for an interface or for all the interfaces.

• (Optional) **detail**—Displays interface-specific REP information.

Example:

```
Device# show interfaces TenGigabitEthernet4/1 rep detail

TenGigabitEthernet4/1 REP enabled
Segment-id: 3 (Primary Edge)
PortID: 03010015FA66FF80
Preferred flag: No
Operational Link Status: TWO_WAY
Current Key: 02040015FA66FF804050
Port Role: Open
Blocked VLAN: <empty>
Admin-vlan: 1
Preempt Delay Timer: disabled
Configured Load-balancing Block Port: none
Configured Load-balancing Block VLAN: none
STCN Propagate to: none
LSL PDU rx: 999, tx: 652
HFL PDU rx: 0, tx: 0
BPA TLV rx: 500, tx: 4
BPA (STCN, LSL) TLV rx: 0, tx: 0
BPA (STCN, HFL) TLV rx: 0, tx: 0
EPA-ELECTION TLV rx: 6, tx: 5
EPA-COMMAND TLV rx: 0, tx: 0
EPA-INFO TLV rx: 135, tx: 136
```

Step 2 **show rep topology** [**segment** *segment-id*] [**archive**] [**detail**]

Displays REP topology information for a segment or for all the segments, including the primary and secondary edge ports in the segment.

• (Optional) **archive**—Displays the last stable topology.

Note   An archive topology is not retained when the switch reloads.

• (Optional) **detail**—Displays detailed archived information.

Example:

```
Device# show rep topology

REP Segment 1
BridgeName       PortName   Edge Role
---------------- ---------- ---- ----
10.64.106.63     Te5/4      Pri  Open
10.64.106.228    Te3/4           Open
10.64.106.228    Te3/3           Open
```

```
10.64.106.67     Te4/3          Open
10.64.106.67     Te4/4          Alt
10.64.106.63     Te4/4     Sec  Open

REP Segment 3
BridgeName       PortName  Edge Role
---------------- --------- ---- ----
10.64.106.63     Gi50/1    Pri  Open
SVT_3400_2       Gi0/3          Open
SVT_3400_2       Gi0/4          Open
10.64.106.68     Gi40/2         Open
10.64.106.68     Gi40/1         Open
10.64.106.63     Gi50/2    Sec  Alt
```

Configuration Examples for Resilient Ethernet Protocol

This section provides the following configuration examples:

Example: Configuring the REP Administrative VLAN

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(config)# rep admin vlan 100
Switch(config)# end
Switch# show interface gigabitethernet1/1 rep detail

GigabitEthernet1/1 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
```

The following example shows how to create an administrative VLAN per segment. Here, VLAN 2 is configured as the administrative VLAN only for REP segment 2. All the remaining segments that are not configured have VLAN 1 as the administrative VLAN by default.

```
Switch# configure terminal
Switch(config)# rep admin vlan 2 segment 2
Switch(config)# end
```

Example: Configuring a REP Interface

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 the VLANs after a preemption delay of 60 seconds after a segment port failure and recovery. The interface is configured to remain up for 6000 ms without receiving a hello from a neighbor.

```
Switch# configure terminal
Switch (conf)# interface gigabitethernet1/1
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 the same configuration when the interface has no external REP neighbor:

```
Switch# configure terminal
Switch (conf)# interface gigabitethernet1/1
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
Switch (conf-if)# end
```

This example shows how to configure the VLAN blocking configuration shown in the Figure 5. 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 the other VLANs are blocked at the primary edge port E1 (Gigabit Ethernet port 1/1).

Figure 48: Example of VLAN Blocking
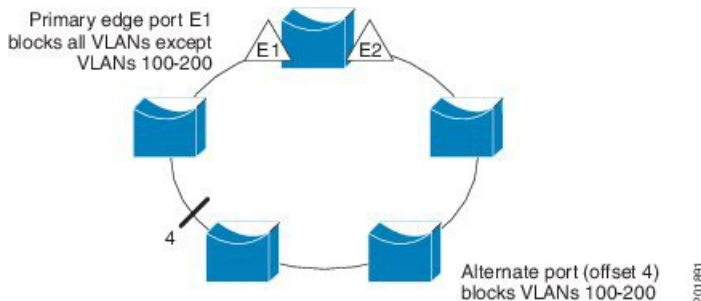

```
Switch# configure terminal
Switch (conf)# interface gigabitethernet1/1
Switch (conf-if)# rep segment 1 edge primary
Switch (conf-if)# rep block port 4 vlan 100-200
Switch (conf-if)# end
```

Example: Configuring a REP Interface

PART V

Network Management

- Configuring Cisco IOS Configuration Engine, on page 379
- Configuring the Cisco Discovery Protocol, on page 401
- Configuring Simple Network Management Protocol, on page 413
- Configuring SPAN and RSPAN, on page 437

C H A P T E R **23**

Configuring Cisco IOS Configuration Engine

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Configuring the Configuration Engine

- Obtain the name of the configuration engine instance to which you are connecting.

- Because the CNS 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.

- All switches configured with the **cns config partial** global configuration command must access the event bus. The DeviceID, as originated on the switch, must match the DeviceID of the corresponding switch definition in the Cisco Configuration Engine. You must know the hostname of the event bus to which you are connecting.

Related Topics

Restrictions for Configuring the Configuration Engine

- 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.

Related Topics

Information About Configuring the Configuration Engine

Cisco Configuration Engine Software

The Cisco Configuration Engine is network management utility software that acts as a configuration service for automating the deployment and management of network devices and services. Each Cisco Configuration Engine manages a group of Cisco devices (switches and routers) and the services that they deliver, storing their configurations and delivering them as needed. The 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.

The Cisco Configuration Engine supports standalone and server modes and has these Cisco Networking Services (CNS) components:

- Configuration service:

    - Web server

    - File manager

    - Namespace mapping server

- Event service (event gateway)

- Data service directory (data models and schema)

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

Figure 49: Cisco Configuration Engine Architectural Overview
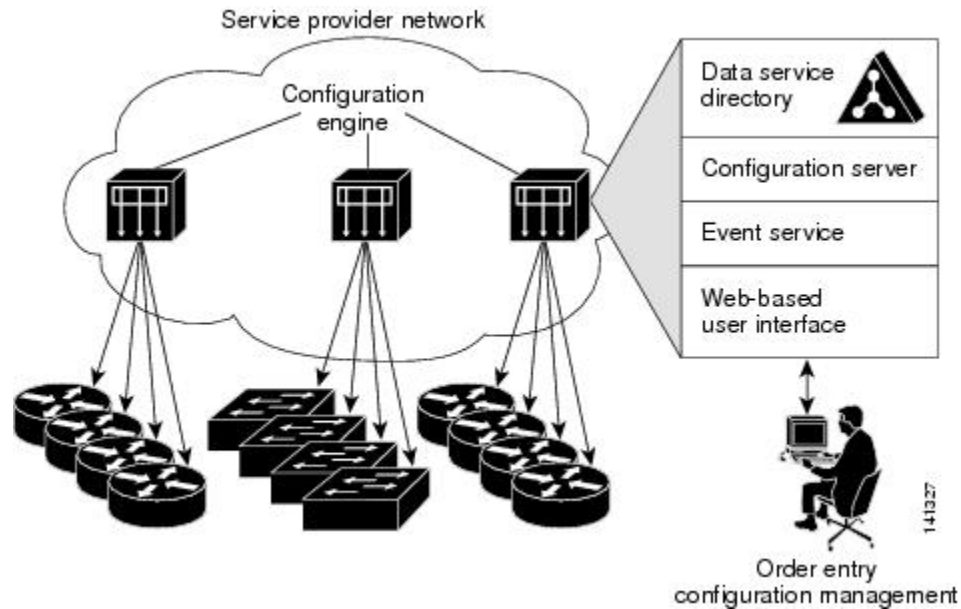
Configuration Service

The Configuration Service is the core component of the Cisco Configuration Engine. It consists of a Configuration Server that works with Cisco IOS CNS agents on the switch. The 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.

The Configuration Service uses the 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 by 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

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

The Event Service is a highly capable publish-and-subscribe communication method. The 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.

Related Topics

NameSpace Mapper

The Cisco Configuration Engine includes the NameSpace Mapper (NSM) that 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.

Cisco Networking Services IDs and Device Hostnames

The Cisco 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.

The Cisco 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.

Related Topics

ConfigID

Each configured switch has a unique ConfigID, which serves as the key into the Cisco 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 the Cisco 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.

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 retains this DeviceID value for the duration of its connection to the switch.

Related Topics

Prerequisites for Configuring the Configuration Engine, on page 379

Hostname and DeviceID

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. For instructions on refreshing DeviceIDs, see "Related Topics."

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.

⚠️

Caution   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*, and you must reinitialize the configuration for your Cisco IOS CNS agent. Otherwise, subsequent partial configuration command operations may malfunction.

Related Topics

Refreshing DeviceIDs, on page 394

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 the Cisco Configuration Engine.

Cisco IOS CNS 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 CNS agent. These agents, embedded in the switch Cisco IOS software, allow the switch to be connected and automatically configured.

Related Topics

Enabling the Cisco IOS CNS Agent, on page 387

Initial Configuration

When the switch first comes up, it attempts to get an IP address by broadcasting a Dynamic Host Configuration Protocol (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 Trivial File Transfer Protocol (TFTP) server Internet Protocol (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 CNS agents initiate communication with the Configuration Engine by using the appropriate ConfigID and EventID. The Configuration Engine maps the Config ID to a template and downloads the full configuration file to the switch.

The following figure shows a sample network configuration for retrieving the initial bootstrap configuration file by using DHCP-based autoconfiguration.

Figure 50: Initial Configuration
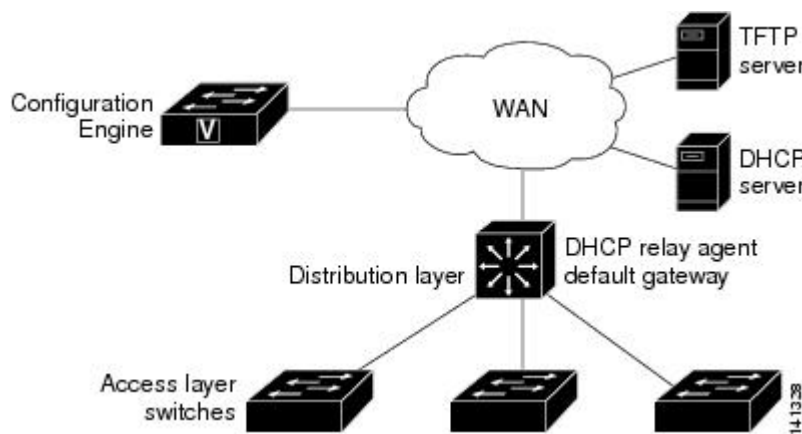
Related Topics

Enabling an Initial Configuration for Cisco IOS CNS Agent, on page 389

Monitoring CNS Configurations, on page 397

Incremental (Partial) Configuration

After the network is running, new services can be added by using the Cisco IOS CNS 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 nonvolatile random-access memory (NVRAM) or wait until signaled to do so.

Related Topics

Enabling a Partial Configuration for Cisco IOS CNS Agent, on page 396

Monitoring CNS Configurations, on page 397

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.

Automated CNS Configuration

To enable automated CNS configuration of the switch, you must first complete the prerequisites listed in this topic. When you complete them, power on the switch. At the **setup** prompt, do nothing; the switch begins the initial configuration. When the full configuration file is loaded on your switch, you do not need to do anything else.

For more information on what happens during initial configuration, see "Related Topics."

Table 40: Prerequisites for Enabling Automatic Configuration

Device	Required Configuration
Access switch	Factory default (no configuration file)
Distribution switch	• IP helper address • Enable DHCP relay agent[1] • 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.

[1] A DHCP Relay is needed only when the DHCP Server is on a different subnet from the client.

How to Configure the Configuration Engine

Enabling the CNS Event Agent

| Note | You must enable the CNS event agent on the switch before you enable the CNS configuration agent. |

Follow these steps to enable the CNS event agent on the switch.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **cns event** {*hostname* | *ip-address*} [*port-number*] [[**keepalive** *seconds retry-count*] [**failover-time** *seconds*] [**reconnect-time** *time*] | **backup**]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** Example: Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 3	**cns event** {*hostname*	*ip-address*} [*port-number*] [[**keepalive** *seconds retry-count*] [**failover-time** *seconds*] [**reconnect-time** *time*]	**backup**] Example: Switch(config)# **cns event 10.180.1.27 keepalive 120 10**	Enables the event agent, and enters the gateway parameters. • For {*hostname*	*ip-address*}, enter either the hostname or the IP address of the event gateway. • (Optional) For *port number*, enter the port number for the event gateway. The default port number is 11011. • (Optional) For **keepalive** *seconds*, enter how often the switch sends keepalive messages. For *retry-count*, enter the number of unanswered keepalive messages that the switch sends before the connection is terminated. The default for each is 0.

	Command or Action	**Purpose**
		• (Optional) For **failover-time** *seconds*, enter how long the switch waits for the primary gateway route after the route to the backup gateway is established.
		• (Optional) For **reconnect-time** *time*, enter the maximum time interval that the switch waits before trying to reconnect to the event gateway.
		• (Optional) Enter **backup** to show that this is the backup gateway. (If omitted, this is the primary gateway.)
		Note Though visible in the command-line help string, the **encrypt** and the **clock-timeout** *time* keywords are not supported.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

What to do next

To verify information about the event agent, use the **show cns event connections** command in privileged EXEC mode.

To disable the CNS event agent, use the **no cns event** { *ip-address* | *hostname* } global configuration command.

Related Topics

Enabling the Cisco IOS CNS Agent

Follow these steps to enable the Cisco IOS CNS agent on the switch.

Before you begin

You must enable the CNS event agent on the switch before you enable this agent.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **cns config initial** {*hostname* | *ip-address*} [*port-number*]
4. **cns config partial** {*hostname* | *ip-address*} [*port-number*]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**
8. Start the Cisco IOS CNS agent on the switch.

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** Example: Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 3	**cns config initial** {*hostname* | *ip-address*} [*port-number*] Example: Switch(config)# **cns config initial 10.180.1.27 10**	Enables the Cisco IOS CNS agent, and enters the configuration server parameters. • For {*hostname* | *ip-address*}, enter either the hostname or the IP address of the configuration server. • (Optional) For *port number*, enter the port number for the configuration server. This command enables the Cisco IOS CNS agent and initiates an initial configuration on the switch.
Step 4	**cns config partial** {*hostname* | *ip-address*} [*port-number*] Example: Switch(config)# **cns config partial 10.180.1.27 10**	Enables the Cisco IOS CNS agent, and enters the configuration server parameters. • For {*hostname* | *ip-address*}, enter either the hostname or the IP address of the configuration server. • (Optional) For *port number*, enter the port number for the configuration server.

	Command or Action	Purpose
		Enables the Cisco IOS CNS agent and initiates a partial configuration on the switch.
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.
Step 8	Start the Cisco IOS CNS agent on the switch.	

What to do next

You can now use the Cisco Configuration Engine to remotely send incremental configurations to the switch.

Related Topics

Cisco IOS CNS Agents, on page 383

Enabling an Initial Configuration for Cisco IOS CNS Agent

Follow these steps to enable the CNS configuration agent and initiate an initial configuration on the switch.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **cns template connect** *name*
4. **cli** *config-text*
5. Repeat Steps 3 to 4 to configure another CNS connect template.
6. **exit**
7. **cns connect** *name* [**retries** *number*] [**retry-interval** *seconds*] [**sleep** *seconds*] [**timeout** *seconds*]
8. **discover** {**controller** *controller-type* | **dlci** [**subinterface** *subinterface-number*] | **interface** [*interface-type*] | **line** *line-type*}
9. **template** *name* [*... name*]
10. Repeat Steps 8 to 9 to specify more interface parameters and CNS connect templates in the CNS connect profile.

11. **exit**

12. **hostname** *name*

13. **ip route** *network-number*

14. **cns id** *interface num* {**dns-reverse** | **ipaddress** | **mac-address**} [**event**] [**image**]

15. **cns id** {**hardware-serial** | **hostname** | **string** *string* | **udi**} [**event**] [**image**]

16. **cns config initial** {*hostname* | *ip-address*} [*port-number*] [**event**] [**no-persist**] [**page** *page*] [**source** *ip-address*] [**syntax-check**]

17. **end**

18. **show running-config**

19. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**cns template connect** *name* **Example:** Switch(config)# **cns template connect template-dhcp**	Enters CNS template connect configuration mode, and specifies the name of the CNS connect template.
Step 4	**cli** *config-text* **Example:** Switch(config-tmpl-conn)# **cli ip address dhcp**	Enters a command line for the CNS connect template. Repeat this step for each command line in the template.
Step 5	Repeat Steps 3 to 4 to configure another CNS connect template.	
Step 6	**exit** **Example:** Switch(config)# **exit**	Returns to global configuration mode.
Step 7	**cns connect** *name* [**retries** *number*] [**retry-interval** *seconds*] [**sleep** *seconds*] [**timeout** *seconds*] **Example:**	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 the Configuration Engine.

	Command or Action	**Purpose**
	`Switch(config)# cns connect dhcp`	• Enter the *name* of the CNS connect profile.
		• (Optional) For **retries** *number*, enter the number of connection retries. The range is 1 to 30. The default is 3.
		• (Optional) For **retry-interval** *seconds*, enter the interval between successive connection attempts to the Configuration Engine. The range is 1 to 40 seconds. The default is 10 seconds.
		• (Optional) For **sleep** *seconds*, enter the amount of time before which the first connection attempt occurs. The range is 0 to 250 seconds. The default is 0.
		• (Optional) For **timeout** *seconds*, enter the amount of time after which the connection attempts end. The range is 10 to 2000 seconds. The default is 120.
Step 8	**discover** {**controller** *controller-type* | **dlci** [**subinterface** *subinterface-number*] | **interface** [*interface-type*] | **line** *line-type*} **Example:** `Switch(config-cns-conn)# discover interface gigabitethernet`	Specifies the interface parameters in the CNS connect profile. • For **controller** *controller-type*, enter the controller type. • For **dlci**, enter the active data-link connection identifiers (DLCIs). (Optional) For **subinterface** *subinterface-number*, specify the point-to-point subinterface number that is used to search for active DLCIs. • For **interface** [*interface-type*], enter the type of interface. • For **line** *line-type*, enter the line type.
Step 9	**template** *name* [... *name*] **Example:** `Switch(config-cns-conn)# template template-dhcp`	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.
Step 10	Repeat Steps 8 to 9 to specify more interface parameters and CNS connect templates in the CNS connect profile.	
Step 11	**exit** **Example:** `Switch(config-cns-conn)# exit`	Returns to global configuration mode.
Step 12	**hostname** *name* **Example:**	Enters the hostname for the switch.

	Command or Action	**Purpose**
	`Switch(config)# hostname device1`	

Step 13	**ip route** *network-number*
	
	Example:
	
	`RemoteSwitch(config)# ip route 172.28.129.22`
	`255.255.255.255 11.11.11.1`
		(Optional) Establishes a static route to the Configuration Engine whose IP address is *network-number*.

Step 14	**cns id** *interface num* {**dns-reverse** | **ipaddress** | **mac-address**} [**event**] [**image**]
	
	Example:
	
	`RemoteSwitch(config)# cns id GigabitEthernet1/0/1`
	`ipaddress`
		(Optional) Sets the unique EventID or ConfigID used by the Configuration Engine. If you enter this command, do not enter the **cns id** {**hardware-serial** | **hostname** | **string** *string* | **udi**} [**event**] [**image**] command.
		
		• For *interface num*, enter 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.
		
		• For {**dns-reverse** | **ipaddress** | **mac-address**}, enter **dns-reverse** to retrieve the hostname and assign it as the unique ID, enter **ipaddress** to use the IP address, or enter **mac-address** to use the MAC address as the unique ID.
		
		• (Optional) Enter **event** to set the ID to be the event-id value used to identify the switch.
		
		• (Optional) Enter **image** to set the ID to be the image-id value used to identify the switch.
		
		Note If both the **event** and **image** keywords are omitted, the image-id value is used to identify the switch.

Step 15	**cns id** {**hardware-serial** | **hostname** | **string** *string* | **udi**} [**event**] [**image**] **Example:** `RemoteSwitch(config)# cns id hostname`	(Optional) Sets the unique EventID or ConfigID used by the Configuration Engine. If you enter this command, do not enter the **cns id** *interface num* {**dns-reverse** | **ipaddress** | **mac-address**} [**event**] [**image**] command. • For {**hardware-serial** | **hostname** | **string** *string* | **udi**}, enter **hardware-serial** to set the switch serial number as the unique ID, enter **hostname** (the default) to select the switch hostname as the unique ID, enter an arbitrary text string for **string** *string* as the unique ID, or enter **udi** to set the unique device identifier (UDI) as the unique ID.
Step 16	**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.

	Command or Action	**Purpose**

	Example:
	
	`RemoteSwitch(config)# cns config initial 10.1.1.1`
	` no-persist`
		• For {*hostname* | *ip-address*}, enter the hostname or the IP address of the configuration server.
		
		• (Optional) For *port-number*, enter the port number of the configuration server. The default port number is 80.
		
		• (Optional) Enable **event** for configuration success, failure, or warning messages when the configuration is finished.
		
		• (Optional) Enable **no-persist** to suppress 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) For **page** *page*, enter the web page of the initial configuration. The default is /Config/config/asp.
		
		• (Optional) Enter **source** *ip-address* to use for source IP address.
		
		• (Optional) Enable **syntax-check** to check 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.

Step 17	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 18	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 19	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

What to do next

To verify information about the configuration agent, use the **show cns config connections** command in privileged EXEC mode.

To disable the CNS Cisco IOS agent, use the **no cns config initial** { *ip-address* | *hostname* } global configuration command.

Related Topics

Refreshing DeviceIDs

Follow these steps to refresh a DeviceID when changing the hostname on the switch.

SUMMARY STEPS

1. **enable**
2. **show cns config connections**
3. Make sure that the CNS event agent is properly connected to the event gateway.
4. **show cns event connections**
5. Record from the output of Step 4 the information for the currently connected connection listed below. You will be using the IP address and port number in subsequent steps of these instructions.
6. **configure terminal**
7. **no cns event** *ip-address port-number*
8. **cns event** *ip-address port-number*
9. **end**
10. Make sure that you have reestablished the connection between the switch and the event connection by examining the output from **show cns event connections**.
11. **show running-config**
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**show cns config connections** **Example:** Switch# **show cns config connections**	Displays whether the CNS event agent is connecting to the gateway, connected, or active, and the gateway used by the event agent, its IP address and port number.
Step 3	Make sure that the CNS event agent is properly connected to the event gateway.	Examine the output of **show cns config connections** for the following:

	Command or Action	**Purpose**
		• Connection is active.
		• Connection is using the currently configured switch hostname. The DeviceID will be refreshed to correspond to the new hostname configuration using these instructions.
Step 4	**show cns event connections** **Example:** Switch# **show cns event connections**	Displays the event connection information for your switch.
Step 5	Record from the output of Step 4 the information for the currently connected connection listed below. You will be using the IP address and port number in subsequent steps of these instructions.	
Step 6	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 7	**no cns event** *ip-address port-number* **Example:** Switch(config)# **no cns event 172.28.129.22 2012**	Specifies the IP address and port number that you recorded in Step 5 in this command. This command breaks the connection between the switch and the event gateway. It is necessary to first break, then reestablish, this connection to refresh the DeviceID.
Step 8	**cns event** *ip-address port-number* **Example:** Switch(config)# **cns event 172.28.129.22 2012**	Specifies the IP address and port number that you recorded in Step 5 in this command. This command reestablishes the connection between the switch and the event gateway.
Step 9	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 10	Make sure that you have reestablished the connection between the switch and the event connection by examining the output from **show cns event connections**.	
Step 11	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.

	Command or Action	**Purpose**
Step 12	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Related Topics

Enabling a Partial Configuration for Cisco IOS CNS Agent

Follow these steps to enable the Cisco IOS CNS agent and to initiate a partial configuration on the switch.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **cns config partial** {*ip-address* | *hostname*} [*port-number*] [**source** *ip-address*]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**cns config partial** {*ip-address*	*hostname*} [*port-number*] [**source** *ip-address*] **Example:** Switch(config)# **cns config partial 172.28.129.22 2013**	Enables the configuration agent, and initiates a partial configuration. • For {*ip-address*	*hostname*}, enter the IP address or the hostname of the configuration server. • (Optional) For *port-number*, enter the port number of the configuration server. The default port number is 80.

	Command or Action	Purpose
		• (Optional) Enter **source** *ip-address* to use for the source IP address. **Note** Though visible in the command-line help string, the **encrypt** keyword is not supported.
Step 4	**end** Example: Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show running-config** Example: Switch# **show running-config**	Verifies your entries.
Step 6	**copy running-config startup-config** Example: Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

What to do next

To verify information about the configuration agent, use either the **show cns config stats** or the **show cns config outstanding** command in privileged EXEC mode.

To disable the Cisco IOS agent, use the **no cns config partial** { *ip-address* | *hostname* } global configuration command. To cancel a partial configuration, use the **cns config cancel** global configuration command.

Related Topics

Monitoring CNS Configurations

Table 41: CNS show Commands

Command	Purpose
show cns config connections Switch# **show cns config connections**	Displays the status of the CNS Cisco IOS CNS agent connections.

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

Related Topics

Additional References

Related Documents

Related Topic	Document Title
Configuration Engine Setup	*Cisco Configuration Engine Installation and Setup Guide, 1.5 for Linux* http://www.cisco.com/en/us/docs/net_mgmt/configuration_engine/1.5/installation_linux/guide/setup_1.html

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	-

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for the Configuration Engine

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Configuring the Cisco Discovery Protocol

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About CDP

CDP Overview

CDP is a device discovery protocol that runs over Layer 2 (the data-link layer) on all Cisco-manufactured devices (routers, bridges, access servers, controllers, 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.

Related Topics

CDP and Stacks

A switch stack appears as a single switch in the network. Therefore, CDP discovers the switch stack, not the individual stack members. The switch stack sends CDP messages to neighboring network devices when there are changes to the switch stack membership, such as stack members being added or removed.

Default CDP Configuration

This table shows the 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

Related Topics

How to Configure CDP

Configuring CDP Characteristics

You can configure these CDP characteristics:

- Frequency of CDP updates

- Amount of time to hold the information before discarding it

- Whether or not to send Version-2 advertisements

Note Steps 3 through 5 are all optional and can be performed in any order.

Follow these steps to configure the CDP characteristics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **cdp timer** *seconds*
4. **cdp holdtime** *seconds*
5. **cdp advertise-v2**
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**cdp timer** *seconds* **Example:** Switch(config)# **cdp timer 20**	(Optional) Sets the transmission frequency of CDP updates in seconds. The range is 5 to 254; the default is 60 seconds.
Step 4	**cdp holdtime** *seconds* **Example:** Switch(config)# **cdp holdtime 60**	(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.
Step 5	**cdp advertise-v2** **Example:** Switch(config)# **cdp advertise-v2**	(Optional) Configures CDP to send Version-2 advertisements. This is the default state.

	Command or Action	**Purpose**
Step 6	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 7	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 8	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

What to do next

Use the **no** form of the CDP commands to return to the default settings.

Related Topics

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.

Follow these steps to disable the CDP device discovery capability.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no cdp run**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**no cdp run** **Example:** Switch(config)# **no cdp run**	Disables CDP.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

What to do next

You must reenable CDP to use it.

Related Topics

Enabling 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.

Follow these steps to enable CDP when it has been disabled.

Before you begin

CDP must be disabled, or it cannot be enabled.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **cdp run**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**cdp run** **Example:** Switch(config)# **cdp run**	Enables CDP if it has been disabled.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:**	Verifies your entries.

	Command or Action	Purpose
	Switch# **show running-config**	
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

What to do next

Use the **show run all** command to show that CDP has been enabled. If you enter only **show run**, the enabling of CDP may not be displayed.

Related Topics

Default CDP Configuration, on page 402

Disabling CDP, on page 404

Disabling CDP on an Interface

CDP is enabled by default on all supported interfaces to send and to receive CDP information.

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.

Note CDP bypass is not supported and may cause a port go into err-disabled state.

Follow these steps to disable CDP on a port.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **no cdp enable**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Switch> **enable**	• Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	Specifies the interface on which you are disabling CDP, and enters interface configuration mode.
Step 4	**no cdp enable** **Example:** Switch(config-if)# **no cdp enable**	Disables CDP on the interface specified in Step 3.
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Related Topics

Enabling CDP on an Interface

CDP is enabled by default on all supported interfaces to send and to receive CDP information.

✎

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.

✎

Note CDP bypass is not supported and may cause a port go into err-disabled state.

Follow these steps to enable CDP on a port on which it has been disabled.

Before you begin

CDP must be disabled on the port that you are trying to CDP enable on, or it cannot be enabled.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **cdp enable**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	Specifies the interface on which you are enabling CDP, and enters interface configuration mode.
Step 4	**cdp enable** **Example:** Switch(config-if)# **cdp enable**	Enables CDP on a disabled interface.

	Command or Action	**Purpose**
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Related Topics

Monitoring and Maintaining CDP

Table 42: Commands for Displaying CDP Information

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* [**version**] [**protocol**]	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.

Command	Description
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.

Related Topics

Configuring CDP Characteristics, on page 402

CDP Overview, on page 401

Additional References

Related Documents

Related Topic	Document Title
System Management Commands	*Network Management Command Reference, Cisco IOS Release 15.2(2)E*

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	-

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Cisco Discovery Protocol

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Configuring Simple Network Management Protocol

Prerequisites for SNMP

Supported SNMP Versions

This software release supports the following 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.

Note To select encryption, enter the **priv** keyword.

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 function and more detailed error message reporting to management stations. The bulk retrieval function 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 method is used when handling an SNMP packet. Available security models are SNMPv1, SNMPv2C, and SNMPv3.

The following table identifies characteristics and compares different combinations of security models and levels:

Table 43: 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.

Model	Level	Authentication	Encryption	Result
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.

Restrictions for SNMP

Version Restrictions

• SNMPv1 does not support informs.

Information About SNMP

SNMP Overview

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 manager can be part of a network management system (NMS) such as Cisco Prime

Infrastructure. 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 Manager Functions

The SNMP manager uses information in the MIB to perform the operations described in the following table:

Table 44: 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.[2]
get-bulk-request[3]	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.

[2] 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.

[3] 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 the following attributes:

• Read-only (RO)—Gives all objects in the MIB except the community strings read access to authorized management stations, but does not allow write access.

• Read-write (RW)—Gives all objects in the MIB read and write access to authorized management stations, 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. The Network Assistant software appends the member switch number (@esN, where N is the switch number) to the first configured RW and RO community strings on the command switch and propagates them to the member switches.

SNMP MIB Variables Access

An example of an NMS is the Cisco Prime Infrastructure network management software. Cisco Prime Infrastructure 2.0 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 the figure, 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 51: SNMP Network
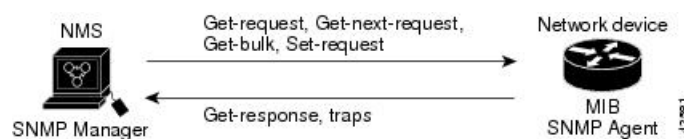
SNMP Notifications

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 the following table to assign an ifIndex value to an interface:

Table 45: ifIndex Values

Interface Type	ifIndex Range
SVI[4]	1–4999
EtherChannel	5001–5048
Tunnel	5078–5142
Physical (such as Gigabit Ethernet or SFP[5]-module interfaces) based on type and port numbers	10000–14500
Null	14501
Loopback and Tunnel	24567+

[4] SVI = switch virtual interface
[5] SFP = small form-factor pluggable

Default SNMP Configuration

Feature	Default Setting
SNMP agent	Disabled[6].
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.

Feature	Default Setting
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.

[6] This is the default when the switch starts and the startup configuration does not have any **snmp-server** global configuration commands.

SNMP Configuration Guidelines

If the switch starts and the switch startup configuration has at least one **snmp-server** global configuration command, the SNMP agent is enabled.

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.

When configuring SNMP, follow these guidelines:

- When configuring an SNMP group, do not specify a notify view. The **snmp-server host** global configuration command auto-generates 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.

- 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 command 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.

- 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 significant results. 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.

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) of the SNMP agent on the device. You reenable all versions of the SNMP agent by the first **snmp-server** global configuration command that you enter. There is no Cisco IOS command specifically designated for enabling SNMP.

Follow these steps to disable the SNMP agent.

Before you begin

The SNMP Agent must be enabled before it can be disabled. The SNMP agent is enabled by the first **snmp-server** global configuration command entered on the device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no snmp-server**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**no snmp-server** **Example:** Switch(config)# **no snmp-server**	Disables the SNMP agent operation.
Step 4	**end** **Example:**	Returns to privileged EXEC mode.

	Command or Action	**Purpose**
	Switch(config)# **end**	
Step 5	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring 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

Follow these steps to configure a community string on the switch.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server community** *string* [**view** *view-name*] [**ro** | **rw**] [*access-list-number*]
4. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	**Purpose**
Step 2	**configure terminal** **Example:** Switch# `configure terminal`	Enters global configuration mode.
Step 3	**snmp-server community** *string* [**view** *view-name*] [**ro** | **rw**] [*access-list-number*] **Example:** Switch(config)# `snmp-server community comaccess ro 4`	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. • For *string*, specify 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) For **view**, specify the view record accessible to the community. • (Optional) Specify either read-only (**ro**) if you want authorized management stations to retrieve MIB objects, or specify 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) For *access-list-number*, enter an IP standard access list numbered from 1 to 99 and 1300 to 1999.
Step 4	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] **Example:** Switch(config)# `access-list 4 deny any`	(Optional) If you specified an IP standard access list number in Step 3, then create the 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 IP address of the SNMP managers that are permitted to use the community string to gain access to the agent. • (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 or Action	Purpose
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

What to do next

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).

To remove a specific community string, use the **no snmp-server** community string global configuration command.

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.

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.

Follow these steps to configure SNMP groups and users on the switch.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server engineID** {**local** *engineid-string* | **remote** *ip-address* [**udp-port** *port-number*] *engineid-string*}
4. **snmp-server group** *group-name* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*]
5. **snmp-server user** *username group-name* {**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*]

6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable**	Enables privileged EXEC mode.
	Example:	• Enter your password if prompted.
	Switch> **enable**	
Step 2	**configure terminal**	Enters global configuration mode.
	Example:	
	Switch# **configure terminal**	
Step 3	**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.
	Example:	• 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. The Step Example configures an engine ID of 123400000000000000000000.
	Switch(config)# **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.
Step 4	**snmp-server group** *group-name* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*]	Configures a new SNMP group on the remote device.
		For *group-name*, specify the name of the group.
	Example:	Specify one of the following security models:
	Switch(config)# **snmp-server group public v2c access lmnop**	• **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 one of the following authentication levels:
		auth—Enables the Message Digest 5 (MD5) and the Secure Hash Algorithm (SHA) packet authentication.

	Command or Action	Purpose
		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).
		(Optional) Enter **read** *readview* with 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) Enter **write** *writeview* with 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) Enter **notify** *notifyview* with a string (not to exceed 64 characters) that is the name of the view in which you specify a notify, inform, or trap.
		(Optional) Enter **access** *access-list* with a string (not to exceed 64 characters) that is the name of the access list.
Step 5	**snmp-server user** *username group-name* {**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*] **Example:** Switch(config)# **snmp-server user Pat public v2c**	Adds a new user for an SNMP group. The *username* is the name of the user on the host that connects to the agent. The *group-name* is the name of the group to which the user is associated. Enter **remote** to specify 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. Enter 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** is 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** you can also configure a private (**priv**) encryption algorithm and password string *priv-password* using the following keywords (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.

Command or Action	Purpose
	• **aes** specifies the use of the DES algorithm. You must select either 128-bit, 192-bit, or 256-bit encryption.
	(Optional) Enter **access** *access-list* with a string (not to exceed 64 characters) that is the name of the access list.
Step 6 **end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 7 **show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 8 **copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring 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.

You can use the **snmp-server host** global configuration command for a specific host to receive the notification types listed in the following table. You can enable any or all of these traps and configure a trap manager to receive them.

Table 46: Device Notification Types

Notification Type Keyword	Description
bridge	Generates STP bridge MIB traps.
cluster	Generates a trap when the cluster configuration changes.
config	Generates a trap for SNMP configuration changes.
copy-config	Generates a trap for SNMP copy configuration changes.

Notification Type Keyword	Description
cpu threshold	Allow CPU-related traps.
entity	Generates a trap for SNMP entity changes.
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 a port VLAN errdisabled. 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. In a switch stack, you can optionally enable notification for flash insertion or removal, which would cause a trap to be issued whenever a switch in the stack is removed or inserted (physical removal, power cycle, or reload).
fru-ctrl	Generates entity field-replaceable unit (FRU) control traps. In the switch stack, this trap refers to the insertion or removal of a switch in the stack.
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.
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: 1. **snmp-server enable traps port-security** 2. **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.

Notification Type Keyword	Description
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.

Follow these steps to configure the switch to send traps or informs to a host.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server engineID remote** *ip-address engineid-string*
4. **snmp-server user** *username group-name* {**remote** *host* [**udp-port** *port*] } {**v1** [**access** *access-list*] | **v2c** [**access** *access-list*] | **v3** [**encrypted**] [**access** *access-list*] [**auth** {**md5** | **sha**} *auth-password*] }
5. **snmp-server group** *group-name* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*]
6. **snmp-server host** *host-addr* [**informs** | **traps**] [**version** {**1** | **2c** | **3** {**auth** | **noauth** | **priv**}}] *community-string* [*notification-type*]
7. **snmp-server enable traps** *notification-types*
8. **snmp-server trap-source** *interface-id*
9. **snmp-server queue-length** *length*
10. **snmp-server trap-timeout** *seconds*
11. **end**
12. **show running-config**
13. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	**Purpose**
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**snmp-server engineID remote** *ip-address engineid-string* **Example:** Switch(config)# **snmp-server engineID remote 192.180.1.27 00000063000100a1c0b4011b**	Specifies the engine ID for the remote host.
Step 4	**snmp-server user** *username group-name* {**remote** *host* [**udp-port** *port*] } {**v1** [**access** *access-list*]	**v2c** [**access** *access-list*]	**v3** [**encrypted**] [**access** *access-list*] [**auth** {**md5**	**sha**} *auth-password*] } **Example:** Switch(config)# **snmp-server user Pat public v2c**	Configures an SNMP user to be associated with the remote host created in Step 3. **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.
Step 5	**snmp-server group** *group-name* {**v1**	**v2c**	**v3** {**auth**	**noauth**	**priv**}} [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*] **Example:** Switch(config)# **snmp-server group public v2c access lmnop**	Configures an SNMP group.
Step 6	**snmp-server host** *host-addr* [**informs**	**traps**] [**version** {**1**	**2c**	**3** {**auth**	**noauth**	**priv**}}] *community-string* [*notification-type*] **Example:** Switch(config)# **snmp-server host 203.0.113.1 comaccess snmp**	Specifies the recipient of an SNMP trap operation. For *host-addr*, specify the name or Internet address of the host (the targeted recipient). (Optional) Specify **traps** (the default) to send SNMP traps to the host. (Optional) Specify **informs** to send SNMP informs to the host. (Optional) Specify the SNMP **version** (**1**, **2c**, or **3**). SNMPv1 does not support informs. (Optional) For Version 3, select authentication level **auth**, **noauth**, or **priv**. **Note** The **priv** keyword is available only when the cryptographic software image is installed. For *community-string*, when **version 1** or **version 2c** is specified, enter the password-like community string sent with the notification operation. When **version 3** is specified, enter the SNMPv3 username.

	Command or Action	**Purpose**
		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) For *notification-type*, use the keywords listed in the table above. If no type is specified, all notifications are sent.

Step 7	**snmp-server enable traps** *notification-types*
	
	Example:
	
	Switch(config)# **snmp-server enable traps snmp**
		Enables the switch to send traps or informs and specifies the type of notifications to be sent. For a list of notification types, see the table above, 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:
		
		     a. snmp-server enable traps port-security
		
		     b. snmp-server enable traps port-security trap-rate *rate*

Step 8	**snmp-server trap-source** *interface-id* **Example:** Switch(config)# **snmp-server trap-source GigabitEthernet1/0/1**	(Optional) Specifies the source interface, which provides the IP address for the trap message. This command also sets the source IP address for informs.
Step 9	**snmp-server queue-length** *length* **Example:** Switch(config)# **snmp-server queue-length 20**	(Optional) Establishes the message queue length for each trap host. The range is 1 to 1000; the default is 10.
Step 10	**snmp-server trap-timeout** *seconds* **Example:** Switch(config)# **snmp-server trap-timeout 60**	(Optional) Defines how often to resend trap messages. The range is 1 to 1000; the default is 30 seconds.
Step 11	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 12	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.

	Command or Action	**Purpose**
Step 13	**copy running-config startup-config** **Example:** Switch# `copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

What to do next

The **snmp-server host** command specifies which hosts receive the notifications. The **snmp-server enable traps** command globally enables the method for the specified notification (for traps and informs). To enable a host to receive an inform, you must configure an **snmp-server host informs** command for the host and globally enable informs by using the **snmp-server enable traps** command.

To remove the specified host from receiving traps, use the **no snmp-server host** *host* global configuration command. The **no snmp-server host** command with no keywords disables traps, but not informs, to the host. To disable informs, use the **no snmp-server host informs** global configuration command. To disable a specific trap type, use the **no snmp-server enable traps** *notification-types* global configuration command.

Setting the Agent Contact and Location Information

Follow these steps to set the system contact and location of the SNMP agent so that these descriptions can be accessed through the configuration file.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server contact** *text*
4. **snmp-server location** *text*
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> `enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# `configure terminal`	Enters global configuration mode.

	Command or Action	**Purpose**
Step 3	**snmp-server contact** *text* **Example:** Switch(config)# **snmp-server contact Dial System Operator at beeper 21555**	Sets the system contact string.
Step 4	**snmp-server location** *text* **Example:** Switch(config)# **snmp-server location Building 3/Room 222**	Sets the system location string.
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Limiting TFTP Servers Used Through SNMP

Follow these steps to limit the TFTP servers used for saving and loading configuration files through SNMP to the servers specified in an access list.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server tftp-server-list** *access-list-number*
4. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**snmp-server tftp-server-list** *access-list-number* **Example:** Switch(config)# **snmp-server tftp-server-list 44**	Limits the TFTP servers used for configuration file copies through SNMP to the servers in the access list. For *access-list-number*, enter an IP standard access list numbered from 1 to 99 and 1300 to 1999.
Step 4	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] **Example:** Switch(config)# **access-list 44 permit 10.1.1.2**	Creates 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 IP address of the TFTP servers that can access the switch. (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. The access list is always terminated by an implicit deny statement for everything.
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.

	Command or Action	Purpose
Step 7	**copy running-config startup-config** **Example:** Switch# `copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Monitoring SNMP Status

To display SNMP input and output statistics, including the number of illegal community string entries, errors, and requested variables, use the **show snmp** privileged EXEC command. You also can use the other privileged EXEC commands listed in the table to display SNMP information.

Table 47: Commands for Displaying SNMP Information

Command	Purpose
show snmp	Displays SNMP statistics.
	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.

SNMP Examples

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
```

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
```

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
```

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
```

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
```

Additional References

Related Documents

Related Topic	Document Title
SNMP Commands	*Network Management Command Reference, Cisco IOS Release 15.2(2)E*

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	-

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Simple Network Management Protocol

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

C H A P T E R **26**

Configuring SPAN and RSPAN

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for SPAN and RSPAN

SPAN

- 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.

RSPAN

- We recommend that you configure an RSPAN VLAN before you configure an RSPAN source or a destination session.

Restrictions for SPAN and RSPAN

SPAN

The restrictions for SPAN are as follows:

- On each switch, you can configure 66 sessions. A maximum of 4 source sessions can be configured and the remaining sessions can be configured as RSPAN destinations sessions. A source session is either a local SPAN session or an RSPAN source session.

- 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, ISL, or IEEE 802.1Q—if the **encapsulation replicate** keywords are specified. If the keywords are not specified, the packets are sent in native form.

- 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 cannot mix source VLANs and filter VLANs within a single SPAN session.

Traffic monitoring in a SPAN session has the following restrictions:

- Sources can be ports or VLANs, but you cannot mix source ports and source VLANs in the same session.

- Wireshark does not capture egress packets when egress span is active.

- You can run both a local SPAN and an RSPAN source session in the same switch or switch stack. The switch or switch stack supports a total of 66 source and RSPAN destination sessions.

- 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.

- You can have multiple destination ports in a SPAN session, but no more than 64 destination ports per switch stack.

- 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 SPAN or RSPAN is enabled, each packet being monitored is sent twice, once as normal traffic and once as a monitored packet. 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.

    - An RSPAN source session cannot have a local destination port.

    - An RSPAN destination session cannot have a local source port.

    - An RSPAN destination session and an RSPAN source session that are using the same RSPAN VLAN cannot run on the same switch or switch stack.

RSPAN

The restrictions for RSPAN are as follows:

- 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.

- 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.

- CDP packets are not forwarded in RSPAN configured VLAN due to limitation in hardware. The workaround is to disable CDP on all the interfaces carrying RSPAN VLAN on the devices connected to the switch.

- 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.

- To use RSPAN, the switch must be running the LAN Base image.

Information About SPAN and RSPAN

SPAN and RSPAN

You can analyze network traffic passing through ports or VLANs by using SPAN or 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 or switch stack. 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.

Figure 52: Example of Local SPAN Configuration on a Single Device

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.
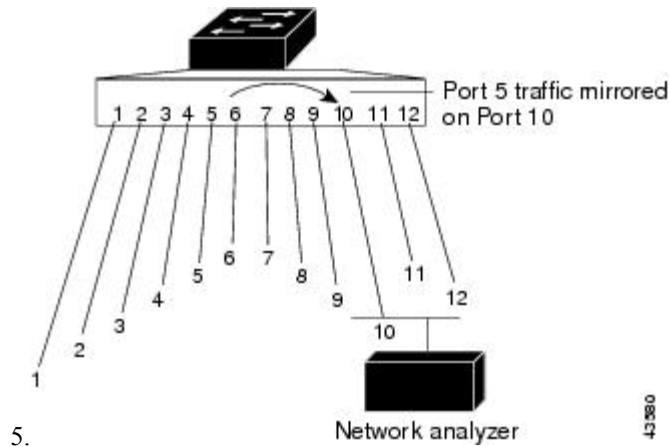
Figure 53: Example of Local SPAN Configuration on a Device Stack

This is an example of a local SPAN in a switch stack, where the source and destination ports reside on different stack members.
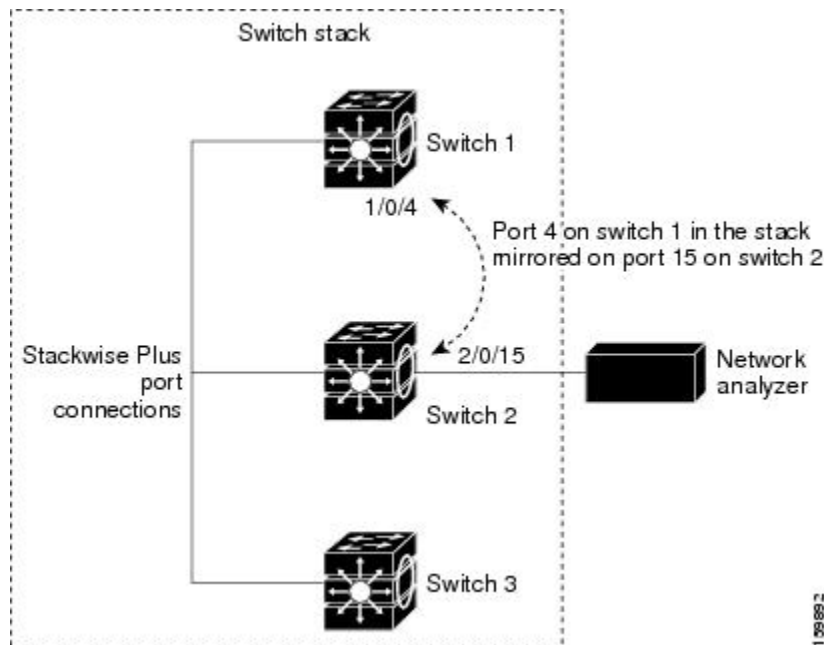
Related Topics

Remote SPAN

RSPAN supports source ports, source VLANs, and destination ports on different switches (or different switch stacks), enabling remote monitoring of multiple switches across your network.

Figure 54: Example of RSPAN Configuration

The figure below 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,
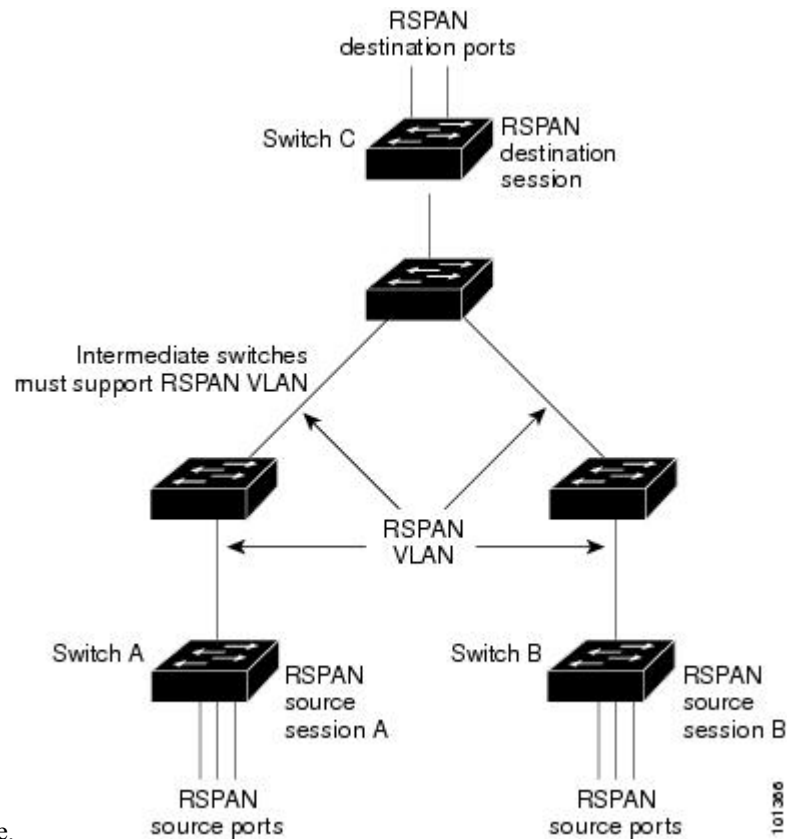
as shown on Switch C in the figure.

Related Topics

Creating an RSPAN Source Session, on page 460

Creating an RSPAN Destination Session, on page 464

Creating an RSPAN Destination Session and Configuring Incoming Traffic, on page 466

Examples: Creating an RSPAN VLAN, on page 476

SPAN and RSPAN Concepts and Terminology

- SPAN Sessions

- Monitored Traffic

- Source Ports

- Source VLANs

- VLAN Filtering

- Destination Port

- RSPAN VLAN

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. The session presents a copy of all RSPAN VLAN packets (except Layer 2 control packets) to the user for analysis.

More than one source session and more than one destination session can be active in the same RSPAN VLAN. Intermediate switches also can separate the RSPAN source and destination sessions. These switches are unable to run RSPAN, but they must respond to the requirements of the RSPAN VLAN.

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.

- You can run both a local SPAN and an RSPAN source session in the same switch or switch stack. The switch or switch stack supports a total of 66 source and RSPAN destination sessions.

- 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.

- You can have multiple destination ports in a SPAN session, but no more than 64 destination ports per switch stack.

- 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 SPAN or RSPAN is enabled, each packet being monitored is sent 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.

    - An RSPAN source session cannot have a local destination port.

> • An RSPAN destination session cannot have a local source port.
>
> • An RSPAN destination session and an RSPAN source session that are using the same RSPAN VLAN cannot run on the same switch or switch stack.

Related Topics

Monitored Traffic

SPAN sessions can monitor these traffic types:

• Receive (Rx) SPAN—Receive (or ingress) SPAN monitors as much as possible all of 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—Transmit (or egress) SPAN monitors as much as possible all of 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 (local or RSPAN) with source ports or VLANs. 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 or switch stack 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 or switch stack 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.

> **Note** When QoS is configured on the SPAN destination port, QoS takes effect immediately.

• 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 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 or switch stack is 64.

Local SPAN and RSPAN destination ports function differently with 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 (untagged, ISL, or 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, ISL, 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. RSPAN VLAN 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 4094), 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.

Related Topics

SPAN and RSPAN Interaction with Other Features

SPAN interacts with these 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.

  Due to limitation in the ASIC, CDP packets are not dropped in RSPAN configured VLAN.

- 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 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.

SPAN and RSPAN and Device Stacks

Because the stack of switches represents one logical switch, local SPAN source ports and destination ports can be in different switches in the stack. Therefore, the addition or deletion of switches in the stack can affect a local SPAN session, as well as an RSPAN source or destination session. An active session can become inactive when a switch is removed from the stack or an inactive session can become active when a switch is added to the stack.

Flow-Based SPAN

You can control the type of network traffic to be monitored in SPAN or RSPAN sessions by using flow-based SPAN (FSPAN) or flow-based RSPAN (FRSPAN), which apply access control lists (ACLs) to the monitored traffic on the source ports. The FSPAN ACLs can be configured to filter IPv4, IPv6, and non-IP monitored traffic.

You apply an ACL to a SPAN session through the interface. It is applied to all the traffic that is monitored on all interfaces in the SPAN session.The packets that are permitted by this ACL are copied to the SPAN destination port. No other packets are copied to the SPAN destination port.

The original traffic continues to be forwarded, and any port, VLAN, and router ACLs attached are applied. The FSPAN ACL does not have any effect on the forwarding decisions. Similarly, the port, VLAN, and router ACLs do not have any effect on the traffic monitoring. If a security input ACL denies a packet and it is not forwarded, the packet is still copied to the SPAN destination ports if the FSPAN ACL permits it. But if the security output ACL denies a packet and it is not sent, it is not copied to the SPAN destination ports. However, if the security output ACL permits the packet to go out, it is only copied to the SPAN destination ports if the FSPAN ACL permits it. This is also true for an RSPAN session.

You can attach three types of FSPAN ACLs to the SPAN session:

- IPv4 FSPAN ACL— Filters only IPv4 packets.

- IPv6 FSPAN ACL— Filters only IPv6 packets.

- MAC FSPAN ACL— Filters only non-IP packets.

The security ACLs have higher priority than the FSPAN ACLs on a switch. If FSPAN ACLs are applied, and you later add more security ACLs that cannot fit in the hardware memory, the FSPAN ACLs that you applied are removed from memory to allow space for the security ACLs. A system message notifies you of this action, which is called unloading. When there is again space for the FSPAN ACLs to reside in memory, they are added to the hardware memory on the switch. A system message notifies you of this action, which is called reloading. The IPv4, IPv6 and MAC FSPAN ACLs can be unloaded or reloaded independently.

If a VLAN-based FSPAN session configured on a stack cannot fit in the hardware memory on one or more switches, it is treated as unloaded on those switches, and traffic meant for the FSPAN ACL and sourcing on that switch is not copied to the SPAN destination ports. The FSPAN ACL continues to be correctly applied, and traffic is copied to the SPAN destination ports on the switches where the FSPAN ACL fits in the hardware memory.

When an empty FSPAN ACL is attached, some hardware functions copy all traffic to the SPAN destination ports for that ACL. If sufficient hardware resources are not available, even an empty FSPAN ACL can be unloaded.

IPv4 and MAC FSPAN ACLs are supported on all feature sets. IPv6 FSPAN ACLs are supported only in the advanced IP Services feature set.

Related Topics

Default SPAN and RSPAN Configuration

Table 48: Default SPAN and RSPAN Configuration

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.

Configuration Guidelines

SPAN Configuration Guidelines

- To remove a source or destination port or VLAN from the SPAN session, use the **no monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} global configuration command or the **no monitor session** *session_number* **destination interface** *interface-id* global configuration command. For destination interfaces, the **encapsulation** options are ignored with the **no** form of the command.

- To monitor all VLANs on the trunk port, use the **no monitor session** *session_number* **filter** global configuration command.

Related Topics

RSPAN Configuration Guidelines

- All the SPAN configuration guidelines apply to RSPAN.

- As 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.

- Access ports (including voice VLAN ports) on the RSPAN VLAN are put in the inactive state.

- 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.

Related Topics

FSPAN and FRSPAN Configuration Guidelines

- When at least one FSPAN ACL is attached, FSPAN is enabled.

- When you attach at least one FSPAN ACL that is not empty to a SPAN session, and you have not attached one or more of the other FSPAN ACLs (for instance, you have attached an IPv4 ACL that is not empty, and have not attached IPv6 and MAC ACLs), FSPAN blocks the traffic that would have been filtered by the unattached ACLs. Therefore, this traffic is not monitored.

Related Topics

How to Configure SPAN and RSPAN

Creating a Local SPAN Session

Follow these steps to create a SPAN session and specify the source (monitored) ports or VLANs and the destination (monitoring) ports.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no monitor session** {*session_number* | **all** | **local** | **remote**}
4. **monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**]
5. **monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**]}

6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**no monitor session** {*session_number* | **all** | **local** | **remote**} **Example:** Switch(config)# **no monitor session all**	Removes any existing SPAN configuration for the session. • For *session_number*, the range is 1 to 66. • **all**—Removes all SPAN sessions. • **local**—Removes all local sessions. • **remote**—Removes all remote SPAN sessions.
Step 4	**monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**] **Example:** Switch(config)# **monitor session 1 source interface gigabitethernet1/0/1**	Specifies the SPAN session and the source port (monitored port). • For *session_number*, the range is 1 to 66. • For *interface-id*, specify 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 48. • For *vlan-id*, specify the source VLAN to monitor. The range is 1 to 4094 (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) [**,** | **-**] Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen.

	Command or Action	**Purpose**
		• (Optional) **both** | **rx** | **tx**—Specifies 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.
		Note You can use the **monitor session** *session_number* **source** command multiple times to configure multiple source ports.
Step 5	**monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**]} **Example:** Switch(config)# **monitor session 1 destination interface gigabitethernet1/0/2 encapsulation replicate**	Specifies the SPAN session and the destination port (monitoring port). The port LED changes to amber when the configuration changes take effect. The LED returns to its original state(green) only after removing the SPAN destination configuration. **Note** For local SPAN, you must use the same session number for the source and destination interfaces. • For *session_number*, specify the session number entered in step 4. • For *interface-id*, specify 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.
Step 6	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 7	**show running-config**	Verifies your entries.

	Command or Action	**Purpose**
	Example: Switch# **show running-config**	
Step 8	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Related Topics

Local SPAN, on page 440

SPAN Sessions, on page 443

SPAN Configuration Guidelines, on page 450

Creating a Local SPAN Session and Configuring Incoming Traffic

Follow these steps to create a SPAN session, to specify the source ports or VLANs and the destination ports, and to enable incoming traffic on the destination port for a network security device (such as a Cisco IDS Sensor Appliance).

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no monitor session** {*session_number* | **all** | **local** | **remote**}
4. **monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**]
5. **monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**] [**ingress** {**dot1q vlan** *vlan-id* | **untagged vlan** *vlan-id* | **vlan** *vlan-id*}]}
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:**	Enters global configuration mode.

	Command or Action	**Purpose**
	Switch# **configure terminal**	
Step 3	**no monitor session** {*session_number*	**all**	**local**	**remote**} **Example:** Switch(config)# **no monitor session all**	Removes any existing SPAN configuration for the session. • For *session_number*, the range is 1 to 66. • **all**—Removes all SPAN sessions. • **local**—Removes all local sessions. • **remote**—Removes all remote SPAN sessions.
Step 4	**monitor session** *session_number* **source** {**interface** *interface-id* / **vlan** *vlan-id*} [**,**	**-**] [**both**	**rx**	**tx**] **Example:** Switch(config)# **monitor session 2 source gigabitethernet1/0/1 rx**	Specifies the SPAN session and the source port (monitored port).
Step 5	**monitor session** *session_number* **destination** {**interface** *interface-id* [**,**	**-**] [**encapsulation replicate**] [**ingress** {**dot1q vlan** *vlan-id*	**untagged vlan** *vlan-id*	**vlan** *vlan-id*}]} **Example:** Switch(config)# **monitor session 2 destination interface gigabitethernet1/0/2 encapsulation replicate ingress dot1q vlan 6**	Specifies the SPAN session, the destination port, the packet encapsulation, and the ingress VLAN and encapsulation. • For *session_number*, specify the session number entered in Step 4. • For *interface-id*, specify 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 to specify 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.

	Command or Action	**Purpose**
Step 6	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 7	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 8	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Related Topics

Specifying VLANs to Filter

Follow these steps to limit SPAN source traffic to specific VLANs.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no monitor session** {*session_number* | **all** | **local** | **remote**}
4. **monitor session** *session_number* **source interface** *interface-id*
5. **monitor session** *session_number* **filter vlan** *vlan-id* [**,** | **-**]
6. **monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**]}
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	**Purpose**
	Switch> **enable**	
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**no monitor session** {*session_number* | **all** | **local** | **remote**} **Example:** Switch(config)# **no monitor session all**	Removes any existing SPAN configuration for the session. • For *session_number*, the range is 1 to 66. • **all**—Removes all SPAN sessions. • **local**—Removes all local sessions. • **remote**—Removes all remote SPAN sessions.
Step 4	**monitor session** *session_number* **source interface** *interface-id* **Example:** Switch(config)# **monitor session 2 source interface gigabitethernet1/0/2 rx**	Specifies the characteristics of the source port (monitored port) and SPAN session. • For *session_number*, the range is 1 to 66. • For *interface-id*, specify the source port to monitor. The interface specified must already be configured as a trunk port.
Step 5	**monitor session** *session_number* **filter vlan** *vlan-id* [**,** | **-**] **Example:** Switch(config)# **monitor session 2 filter vlan 1 - 5 , 9**	Limits the SPAN source traffic to specific VLANs. • For *session_number*, enter the session number specified in Step 4. • For *vlan-id*, the range is 1 to 4094. • (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.
Step 6	**monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**]} **Example:** Switch(config)# **monitor session 2 destination interface gigabitethernet1/0/1**	Specifies the SPAN session and the destination port (monitoring port). • For *session_number*, specify the session number entered in Step 4. • For *interface-id*, specify 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.

	Command or Action	**Purpose**
		• (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).
Step 7	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 8	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 9	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring a VLAN as an RSPAN VLAN

Follow these steps to create a new VLAN, then configure it to be the RSPAN VLAN for the RSPAN session.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vlan** *vlan-id*
4. **remote-span**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	**Purpose**
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**vlan** *vlan-id* **Example:** Switch(config)# **vlan 100**	Enters a VLAN ID to create a VLAN, or enters the VLAN ID of an existing VLAN, and enters VLAN configuration mode. The range is 2 to 1001 and 1006 to 4094. The RSPAN VLAN cannot be VLAN 1 (the default VLAN) or VLAN IDs 1002 through 1005 (reserved for Token Ring and FDDI VLANs).
Step 4	**remote-span** **Example:** Switch(config-vlan)# **remote-span**	Configures the VLAN as an RSPAN VLAN.
Step 5	**end** **Example:** Switch(config-vlan)# **end**	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

What to do next

You must create the RSPAN VLAN in all switches that will participate in RSPAN. If the RSPAN VLAN-ID is in the normal range (lower than 1005) and VTP is enabled in the network, you can create the RSPAN VLAN in one switch, and VTP propagates it to the other switches in the VTP domain. For extended-range VLANs (greater than 1005), you must configure RSPAN VLAN on both source and destination switches and any intermediate switches.

Use VTP pruning to get an efficient flow of RSPAN traffic, or manually delete the RSPAN VLAN from all trunks that do not need to carry the RSPAN traffic.

To remove the remote SPAN characteristic from a VLAN and convert it back to a normal VLAN, use the **no remote-span** VLAN configuration command.

To remove a source port or VLAN from the SPAN session, use the **no monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} global configuration command. To remove the RSPAN VLAN from the session, use the **no monitor session** *session_number* **destination remote vlan** *vlan-id*.

Creating an RSPAN Source Session

Follow these steps to create and start an RSPAN source session and to specify the monitored source and the destination RSPAN VLAN.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no monitor session** {*session_number* | **all** | **local** | **remote**}
4. **monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**]
5. **monitor session** *session_number* **destination remote vlan** *vlan-id*
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**no monitor session** {*session_number* | **all** | **local** | **remote**} **Example:** Switch(config)# **no monitor session 1**	Removes any existing SPAN configuration for the session. • For *session_number*, the range is 1 to 66. • **all**—Removes all SPAN sessions. • **local**—Removes all local sessions. • **remote**—Removes all remote SPAN sessions.
Step 4	**monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**] **Example:**	Specifies the RSPAN session and the source port (monitored port). • For *session_number*, the range is 1 to 66.

	Command or Action	**Purpose**
	Switch(config)# **monitor session 1 source interface gigabitethernet1/0/1 tx**	• Enter a source port or source VLAN for the RSPAN session:
		• For *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 48.
		• For *vlan-id*, specifies the source VLAN to monitor. The range is 1 to 4094 (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) **both**	**rx**	**tx**—Specifies 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.
Step 5	**monitor session** *session_number* **destination remote vlan** *vlan-id* **Example:** Switch(config)# **monitor session 1 destination remote vlan 100**	Specifies the RSPAN session, the destination RSPAN VLAN, and the destination-port group. • For *session_number*, enter the number defined in Step 4. • For *vlan-id*, specify the source RSPAN VLAN to monitor.
Step 6	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 7	**show running-config** **Example:**	Verifies your entries.

Command or Action	Purpose
Switch# **show running-config**	
Step 8	copy running-config startup-config **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Related Topics

Remote SPAN, on page 441

RSPAN VLAN, on page 447

RSPAN Configuration Guidelines, on page 450

Specifying VLANs to Filter

Follow these steps to configure the RSPAN source session to limit RSPAN source traffic to specific VLANs.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no monitor session** {*session_number* | **all** | **local** | **remote**}
4. **monitor session** *session_number* **source interface** *interface-id*
5. **monitor session** *session_number* **filter vlan** *vlan-id* [**,** | **-**]
6. **monitor session** *session_number* **destination remote vlan** *vlan-id*
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**no monitor session** {*session_number* | **all** | **local** | **remote**}	Removes any existing SPAN configuration for the session.

	Command or Action	Purpose
	Example: Switch(config)# **no monitor session 2**	• For *session_number*, the range is 1 to 66. • **all**—Removes all SPAN sessions. • **local**—Removes all local sessions. • **remote**—Removes all remote SPAN sessions.
Step 4	**monitor session** *session_number* **source interface** *interface-id* **Example:** Switch(config)# **monitor session 2 source interface gigabitethernet1/0/2 rx**	Specifies the characteristics of the source port (monitored port) and SPAN session. • For *session_number*, the range is 1 to 66. • For *interface-id*, specify the source port to monitor. The interface specified must already be configured as a trunk port.
Step 5	**monitor session** *session_number* **filter vlan** *vlan-id* [**,** | **-**] **Example:** Switch(config)# **monitor session 2 filter vlan 1 - 5 , 9**	Limits the SPAN source traffic to specific VLANs. • For *session_number*, enter the session number specified in step 4. • For *vlan-id*, the range is 1 to 4094. • (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.
Step 6	**monitor session** *session_number* **destination remote vlan** *vlan-id* **Example:** Switch(config)# **monitor session 2 destination remote vlan 902**	Specifies the RSPAN session and the destination remote VLAN (RSPAN VLAN). • For *session_number*, enter the session number specified in Step 4. • For *vlan-id*, specify the RSPAN VLAN to carry the monitored traffic to the destination port.
Step 7	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 8	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 9	**copy running-config startup-config** **Example:**	(Optional) Saves your entries in the configuration file.

Command or Action	Purpose
Switch# **copy running-config startup-config**	

Creating an RSPAN Destination Session

You configure an RSPAN destination session on a different switch or switch stack; that is, not the switch or switch stack on which the source session was configured.

Follow these steps to define the RSPAN VLAN on that switch, to create an RSPAN destination session, and to specify the source RSPAN VLAN and the destination port.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vlan** *vlan-id*
4. **remote-span**
5. **exit**
6. **no monitor session** {*session_number* | **all** | **local** | **remote**}
7. **monitor session** *session_number* **source remote vlan** *vlan-id*
8. **monitor session** *session_number* **destination interface** *interface-id*
9. **end**
10. **show running-config**
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**vlan** *vlan-id* **Example:** Switch(config)# **vlan 901**	Specifies 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 3 through 5 are not

	Command or Action	**Purpose**
		required because the RSPAN VLAN ID is propagated through the VTP network.
Step 4	**remote-span** **Example:** Switch(config-vlan)# **remote-span**	Identifies the VLAN as the RSPAN VLAN.
Step 5	**exit** **Example:** Switch(config-vlan)# **exit**	Returns to global configuration mode.
Step 6	**no monitor session** {*session_number* | **all** | **local** | **remote**} **Example:** Switch(config)# **no monitor session 1**	Removes any existing SPAN configuration for the session. • For *session_number*, the range is 1 to 66. • **all**—Removes all SPAN sessions. • **local**—Removes all local sessions. • **remote**—Removes all remote SPAN sessions.
Step 7	**monitor session** *session_number* **source remote vlan** *vlan-id* **Example:** Switch(config)# **monitor session 1 source remote vlan 901**	Specifies the RSPAN session and the source RSPAN VLAN. • For *session_number*, the range is 1 to 66. • For *vlan-id*, specify the source RSPAN VLAN to monitor.
Step 8	**monitor session** *session_number* **destination interface** *interface-id* **Example:** Switch(config)# **monitor session 1 destination interface gigabitethernet2/0/1**	Specifies the RSPAN session and the destination interface. • For *session_number*, enter the number defined in Step 7. In an RSPAN destination session, you must use the same session number for the source RSPAN VLAN and the destination port. • For *interface-id*, specify 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.
Step 9	**end** **Example:**	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Switch(config)# **end**	
Step 10	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 11	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Related Topics

Creating an RSPAN Destination Session and Configuring Incoming Traffic

Follow these steps to create an RSPAN destination session, to specify the source RSPAN VLAN and the destination port, and to enable incoming traffic on the destination port for a network security device (such as a Cisco IDS Sensor Appliance).

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no monitor session** {*session_number* | **all** | **local** | **remote**}
4. **monitor session** *session_number* **source remote vlan** *vlan-id*
5. **monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**ingress** {**dot1q vlan** *vlan-id* | **untagged vlan** *vlan-id* | **vlan** *vlan-id*}]}
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	**Purpose**
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**no monitor session** {*session_number* | **all** | **local** | **remote**} **Example:** Switch(config)# **no monitor session 2**	Removes any existing SPAN configuration for the session. • For *session_number*, the range is 1 to 66. • **all**—Removes all SPAN sessions. • **local**—Removes all local sessions. • **remote**—Removes all remote SPAN sessions.
Step 4	**monitor session** *session_number* **source remote vlan** *vlan-id* **Example:** Switch(config)# **monitor session 2 source remote vlan 901**	Specifies the RSPAN session and the source RSPAN VLAN. • For *session_number*, the range is 1 to 66. • For *vlan-id*, specify the source RSPAN VLAN to monitor.
Step 5	**monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**ingress** {**dot1q vlan** *vlan-id* | **untagged vlan** *vlan-id* | **vlan** *vlan-id*}]} **Example:** Switch(config)# **monitor session 2 destination interface gigabitethernet1/0/2 ingress vlan 6**	Specifies the SPAN session, the destination port, the packet encapsulation, and the incoming VLAN and encapsulation. • For *session_number*, enter the number defined in Step 5. In an RSPAN destination session, you must use the same session number for the source RSPAN VLAN and the destination port. • For *interface-id*, specify 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.

	Command or Action	**Purpose**
		• **untagged vlan** *vlan-id* or **vlan** *vlan-id*—Forwards incoming packets with untagged encapsulation type with the specified VLAN as the default VLAN.
Step 6	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 7	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 8	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Related Topics

Configuring an FSPAN Session

Follow these steps to create a SPAN session, specify the source (monitored) ports or VLANs and the destination (monitoring) ports, and configure FSPAN for the session.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no monitor session** {*session_number* | **all** | **local** | **remote**}
4. **monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**]
5. **monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**]}
6. **monitor session** *session_number* **filter** {**ip** | **ipv6** | **mac**} **access-group** {*access-list-number* | *name*}
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**no monitor session** {*session_number* | **all** | **local** | **remote**} **Example:** Switch(config)# **no monitor session 2**	Removes any existing SPAN configuration for the session. • For *session_number*, the range is 1 to 66. • **all**—Removes all SPAN sessions. • **local**—Removes all local sessions. • **remote**—Removes all remote SPAN sessions.
Step 4	**monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**] **Example:** Switch(config)# **monitor session 2 source interface gigabitethernet1/0/1**	Specifies the SPAN session and the source port (monitored port). • For *session_number*, the range is 1 to 66. • For *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 48. • For *vlan-id*, specify the source VLAN to monitor. The range is 1 to 4094 (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) [**,** | **-**]—Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. • (Optional) [**both** | **rx** | **tx**]—Specifies the direction of traffic to monitor. If you do not specify a traffic direction, the SPAN monitors both sent and received traffic.

	Command or Action	**Purpose**
		• **both**—Monitors both sent and received 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.
Step 5	**monitor session** *session_number* **destination** {**interface** *interface-id* [**,**	**-**] [**encapsulation replicate**]}	Specifies the SPAN session and the destination port (monitoring port).
	Example:	• For *session_number*, specify the session number entered in Step 4.
	Switch(config)# **monitor session 2 destination interface gigabitethernet1/0/2 encapsulation replicate**	• For **destination**, specify the following parameters:
		• For *interface-id*, specify 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 For local SPAN, you must use the same session number for the source and destination interfaces.
		You can use **monitor session** *session_number* **destination** command multiple times to configure multiple destination ports.
Step 6	**monitor session** *session_number* **filter** {**ip**	**ipv6**	**mac**} **access-group** {*access-list-number*	*name*}	Specifies the SPAN session, the types of packets to filter, and the ACLs to use in an FSPAN session.
	Example:	• For *session_number*, specify the session number entered in Step 4.
	Switch(config)# **monitor session 2 filter ipv6 access-group 4**	• For *access-list-number*, specify the ACL number that you want to use to filter traffic.
		• For *name*, specify the ACL name that you want to use to filter traffic.

	Command or Action	**Purpose**
Step 7	**end** Example: Switch(config)# **end**	Returns to privileged EXEC mode.
Step 8	**show running-config** Example: Switch# **show running-config**	Verifies your entries.
Step 9	**copy running-config startup-config** Example: Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Related Topics

Configuring an FRSPAN Session

Follow these steps to start an RSPAN source session, specify the monitored source and the destination RSPAN VLAN, and configure FRSPAN for the session.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no monitor session** {*session_number* | **all** | **local** | **remote**}
4. **monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**]
5. **monitor session** *session_number* **destination remote vlan** *vlan-id*
6. **vlan** *vlan-id*
7. **remote-span**
8. **exit**
9. **monitor session** *session_number* **filter** {**ip** | **ipv6** | **mac**} **access-group** {*access-list-number* | *name*}
10. **end**
11. **show running-config**
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable**	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Switch> **enable**	• Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**no monitor session** {*session_number* | **all** | **local** | **remote**} **Example:** Switch(config)# **no monitor session 2**	Removes any existing SPAN configuration for the session. • For *session_number*, the range is 1 to 66. • **all**—Removes all SPAN sessions. • **local**—Removes all local sessions. • **remote**—Removes all remote SPAN sessions.
Step 4	**monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**] **Example:** Switch(config)# **monitor session 2 source interface gigabitethernet1/0/1**	Specifies the SPAN session and the source port (monitored port). • For *session_number*, the range is 1 to 66. • For *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 48. • For *vlan-id*, specify the source VLAN to monitor. The range is 1 to 4094 (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) [**,** | **-**]—Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. • (Optional) [**both** | **rx** | **tx**]—Specifies 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 sent and received traffic. This is the default. • **rx**—Monitors received traffic.

	Command or Action	Purpose
		• **tx**—Monitors sent traffic. **Note** You can use the **monitor session** *session_number* **source** command multiple times to configure multiple source ports.
Step 5	**monitor session** *session_number* **destination remote vlan** *vlan-id* **Example:** Switch(config)# **monitor session 2 destination remote vlan 5**	Specifies the RSPAN session and the destination RSPAN VLAN. • For *session_number*, enter the number defined in Step 4. • For *vlan-id*, specify the destination RSPAN VLAN to monitor.
Step 6	**vlan** *vlan-id* **Example:** Switch(config)# **vlan 10**	Enters the VLAN configuration mode. For *vlan-id*, specify the source RSPAN VLAN to monitor.
Step 7	**remote-span** **Example:** Switch(config-vlan)# **remote-span**	Specifies that the VLAN you specified in Step 5 is part of the RSPAN VLAN.
Step 8	**exit** **Example:** Switch(config-vlan)# **exit**	Returns to global configuration mode.
Step 9	**monitor session** *session_number* **filter** {**ip**	**ipv6**	**mac**} **access-group** {*access-list-number*	*name*} **Example:** Switch(config)# **monitor session 2 filter ip access-group 7**	Specifies the RSPAN session, the types of packets to filter, and the ACLs to use in an FRSPAN session. • For *session_number*, specify the session number entered in Step 4. • For *access-list-number*, specify the ACL number that you want to use to filter traffic. • For *name*, specify the ACL name that you want to use to filter traffic.
Step 10	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 11	**show running-config** **Example:**	Verifies your entries.

Command or Action	Purpose
Switch# **show running-config**	
Step 12	copy running-config startup-config **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Related Topics

Monitoring SPAN and RSPAN Operations

The following table describes the command used to display SPAN and RSPAN operations configuration and results to monitor operations:

Table 49: Monitoring SPAN and RSPAN Operations

Command	Purpose
show monitor	Displays the current SPAN, RSPAN, FSPAN, or FRSPAN configuration.

SPAN and RSPAN Configuration Examples

Example: Configuring Local SPAN

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> enable
Switch# configure terminal
Switch(config)# no monitor session 1
Switch(config)# monitor session 1 source interface gigabitethernet1/0/1
Switch(config)# monitor session 1 destination interface gigabitethernet1/0/2
encapsulation replicate
Switch(config)# end
```

This example shows how to remove port 1 as a SPAN source for SPAN session 1:

```
Switch> enable
Switch# configure terminal
```

```
Switch(config)# no monitor session 1 source interface gigabitethernet1/0/1
Switch(config)# end
```

This example shows how to disable received traffic monitoring on port 1, which was configured for bidirectional monitoring:

```
Switch> enable
Switch# configure terminal
Switch(config)# no monitor session 1 source interface gigabitethernet1/0/1 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> enable
Switch# configure terminal
Switch(config)# no monitor session 2
Switch(config)# monitor session 2 source vlan 1 - 3 rx
Switch(config)# monitor session 2 destination interface gigabitethernet1/0/2
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> enable
Switch# configure terminal
Switch(config)# no monitor session 2
Switch(config)# monitor session 2 source gigabitethernet1/0/1 rx
Switch(config)# monitor session 2 destination interface gigabitethernet1/0/2 encapsulation
 replicate ingress dot1q vlan 6
Switch(config)# end
```

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> enable
Switch# configure terminal
Switch(config)# no monitor session 2
Switch(config)# monitor session 2 source interface gigabitethernet1/0/2 rx
Switch(config)# monitor session 2 filter vlan 1 - 5 , 9
Switch(config)# monitor session 2 destination interface gigabitethernet1/0/1
Switch(config)# end
```

Related Topics

Examples: Creating an RSPAN VLAN

This example shows how to create the RSPAN VLAN 901:

```
Switch> enable
Switch# configure terminal
Switch(config)# vlan 901
Switch(config-vlan)# remote span
Switch(config-vlan)# end
```

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> enable
Switch# configure terminal
Switch(config)# no monitor session 1
Switch(config)# monitor session 1 source interface gigabitethernet1/0/1 tx
Switch(config)# monitor session 1 source interface gigabitethernet1/0/2 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 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> enable
Switch# configure terminal
Switch(config)# no monitor session 2
Switch(config)# monitor session 2 source interface gigabitethernet1/0/2 rx
Switch(config)# monitor session 2 filter vlan 1 - 5 , 9
Switch(config)# monitor session 2 destination remote vlan 902
Switch(config)# end
```

This example shows how to configure VLAN 901 as the source remote VLAN and port 1 as the destination interface:

```
Switch> enable
Switch# configure terminal
Switch(config)# monitor session 1 source remote vlan 901
Switch(config)# monitor session 1 destination interface gigabitethernet2/0/1
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> enable
Switch# configure terminal
Switch(config)# monitor session 2 source remote vlan 901
Switch(config)# monitor session 2 destination interface gigabitethernet1/0/2 ingress vlan 6
Switch(config)# end
```

Related Topics

Additional References

Related Documents

Related Topic	Document Title
System Commands	*Network Management Command Reference, Cisco IOS Release 15.2(2)E*

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	-

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for SPAN and RSPAN

Release	Modification
Cisco IOS Release 15.0(2)EX1	Switch Port Analyzer (SPAN): Allows monitoring of switch traffic on a port or VLAN using a sniffer/analyzer or RMON probe. This feature was introduced.
Cisco IOS 15.0(2)EX1	Flow-based Switch Port Analyzer (SPAN): Provides a method to capture only required data between end hosts by using specified filters. The filters are defined in terms of access lists that limit IPv4, IPv6 or IPv4 + IPv6, or non-IP traffic (MAC) between specified source and destination addresses. This feature was introduced.
Cisco IOS Release 15.0(2)EX1	SPAN destination port support on EtherChannels: Provides the ability to configure a SPAN destination port on an EtherChannel. This feature was introduced.
Cisco IOS Release 15.0(2)EX1	Switch Port Analyzer (SPAN) - distributed egress SPAN: Provides distributed egress SPAN functionality onto line cards in conjunction with ingress SPAN already been distributed to line cards. By distributing egress SPAN functionalities onto line cards, the performance of the system is improved. This feature was introduced.

CHAPTER 27

Configuring QoS

Prerequisites for QoS

Before configuring standard QoS, you must have a thorough understanding of these items:

- The types of applications used and the traffic patterns on your network.

- Traffic characteristics and needs of your network. For example, is the traffic on your network 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.

You can configure QoS on physical ports and on switch virtual interfaces (SVIs). Other than to apply policy maps, you configure the QoS settings, such as classification, queueing, and scheduling, the same way on physical ports and SVIs. When configuring QoS on a physical port, you apply a nonhierarchical policy map. When configuring QoS on an SVI, you apply a nonhierarchical or a hierarchical policy map.

QoS ACL Guidelines

Follow these guidelines when configuring QoS with access control lists (ACLs):

- It is not possible to 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.

- Only one ACL per class map and only one **match** class-map configuration command per class map are supported. The ACL can have multiple ACEs, which match fields against the contents of the packet.

- A trust statement in a policy map requires multiple hardware entries per ACL line. If an input service policy map contains a trust statement in an ACL, the access list might be too large to fit into the available QoS hardware memory, and an error can occur when you apply the policy map to a port. Whenever possible, you should minimize the number of lines is a QoS ACL.

Applying QoS on Interfaces Guidelines

These are the guidelines for configuring QoS on physical ports and SVIs (Layer 3 VLAN interfaces):

- You can configure QoS on physical ports and SVIs. When configuring QoS on physical ports, you create and apply nonhierarchical policy maps. When configuring QoS on SVIs, you can create and apply nonhierarchical and hierarchical policy maps.

- Incoming traffic is classified, policed, and marked down (if configured) regardless of whether the traffic is bridged, routed, or sent to the CPU. It is possible for bridged frames to be dropped or to have their DSCP and CoS values modified.

- Follow these guidelines when configuring policy maps on physical ports or SVIs:

  - You cannot apply the same policy map to a physical port and to an SVI.

  - If VLAN-based QoS is configured on a physical port, the switch removes all the port-based policy maps on the port. The traffic on this physical port is now affected by the policy map attached to the SVI to which the physical port belongs.

  - In a hierarchical policy map attached to an SVI, you can only configure an individual policer at the interface level on a physical port to specify the bandwidth limits for the traffic on the port. The ingress port must be configured as a trunk or as a static-access port. You cannot configure policers at the VLAN level of the hierarchical policy map.

  - The switch does not support aggregate policers in hierarchical policy maps.

  - After the hierarchical policy map is attached to an SVI, the interface-level policy map cannot be modified or removed from the hierarchical policy map. A new interface-level policy map also cannot be added to the hierarchical policy map. If you want these changes to occur, the hierarchical policy map must first be removed from the SVI. You also cannot add or remove a class map specified in the hierarchical policy map.

Policing Guidelines

- The port ASIC device, which controls more than one physical port, supports 256 policers (255 user-configurable policers plus 1 policer reserved for system internal use). The maximum number of user-configurable policers supported per port is 63. Policers are allocated on demand by the software and are constrained by the hardware and ASIC boundaries.

  For example, you could configure 32 policers on a Gigabit Ethernet port and 7 policers on a 10-Gigabit Ethernet port, or you could configure 64 policers on a Gigabit Ethernet port and 4 policers on a 10-Gigabit Ethernet port. Policers are allocated on demand by the software and are constrained by the hardware and ASIC boundaries.

  You cannot reserve policers per port; there is no guarantee that a port will be assigned to any policer.

- Only one policer is applied to a packet on an ingress port. Only the average rate and committed burst parameters are configurable.

- You can create an aggregate policer that is shared by multiple traffic classes within the same nonhierarchical policy map. However, you cannot use the aggregate policer across different policy maps.

- On a port configured for QoS, all traffic received through the port is classified, policed, and marked according to the 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 you have EtherChannel ports configured on your switch, you must configure QoS classification, policing, mapping, and queueing on the individual physical ports that comprise the EtherChannel. You must decide whether the QoS configuration should match on all ports in the EtherChannel.

- If you need to modify a policy map of an existing QoS policy, first remove the policy map from all interfaces, and then modify or copy the policy map. After you finish the modification, apply the modified policy map to the interfaces. If you do not first remove the policy map from all interfaces, high CPU usage can occur, which, in turn, can cause the console to pause for a very long time.

General QoS Guidelines

- 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.

- A switch that is running the IP services feature set supports QoS DSCP and IP precedence matching in policy-based routing (PBR) route maps with these limitations:

  - You cannot apply QoS DSCP mutation maps and PBR route maps to the same interface.

  - You cannot configure DSCP transparency and PBR DSCP route maps on the same switch.

Restrictions for QoS

The following are the restrictions for QoS:

- Ingress queueing and scheduling are not supported on the switch.

- IPv6 QoS is not supported on switches running the LAN base feature set.

- IPv6 ACLs are not supported on switches running the LAN base feature set.

- The switch supports 3 templates: default, vlan, and IPv4. Both the default and vlan templates support IPv6. The IPv4 template does not support IPv6.

- You can configure only individual policers on an SVI.

- For the **class-map** [**match-all** | **match-any**] *class-map-name* global configuration command, because only one **match** command per class map is supported, the **match-all** and **match-any** keywords function the same.

- The number of policers supported is 63 irrespective of the number of SVIs configured. For example, if two SVIs are configured, the number of policers supported is 63 for both SVIs together.

Information About QoS

QoS Implementation

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 congestion-management and congestion-avoidance techniques to provide preferential treatment. Implementing QoS in your network makes network performance more predictable and bandwidth utilization more effective.

The QoS implementation is based on the Differentiated Services (Diff-Serv) architecture, a standard from the Internet Engineering Task Force (IETF). This architecture specifies that each packet is classified upon entry into the network.

The classification is carried in the IP packet header, using 6 bits from the deprecated IP type of service (ToS) field to carry the classification (*class*) information. Classification can also be carried in the Layer 2 frame.

Figure 55: QoS Classification Layers in Frames and Packets

The special bits in the Layer 2 frame or a Layer 3 packet are shown in the following
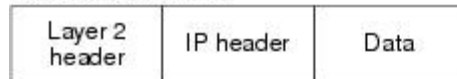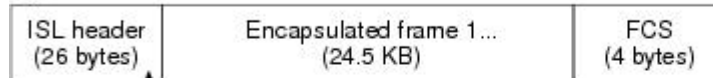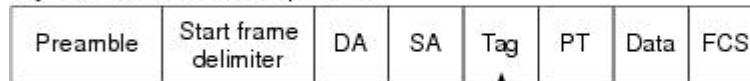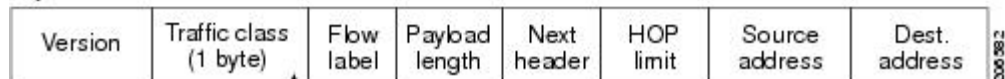
figure:

Layer 2 Frame Prioritization Bits

Layer 2 Inter-Switch Link (ISL) frame headers have a 1-byte User field that carries an IEEE 802.1p class of service (CoS) value in the three least-significant bits. On ports configured as Layer 2 ISL trunks, all traffic is in ISL frames.

Layer 2 802.1Q frame headers have a 2-byte Tag Control Information field that carries the CoS value in the three most-significant bits, which are called the User Priority bits. On ports configured as Layer 2 802.1Q trunks, all traffic is in 802.1Q frames except for traffic in the native VLAN.

Other frame types cannot carry Layer 2 CoS values.

Layer 2 CoS values range from 0 for low priority to 7 for high priority.

Layer 3 Packet Prioritization Bits

Layer 3 IP packets can carry either an IP precedence value or a Differentiated Services Code Point (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.

End-to-End QoS Solution Using Classification

All switches and routers that access the Internet rely on the class information to provide the same forwarding treatment to packets with the same class information and different treatment to packets with different class information. The class information in the packet can be assigned by end hosts or by switches or routers along the way, based on a configured policy, detailed examination of the packet, or both. Detailed examination of the packet is expected to occur closer to the edge of the network, so that the core switches and routers are not overloaded with this task.

Switches and routers along the path can use the class information to limit the amount of resources allocated per traffic class. The behavior of an individual device when handling traffic in the Diff-Serv architecture is called per-hop behavior. If all devices along a path provide a consistent per-hop behavior, you can construct an end-to-end QoS solution.

Implementing QoS in your network can be a simple task or complex task and depends on the QoS features offered by your internetworking devices, the traffic types and patterns in your network, and the granularity of control that you need over incoming and outgoing traffic.

QoS Basic Model

To implement QoS, the switch must distinguish packets or flows from one another (classify), assign a label to indicate the given quality of service as the packets move through the switch, make the packets comply with the configured resource usage limits (police and mark), and provide different treatment (queue and schedule) in all situations where resource contention exists. The switch also needs to ensure that traffic sent from it meets a specific traffic profile (shape).

Figure 56: QoS Basic Model
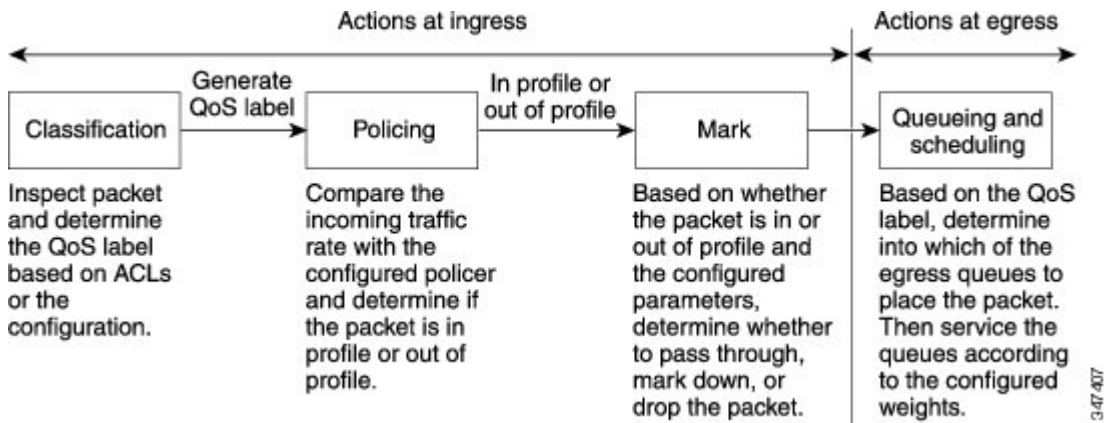
Actions at Ingress Port

Actions at the ingress port include classifying traffic, policing, marking, and scheduling:

• Classifying a distinct path for a packet by associating it with a QoS label. The switch maps the CoS or DSCP in the packet to a QoS label to distinguish one kind of traffic from another. The QoS label that is generated identifies all future QoS actions to be performed on this packet.

• Policing determines whether a packet is in or out of profile by comparing the rate of the incoming traffic to the configured policer. The policer limits the bandwidth consumed by a flow of traffic. The result is passed to the marker.

- Marking evaluates the policer and configuration information for the action to be taken when a packet is out of profile and determines what to do with the packet (pass through a packet without modification, marking down the QoS label in the packet, or dropping the packet).

Note   Queueing and scheduling are only supported at egress and not at ingress on the switch.

Actions at Egress Port

Actions at the egress port include queueing and scheduling:

- Queueing evaluates the QoS packet label and the corresponding DSCP or CoS value before selecting which of the four egress queues to use. Because congestion can occur when multiple ingress ports simultaneously send data to an egress port, WTD differentiates traffic classes and subjects the packets to different thresholds based on the QoS label. If the threshold is exceeded, the packet is dropped.

- Scheduling services the four egress queues based on their configured SRR shared or shaped weights. One of the queues (queue 1) can be the expedited queue, which is serviced until empty before the other queues are serviced.

Classification Overview

Classification is the process of distinguishing one kind of traffic from another by examining the fields in the packet. Classification is enabled only if QoS is globally enabled on the switch. By default, QoS is globally disabled, so no classification occurs.

During classification, the switch performs a lookup and assigns a QoS label to the packet. The QoS label identifies all QoS actions to be performed on the packet and from which queue the packet is sent.

The QoS label is based on the DSCP or the CoS value in the packet and decides the queuing and scheduling actions to perform on the packet. The label is mapped according to the trust setting and the packet type as shown in the Classification Flowchart.

You specify which fields in the frame or packet that you want to use to classify incoming traffic.

Non-IP Traffic Classification

The following table describes the non-IP traffic classification options for your QoS configuration.

Table 50: Non- IP Traffic Classifications

Non-IP Traffic Classification	Description
Trust the CoS value	Trust the CoS value in the incoming frame (configure the port to trust CoS), and then use the configurable CoS-to-DSCP map to generate a DSCP value for the packet.
	Layer 2 ISL frame headers carry the CoS value in the 3 least-significant bits of the 1-byte User field.
	Layer 2 802.1Q frame headers carry the CoS value in the 3 most-significant bits of the Tag Control Information field. CoS values range from 0 for low priority to 7 for high priority.

Non-IP Traffic Classification	Description
Trust the DSCP or trust IP precedence value	Trust the DSCP or trust IP precedence value in the incoming frame. These configurations are meaningless for non-IP traffic. If you configure a port with either of these options and non-IP traffic is received, the switch assigns a CoS value and generates an internal DSCP value from the CoS-to-DSCP map. The switch uses the internal DSCP value to generate a CoS value representing the priority of the traffic.
Perform classification based on configured Layer 2 MAC ACL	Perform the classification based on a configured Layer 2 MAC access control list (ACL), which can examine the MAC source address, the MAC destination address, and other fields. If no ACL is configured, the packet is assigned 0 as the DSCP and CoS values, which means best-effort traffic. Otherwise, the policy-map action specifies a DSCP or CoS value to assign to the incoming frame.

After classification, the packet is sent to the policing and marking stages.

IP Traffic Classification

The following table describes the IP traffic classification options for your QoS configuration.

Table 51: IP Traffic Classifications

IP Traffic Classification	Description
Trust the DSCP value	Trust the DSCP value in the incoming packet (configure the port to trust DSCP), and assign the same DSCP value to the packet. The IETF defines the 6 most-significant bits of the 1-byte ToS field as the DSCP. The priority represented by a particular DSCP value is configurable. DSCP values range from 0 to 63.
	You can also classify IP traffic based on IPv6 DSCP.
	For ports that are on the boundary between two QoS administrative domains, you can modify the DSCP to another value by using the configurable DSCP-to-DSCP-mutation map.
Trust the IP precedence value	Trust the IP precedence value in the incoming packet (configure the port to trust IP precedence), and generate a DSCP value for the packet by using the configurable IP-precedence-to-DSCP map. The IP Version 4 specification defines the 3 most-significant bits of the 1-byte ToS field as the IP precedence. IP precedence values range from 0 for low priority to 7 for high priority.
	You can also classify IP traffic based on IPv6 precedence.
Trust the CoS value	Trust the CoS value (if present) in the incoming packet, and generate a DSCP value for the packet by using the CoS-to-DSCP map. If the CoS value is not present, use the default port CoS value.

IP Traffic Classification	Description
IP standard or an extended ACL	Perform the classification based on a configured IP standard or an extended ACL, which examines various fields in the IP header. If no ACL is configured, the packet is assigned 0 as the DSCP and CoS values, which means best-effort traffic. Otherwise, the policy-map action specifies a DSCP or CoS value to assign to the incoming frame.
Override configured CoS	Override the configured CoS of incoming packets, and apply the default port CoS value to them. For IPv6 packets, the DSCP value is rewritten by using the CoS-to-DSCP map and by using the default CoS of the port. You can do this for both IPv4 and IPv6 traffic.

After classification, the packet is sent to the policing and marking stages.

Classification Flowchart

Figure 57: Classification Flowchart
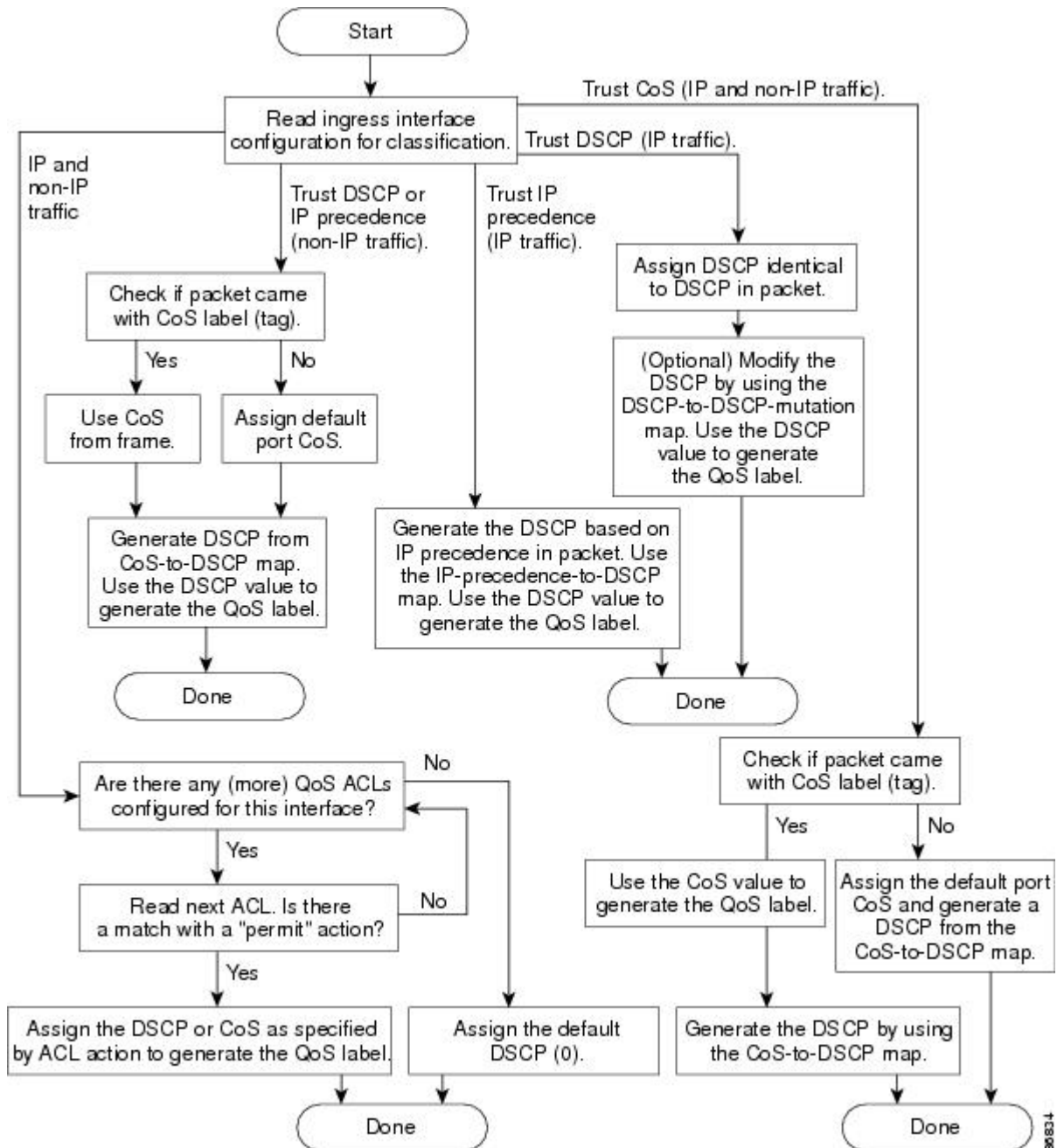
Access Control Lists

You can use IP standard, IP extended, or Layer 2 MAC ACLs to define a group of packets with the same characteristics (class). You can also classify IP traffic based on IPv6 ACLs.

In the QoS context, the permit and deny actions in the access control entries (ACEs) have different meanings from security ACLs:

- If a match with a permit action is encountered (first-match principle), the specified QoS-related action is taken.

- If a match with a deny action is encountered, the ACL being processed is skipped, and the next ACL is processed.

> **Note** Deny action is supported in Cisco IOS Release 3.7.4E and later releases.

- If no match with a permit action is encountered and all the ACEs have been examined, no QoS processing occurs on the packet, and the switch offers best-effort service to the packet.

- If multiple ACLs are configured on a port, the lookup stops after the packet matches the first ACL with a permit action, and QoS processing begins.

> **Note** When creating an access list, note that by default the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.

After a traffic class has been defined with the ACL, you can attach a policy to it. A policy might contain multiple classes with actions specified for each one of them. A policy might include commands to classify the class as a particular aggregate (for example, assign a DSCP) or rate-limit the class. This policy is then attached to a particular port on which it becomes effective.

You implement IP ACLs to classify IP traffic by using the **access-list** global configuration command; you implement Layer 2 MAC ACLs to classify non-IP traffic by using the **mac access-list extended** global configuration command.

Classification Based on Class Maps and Policy Maps

A class map is a mechanism that you use to name a specific traffic flow (or class) and to isolate it from all other traffic. The class map defines the criteria used to match against a specific traffic flow to further classify it. The criteria can include matching the access group defined by the ACL or matching a specific list of DSCP or IP precedence values. 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. After a packet is matched against the class-map criteria, you further classify it through the use of a policy map.

A policy map specifies which traffic class to act on. Actions can include trusting the CoS, DSCP, or IP precedence values in the traffic class; setting a specific DSCP or IP precedence value in the traffic class; or specifying the traffic bandwidth limitations and the action to take when the traffic is out of profile. Before a policy map can be effective, you must attach it to a port.

You create a class map by using the **class-map** global configuration command or the **class** policy-map configuration command. You should use the **class-map** command when the map is shared among many ports. When you enter the **class-map** command, 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.

You can configure a default class by using the **class class-default** policy-map configuration command. Unclassified traffic (traffic specified in the other traffic classes configured on the policy-map) is treated as default traffic.

You create and name a policy map by using the **policy-map** global configuration command. When you enter this command, the switch enters the policy-map configuration mode. In this mode, you specify the actions to take on a specific traffic class by using the **class**, **trust**, or **set** policy-map configuration and policy-map class configuration commands.

The policy map can contain the **police** and **police aggregate** policy-map class configuration commands, which define the policer, the bandwidth limitations of the traffic, and the action to take if the limits are exceeded.

To enable the policy map, you attach it to a port by using the **service-policy** interface configuration command.

You can apply a nonhierarchical policy map to a physical port or an SVI. However, a hierarchical policy map can only be applied to an SVI. A hierarchical policy map contains two levels. The first level, the VLAN level, specifies the actions to be taken against a traffic flow on the SVI. The second level, the interface level, specifies the actions to be taken against the traffic on the physical ports that belong to the SVI. The interface-level actions are specified in the interface-level policy map.

Policing and Marking Overview

After a packet is classified and has a DSCP-based or CoS-based QoS label assigned to it, the policing and marking process can begin.

Policing involves creating a policer that specifies the bandwidth limits for the traffic. Packets that exceed the limits are *out of profile* or *nonconforming*. Each policer decides on a packet-by-packet basis whether the packet is in or out of profile and specifies the actions on the packet. These actions, carried out by the marker, include passing through the packet without modification, dropping the packet, or modifying (marking down) the assigned DSCP of the packet and allowing the packet to pass through. The configurable policed-DSCP map provides the packet with a new DSCP-based QoS label. Marked-down packets use the same queues as the original QoS label to prevent packets in a flow from getting out of order.

Note

All traffic, regardless of whether it is bridged or routed, is subjected to a policer, if one is configured. As a result, bridged packets might be dropped or might have their DSCP or CoS fields modified when they are policed and marked.

You can configure policing (either individual or aggregate policers) on a physical port or an SVI. When configuring policy maps on an SVI, you can create a hierarchical policy map and can define an individual policer only in the secondary interface-level policy map.

After you configure the policy map and policing actions, attach the policy to an ingress port or SVI by using the **service-policy** interface configuration command.

Physical Port Policing

In policy maps on physical ports, you can create the following types of policers:

- Individual—QoS applies the bandwidth limits specified in the policer separately to each matched traffic class. You configure this type of policer within a policy map by using the **police** policy-map class configuration command.

- Aggregate—QoS applies the bandwidth limits specified in an aggregate policer cumulatively to all matched traffic flows. You configure this type of policer by specifying the aggregate policer name within a policy map by using the **police aggregate** policy-map class configuration command. You specify the bandwidth limits of the policer by using the **mls qos aggregate-policer** global configuration command. In this way, the aggregate policer is shared by multiple classes of traffic within a policy map.

Note   You can configure only individual policers on an SVI.

Policing uses a token-bucket algorithm. As each frame is received by the switch, a token is added to the bucket. The bucket has a hole in it and leaks at a rate that you specify as the average traffic rate in bits per second. Each time a token is added to the bucket, the switch verifies that there is enough room in the bucket. If there is not enough room, the packet is marked as nonconforming, and the specified policer action is taken (dropped or marked down).

How quickly the bucket fills is a function of the bucket depth (burst-byte), the rate at which the tokens are removed (rate-bps), and the duration of the burst above the average rate. The size of the bucket imposes an upper limit on the burst length and limits the number of frames that can be transmitted back-to-back. If the burst is short, the bucket does not overflow, and no action is taken against the traffic flow. However, if a burst is long and at a higher rate, the bucket overflows, and the policing actions are taken against the frames in that burst.

You configure the bucket depth (the maximum burst that is tolerated before the bucket overflows) by using the burst-byte option of the **police** policy-map class configuration command or the **mls qos aggregate-policer** global configuration command. You configure how fast (the average rate) that the tokens are removed from the bucket by using the *rate-bps* option of the **police** policy-map class configuration command or the **mls qos aggregate-policer** global configuration command.

Figure 58: Policing and Marking Flowchart on Physical Ports

The following figure shows the policing and marking process when these types of policy maps are configured:

- A nonhierarchical policy map on a physical port.

- The interface level of a hierarchical policy map attached to an SVI. The physical ports are specified in this secondary policy map.
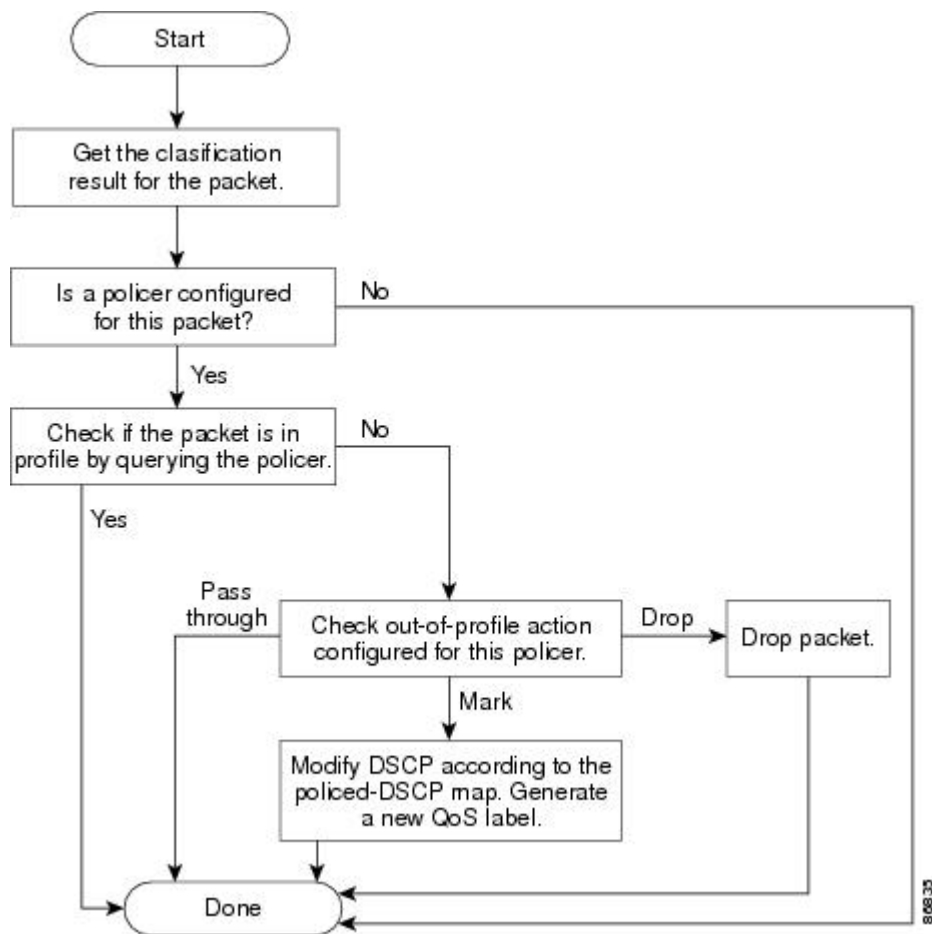
SVI Policing

> **Note**   Before configuring a hierarchical policy map with individual policers on an SVI, you must enable VLAN-based QoS on the physical ports that belong to the SVI. Though a policy map is attached to the SVI, the individual policers only affect traffic on the physical ports specified in the secondary interface level of the hierarchical policy map.

A hierarchical policy map has two levels. The first level, the VLAN level, specifies the actions to be taken against a traffic flow on an SVI. The second level, the interface level, specifies the actions to be taken against the traffic on the physical ports that belong to the SVI and are specified in the interface-level policy map.

When configuring policing on an SVI, you can create and configure a hierarchical policy map with these two levels:

- VLAN level—Create this primary level by configuring class maps and classes that specify the port trust state or set a new DSCP or IP precedence value in the packet. The VLAN-level policy map applies only to the VLAN in an SVI and does not support policers.

- Interface level—Create this secondary level by configuring class maps and classes that specify the individual policers on physical ports the belong to the SVI. The interface-level policy map only supports

individual policers and does not support aggregate policers. You can configure different interface-level policy maps for each class defined in the VLAN-level policy map.

Figure 59: Policing and Marking Flowchart on SVIs

The following figure shows the policing and marking process when hierarchical policy maps on an

SVI.
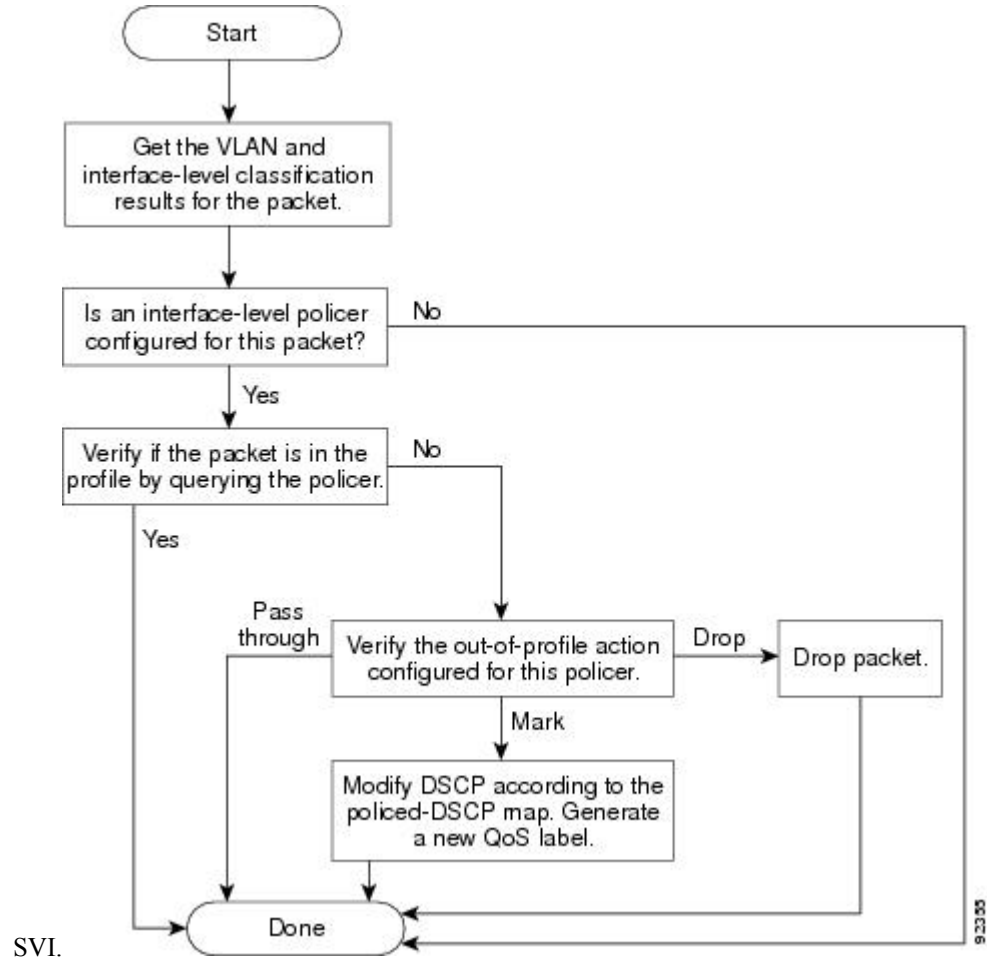
Mapping Tables Overview

During QoS processing, the switch represents the priority of all traffic (including non-IP traffic) with a QoS label based on the DSCP or CoS value from the classification stage.

The following table describes QoS processing and mapping tables.

Table 52: QoS Processing and Mapping Tables

QoS Processing Stage	Mapping Table Usage
Classification	During the classification stage, QoS uses configurable mapping tables to derive a corresponding DSCP or CoS value from a received CoS, DSCP, or IP precedence value. These maps include the CoS-to-DSCP map and the IP-precedence-to-DSCP map. You configure these maps by using the **mls qos map cos-dscp** and the **mls qos map ip-prec-dscp** global configuration commands. On an ingress port configured in the DSCP-trusted state, if the DSCP values are different between the QoS domains, you can apply the configurable DSCP-to-DSCP-mutation map to the port that is on the boundary between the two QoS domains. You configure this map by using the **mls qos map dscp-mutation** global configuration command.
Policing	During policing stage, QoS can assign another DSCP value to an IP or a non-IP packet (if the packet is out of profile and the policer specifies a marked-down value). This configurable map is called the policed-DSCP map. You configure this map by using the **mls qos map policed-dscp** global configuration command.
Pre-scheduling	Before the traffic reaches the scheduling stage, QoS stores the packet in an egress queue according to the QoS label. The QoS label is based on the DSCP or the CoS value in the packet and selects the queue through the DSCP output queue threshold maps or through the CoS output queue threshold maps. In addition to an egress queue, the QOS label also identifies the WTD threshold value. You configure these maps by using the **mls qos srr-queue** { **output** } **dscp-map** and the **mls qos srr-queue** { **output** } **cos-map** global configuration commands.

The CoS-to-DSCP, DSCP-to-CoS, and the IP-precedence-to-DSCP maps have default values that might or might not be appropriate for your network.

The default DSCP-to-DSCP-mutation map and the default policed-DSCP map are null maps; they map an incoming DSCP value to the same DSCP value. The DSCP-to-DSCP-mutation map is the only map you apply to a specific port. All other maps apply to the entire switch.

Queueing and Scheduling Overview

The switch has queues at specific points to help prevent congestion.

Figure 60: Egress Queue Location on Switch
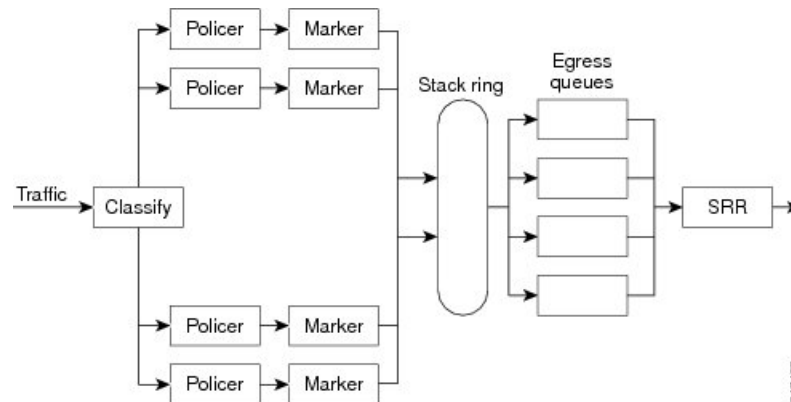
Note   The switch supports 4 egress queues by default and there is an option to enable a total of 8 egress queues. The 8 egress queue configuration is only supported on a standalone switch.

Weighted Tail Drop

As a frame is enqueued to a particular queue, WTD uses the frame's assigned QoS label to subject it to different thresholds. If the threshold is exceeded for that QoS label (the space available in the destination queue is less than the size of the frame), the switch drops the frame.

Each queue has three threshold values. The QoS label determines which of the three threshold values is subjected to the frame. Of the three thresholds, two are configurable (explicit) and one is not (implicit).

Figure 61: WTD and Queue Operation

The following figure shows an example of WTD operating on a queue whose size is 1000 frames. Three drop percentages are configured: 40 percent (400 frames), 60 percent (600 frames), and 100 percent (1000 frames). These percentages indicate that 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
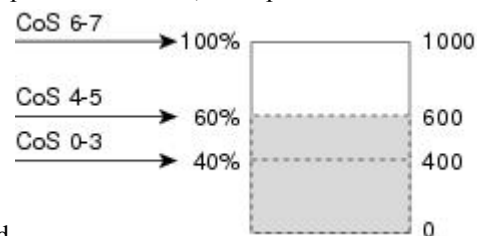
threshold.

In the 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.

Suppose the queue is already filled with 600 frames, and a new frame arrives. It contains CoS values 4 and 5 and is subjected to the 60-percent threshold. If this frame is added to the queue, the threshold will be exceeded, so the switch drops it.

SRR Shaping and Sharing

You can configure SRR on egress queues for sharing or for shaping.

In shaped mode, the egress queues are guaranteed a percentage of the bandwidth, and they are rate-limited to that amount. Shaped traffic does not use more than the allocated bandwidth even if the link is idle. Shaping provides a more even flow of traffic over time and reduces the peaks and valleys of bursty traffic. With shaping, the absolute value of each weight is used to compute the bandwidth available for the queues.

In shared mode, the queues share the bandwidth among them according to the configured weights. The bandwidth is guaranteed at this level but not limited to it. For example, if a queue is empty and no longer requires a share of the link, the remaining queues can expand into the unused bandwidth and share it among them. With sharing, the ratio of the weights controls the frequency of dequeuing; the absolute values are meaningless. Shaping and sharing is configured per interface. Each interface can be uniquely configured.

Queueing and Scheduling on Egress Queues

The following figure shows queueing and scheduling flowcharts for egress ports on the switch.

Figure 62: Queueing and Scheduling Flowchart for Egress Ports on the Switch
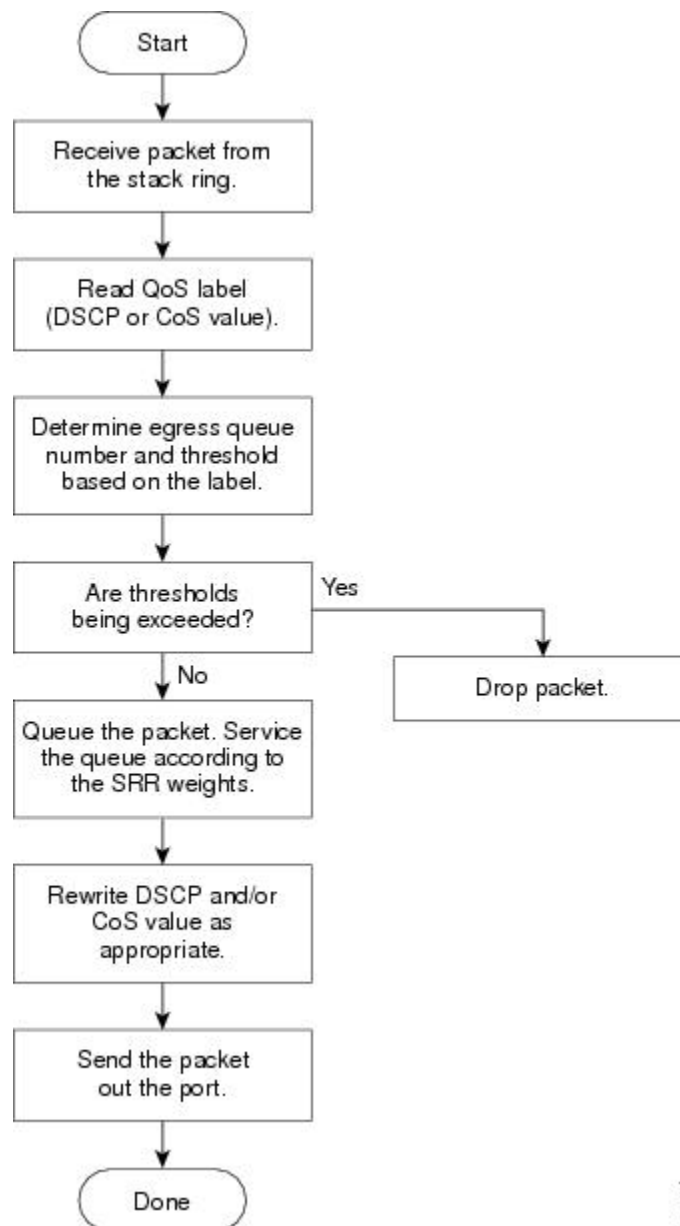
Note   If the expedite queue is enabled, SRR services it until it is empty before servicing the other three queues.

Egress Expedite Queue

Each port supports four egress queues, one of which (queue 1) can be the egress expedite queue. These queues are assigned to a queue-set. All traffic exiting the switch flows through one of these four queues and is subjected to a threshold based on the QoS label assigned to the packet.

Note   If the expedite queue is enabled, SRR services it until it is empty before servicing the other three queues.

Egress Queue Buffer Allocation

The following figure shows the egress queue buffer.

Figure 63: Egress Queue Buffer Allocation

The buffer space is divided between the common pool and the reserved pool. The switch uses a buffer allocation scheme to reserve a minimum amount of buffers for each egress queue, to prevent any queue or port from consuming all the buffers and depriving other queues, and to control whether to grant buffer space to a requesting queue. The switch detects whether the target queue has not consumed more buffers than its reserved amount (under-limit), whether it has consumed all of its maximum buffers (over limit), and whether the common pool is empty (no free buffers) or not empty (free buffers). If the queue is not over-limit, the switch can allocate buffer space from the reserved pool or from the common pool (if it is not empty). If there are no free buffers in the common pool or if the queue is over-limit, the switch drops the

frame.
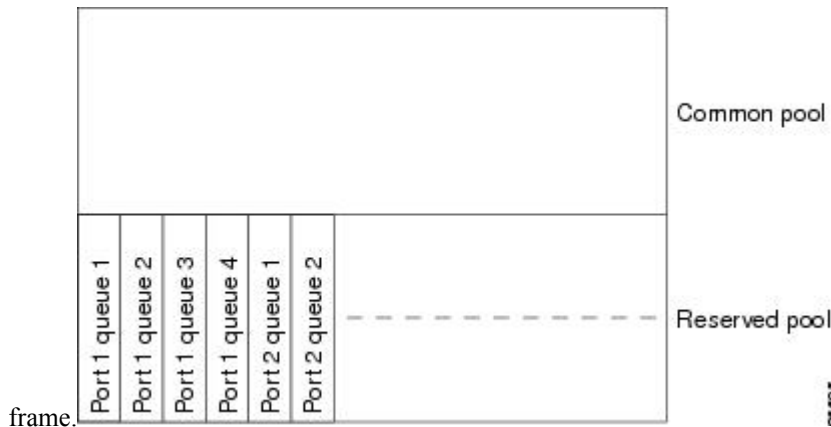
Buffer and Memory Allocation

You guarantee the availability of buffers, set drop thresholds, and configure the maximum memory allocation for a queue-set by using the **mls qos queue-set output** *qset-id* **threshold** *queue-id drop-threshold1 drop-threshold2 reserved-threshold maximum-threshold* global configuration command. Each threshold value is a percentage of the queue's allocated memory, which you specify by using the **mls qos queue-set output** *qset-id* **buffers** *allocation1 ... allocation4* global configuration command. The sum of all the allocated buffers represents the reserved pool, and the remaining buffers are part of the common pool.

Through buffer allocation, you can ensure that high-priority traffic is buffered. For example, if the buffer space is 400, you can allocate 70 percent of it to queue 1 and 10 percent to queues 2 through 4. Queue 1 then has 280 buffers allocated to it, and queues 2 through 4 each have 40 buffers allocated to them.

You can guarantee that the allocated buffers are reserved for a specific queue in a queue-set. For example, if there are 100 buffers for a queue, you can reserve 50 percent (50 buffers). The switch returns the remaining 50 buffers to the common pool. You also can enable a queue in the full condition to obtain more buffers than are reserved for it by setting a maximum threshold. The switch can allocate the needed buffers from the common pool if the common pool is not empty.

Queues and WTD Thresholds

You can assign each packet that flows through the switch to a queue and to a threshold.

Specifically, you map DSCP or CoS values to an egress queue and map DSCP or CoS values to a threshold ID. You use the **mls qos srr-queue output dscp-map queue** *queue-id* {*dscp1...dscp8* | **threshold** *threshold-id dscp1...dscp8*} or the **mls qos srr-queue output cos-map queue** *queue-id* {*cos1...cos8* | **threshold** *threshold-id cos1...cos8*} global configuration command. You can display the DSCP output queue threshold map and the CoS output queue threshold map by using the **show mls qos maps** privileged EXEC command.

The queues use WTD to support distinct drop percentages for different traffic classes. Each queue has three drop thresholds: two configurable (*explicit*) WTD thresholds and one nonconfigurable (*implicit*) threshold preset to the queue-full state. You assign the two WTD threshold percentages for threshold ID 1 and ID 2. The drop threshold for threshold ID 3 is preset to the queue-full state, and you cannot modify it. You map a port to queue-set by using the **queue-set qset-id** interface configuration command. Modify the queue-set configuration to change the WTD threshold percentages.

Shaped or Shared Mode

SRR services each queue-set in shared or shaped mode. You map a port to a queue-set by using the **queue-set** *qset-id* interface configuration command.

You assign shared or shaped weights to the port by using the **srr-queue bandwidth share** *weight1 weight2 weight3 weight4* or the **srr-queue bandwidth shape** *weight1 weight2 weight3 weight4* interface configuration command.

The buffer allocation together with the SRR weight ratios control how much data can be buffered and sent before packets are dropped. The weight ratio is the ratio of the frequency in which the SRR scheduler sends packets from each queue.

All four queues participate in the SRR unless the expedite queue is enabled, in which case the first bandwidth weight is ignored and is not used in the ratio calculation. The expedite queue is a priority queue, and it is serviced until empty before the other queues are serviced. You enable the expedite queue by using the **priority-queue out** interface configuration command.

You can combine the commands described in this section to prioritize traffic by placing packets with particular DSCPs or CoSs into certain queues, by allocating a large queue size or by servicing the queue more frequently, and by adjusting queue thresholds so that packets with lower priorities are dropped.

> **Note**   The egress queue default settings are suitable for most situations. You should change them only when you have a thorough understanding of the egress queues and if these settings do not meet your QoS solution.

Packet Modification

A packet is classified, policed, and queued to provide QoS. The following packet modifications can occur during the process to provide QoS:

- For IP and non-IP packets, classification involves assigning a QoS label to a packet based on the DSCP or CoS of the received packet. However, the packet is not modified at this stage; only an indication of the assigned DSCP or CoS value is carried along.

- During policing, IP and non-IP packets can have another DSCP assigned to them (if they are out of profile and the policer specifies a markdown DSCP). Once again, the DSCP in the packet is not modified, but an indication of the marked-down value is carried along. For IP packets, the packet modification occurs

at a later stage; for non-IP packets the DSCP is converted to CoS and used for queueing and scheduling decisions.

- Depending on the QoS label assigned to a frame and the mutation chosen, the DSCP and CoS values of the frame are rewritten. If you do not configure a table map and if you configure the port to trust the DSCP of the incoming frame, the DSCP value in the frame is not changed, but the CoS is rewritten according to the DSCP-to-CoS map. If you configure the port to trust the CoS of the incoming frame and it is an IP packet, the CoS value in the frame is not changed, but the DSCP might be changed according to the CoS-to-DSCP map.

  The input mutation causes the DSCP to be rewritten depending on the new value of DSCP chosen. The set action in a policy map also causes the DSCP to be rewritten.

Standard QoS Default Configuration

Standard QoS is disabled by default.

When QoS is disabled, there is no concept of trusted or untrusted ports because 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. The packets are switched without any rewrites and classified as best effort without any policing.

When QoS is enabled using the **mls qos** global configuration command and all other QoS settings are at their defaults, traffic is classified as best effort (the DSCP and CoS value is set to 0) without any policing. No policy maps are configured. The default port trust state on all ports is untrusted.

> **Note**   Starting Cisco IOS Release 15.2(1)E, IPv6 QoS is supported on switches running the LAN base license with lanbase-routing template.

Default Egress Queue Configuration

The following tables describe the default egress queue configurations.

The following table shows the default egress queue configuration for each queue-set when QoS is enabled. All ports are mapped to queue-set 1. The port bandwidth limit is set to 100 percent and rate unlimited. Note that for the SRR shaped weights (absolute) feature, a shaped weight of zero indicates that the queue is operating in shared mode. Note that for the SRR shared weights feature, one quarter of the bandwidth is allocated to each queue.

Table 53: Default Egress Queue Configuration

Feature	Queue 1	Queue 2	Queue 3	Queue 4
Buffer allocation	25 percent	25 percent	25 percent	25 percent
WTD drop threshold 1	100 percent	200 percent	100 percent	100 percent
WTD drop threshold 2	100 percent	200 percent	100 percent	100 percent

Feature	Queue 1	Queue 2	Queue 3	Queue 4
Reserved threshold	50 percent	50 percent	50 percent	50 percent
Maximum threshold	400 percent	400 percent	400 percent	400 percent
SRR shaped weights (absolute)	25	0	0	0
SRR shared weights	25	25	25	25

The following table shows the default CoS output queue threshold map when QoS is enabled.

Table 54: Default CoS Output Queue Threshold Map

CoS Value	Queue ID–Threshold ID
0, 1	2–1
2, 3	3–1
4	4–1
5	1–1
6, 7	4–1

The following table shows the default DSCP output queue threshold map when QoS is enabled.

Table 55: Default DSCP Output Queue Threshold Map

DSCP Value	Queue ID–Threshold ID
0–15	2–1
16–31	3–1
32–39	4–1
40–47	1–1
48–63	4–1

Default Mapping Table Configuration

The default DSCP-to-DSCP-mutation map is a null map, which maps an incoming DSCP value to the same DSCP value.

The default policed-DSCP map is a null map, which maps an incoming DSCP value to the same DSCP value (no markdown).

DSCP Maps

Default CoS-to-DSCP Map

You use the CoS-to-DSCP map to map CoS values in incoming packets to a DSCP value that QoS uses internally to represent the priority of the traffic. The following table shows the default CoS-to-DSCP map. If these values are not appropriate for your network, you need to modify them.

Table 56: Default CoS-to-DSCP Map

CoS Value	DSCP Value
0	0
1	8
2	16
3	24
4	32
5	40
6	48
7	56

Default IP-Precedence-to-DSCP Map

You use the IP-precedence-to-DSCP map to map IP precedence values in incoming packets to a DSCP value that QoS uses internally to represent the priority of the traffic. The following table shows the default IP-precedence-to-DSCP map. If these values are not appropriate for your network, you need to modify them.

Table 57: Default IP-Precedence-to-DSCP Map

IP Precedence Value	DSCP Value
0	0
1	8
2	16
3	24
4	32
5	40
6	48
7	56

Default DSCP-to-CoS Map

You use the DSCP-to-CoS map to generate a CoS value, which is used to select one of the four egress queues. The following table shows the default DSCP-to-CoS map. If these values are not appropriate for your network, you need to modify them.

Table 58: Default DSCP-to-CoS Map

DSCP Value	CoS Value
0–7	0
8–15	1
16–23	2
24–31	3
32–39	4
40–47	5
48–55	6
56–63	7

How to Configure QoS

Enabling QoS Globally

By default, QoS is disabled on the switch.

The following procedure to enable QoS globally is required.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos**
3. **end**
4. **show mls qos**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.

	Command or Action	**Purpose**
Step 2	**mls qos** Example: Switch(config)# **mls qos**	Enables QoS globally. QoS operates with the default settings described in the related topic sections below. **Note**　To disable QoS, use the **no mls qos** global configuration command.
Step 3	**end** Example: Switch(config)# **end**	Returns to privileged EXEC mode.
Step 4	**show mls qos** Example: Switch# **show mls qos**	Verifies the QoS configuration.
Step 5	**copy running-config startup-config** Example: Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Enabling VLAN-Based QoS on Physical Ports

By default, VLAN-based QoS is disabled on all physical switch ports. The switch applies QoS, including class maps and policy maps, only on a physical-port basis. You can enable VLAN-based QoS on a switch port.

The following procedure is required on physical ports that are specified in the interface level of a hierarchical policy map on a Switch Virtual Interface (SVI).

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **mls qos vlan-based**
4. **end**
5. **show mls qos interface** *interface-id*
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* Example: Switch(config)# **interface gigabitethernet 1/0/1**	Specifies the physical port, and enter interface configuration mode.
Step 3	**mls qos vlan-based** Example: Switch(config-if)# **mls qos vlan-based**	Enables VLAN-based QoS on the port. **Note** Use the **no mls qos vlan-based** interface configuration command to disable VLAN-based QoS on the physical port.
Step 4	**end** Example: Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 5	**show mls qos interface** *interface-id* Example: Switch# **show mls qos interface gigabitethernet 1/0/1**	Verifies if VLAN-based QoS is enabled on the physical port.
Step 6	**copy running-config startup-config** Example: Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Classification Using Port Trust States

These sections describe how to classify incoming traffic by using port trust states.

Note   Depending on your network configuration, you must perform one or more of these tasks in this module or one or more of the tasks in the Configuring a QoS Policy.

Configuring the Trust State on Ports Within the QoS Domain

Packets entering a QoS domain are classified at the edge of the QoS domain. When the packets are classified at the edge, the switch port within the QoS domain can be configured to one of the trusted states because there is no need to classify the packets at every switch within the QoS domain.

Figure 64: Port Trusted States on Ports Within the QoS Domain
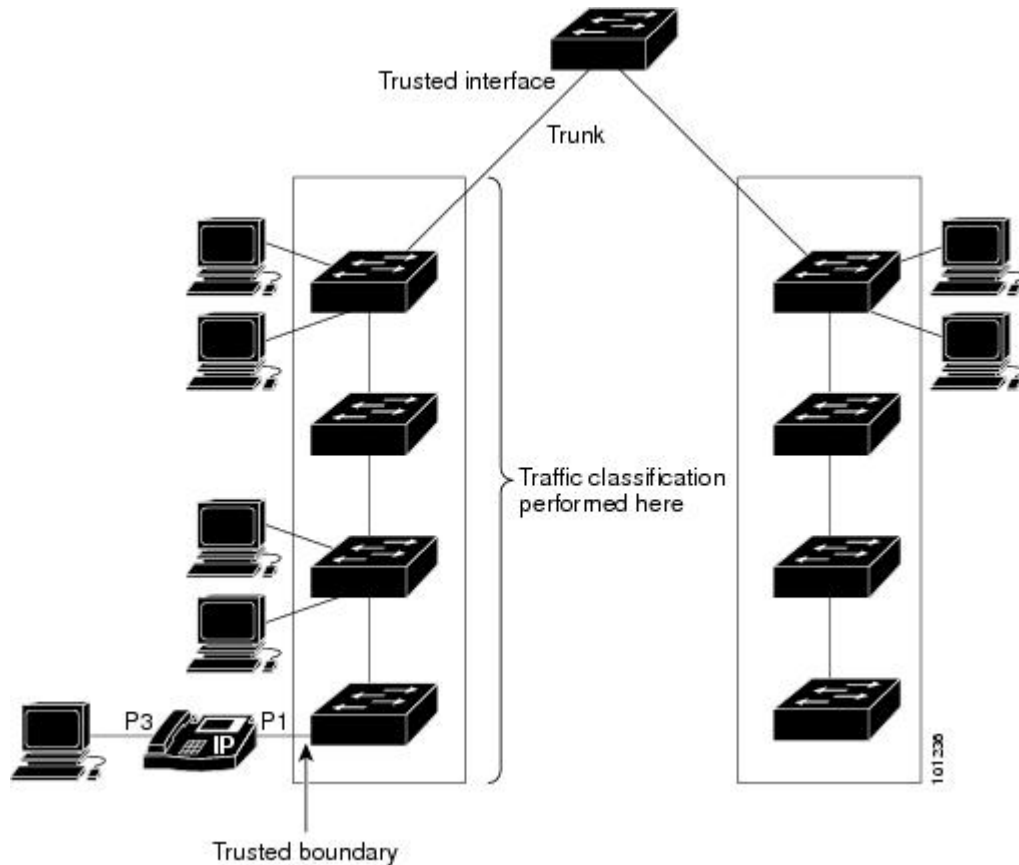
SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **mls qos trust** [**cos** | **dscp** | **ip-precedence**]
4. **end**
5. **show mls qos interface**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal**	Enters global configuration mode.
	Example:	

	Command or Action	Purpose
	Switch# **configure terminal**	
Step 2	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet 1/0/2**	Specifies the port to be trusted, and enters interface configuration mode. Valid interfaces are physical ports.
Step 3	**mls qos trust** [**cos** | **dscp** | **ip-precedence**] **Example:** Switch(config-if)# **mls qos trust cos**	Configures the port trust state. By default, the port is not trusted. If no keyword is specified, the default is **dscp**. The keywords have these meanings: • **cos**—Classifies an ingress packet by using the packet CoS value. For an untagged packet, the port default CoS value is used. The default port CoS value is 0. • **dscp**—Classifies an ingress packet by using the packet DSCP value. For a non-IP packet, the packet CoS value is used if the packet is tagged; for an untagged packet, the default port CoS is used. Internally, the switch maps the CoS value to a DSCP value by using the CoS-to-DSCP map. • **ip-precedence**—Classifies an ingress packet by using the packet IP-precedence value. For a non-IP packet, the packet CoS value is used if the packet is tagged; for an untagged packet, the default port CoS is used. Internally, the switch maps the CoS value to a DSCP value by using the CoS-to-DSCP map. To return a port to its untrusted state, use the **no mls qos trust** interface configuration command.
Step 4	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 5	**show mls qos interface** **Example:** Switch# **show mls qos interface**	Verifies your entries.

	Command or Action	**Purpose**
Step 6	**copy running-config startup-config** Example: Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring the CoS Value for an Interface

QoS assigns the CoS value specified with the **mls qos cos** interface configuration command to untagged frames received on trusted and untrusted ports.

Beginning in privileged EXEC mode, follow these steps to define the default CoS value of a port or to assign the default CoS to all incoming packets on the port.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **mls qos cos** {*default-cos* | **override**}
4. **end**
5. **show mls qos interface**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* Example: Switch(config)# **interface gigabitethernet 1/1/1**	Specifies the port to be configured, and enters interface configuration mode. Valid interfaces include physical ports.
Step 3	**mls qos cos** {*default-cos*	**override**} Example: Switch(config-if)# **mls qos override**	Configures the default CoS value for the port. • For *default-cos,* specify a default CoS value to be assigned to a port. If the packet is untagged, the default CoS value becomes the packet CoS value. The CoS range is 0 to 7. The default is 0. • Use the **override** keyword to override the previously configured trust state of the incoming packet and to

	Command or Action	Purpose
		apply the default port CoS value to the port on all incoming packets. By default, CoS override is disabled.
		Use the **override** keyword when all incoming packets on specified ports deserve higher or lower priority than packets entering from other ports. Even if a port was previously set to trust DSCP, CoS, or IP precedence, this command overrides the previously configured trust state, and all the incoming CoS values are assigned the default CoS value configured with this command. If an incoming packet is tagged, the CoS value of the packet is modified with the default CoS of the port at the ingress port.
		Note To return to the default setting, use the **no mls qos cos** {*default-cos*	**override**} interface configuration command.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show mls qos interface** **Example:** `Switch# show mls qos interface`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring a Trusted Boundary to Ensure Port Security

In a typical network, you connect a Cisco IP Phone to a switch port and cascade devices that generate data packets from the back of the telephone. The Cisco IP Phone guarantees the voice quality through a shared data link by marking the CoS level of the voice packets as high priority (CoS = 5) and by marking the data packets as low priority (CoS = 0). Traffic sent from the telephone to the switch is typically marked with a tag that uses the 802.1Q header. The header contains the VLAN information and the class of service (CoS) 3-bit field, which is the priority of the packet.

For most Cisco IP Phone configurations, the traffic sent from the telephone to the switch should be trusted to ensure that voice traffic is properly prioritized over other types of traffic in the network. By using the **mls qos trust cos** interface configuration command, you configure the switch port to which the telephone is connected to trust the CoS labels of all traffic received on that port. Use the **mls qos trust dscp** interface configuration

command to configure a routed port to which the telephone is connected to trust the DSCP labels of all traffic received on that port.

With the trusted setting, you also can use the trusted boundary feature to prevent misuse of a high-priority queue if a user bypasses the telephone and connects the PC directly to the switch. Without trusted boundary, the CoS labels generated by the PC are trusted by the switch (because of the trusted CoS setting). By contrast, trusted boundary uses CDP to detect the presence of a Cisco IP Phone (such as the Cisco IP Phone 7910, 7935, 7940, and 7960) on a switch port. If the telephone is not detected, the trusted boundary feature disables the trusted setting on the switch port and prevents misuse of a high-priority queue. Note that the trusted boundary feature is not effective if the PC and Cisco IP Phone are connected to a hub that is connected to the switch.

In some situations, you can prevent a PC connected to the Cisco IP Phone from taking advantage of a high-priority data queue. You can use the **switchport priority extend cos** interface configuration command to configure the telephone through the switch CLI to override the priority of the traffic received from the PC.

SUMMARY STEPS

1. **configure terminal**
2. **cdp run**
3. **interface** *interface-id*
4. **cdp enable**
5. Use one of the following:

    - **mls qos trust cos**
    - **mls qos trust dscp**

6. **mls qos trust device cisco-phone**
7. **end**
8. **show mls qos interface**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**cdp run** **Example:** Switch(config)# **cdp run**	Enables CDP globally. By default, CDP is enabled.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface**	Specifies the port connected to the Cisco IP Phone, and enters interface configuration mode. Valid interfaces include physical ports.

	Command or Action	Purpose
	`gigabitethernet 2/1/1`	
Step 4	**cdp enable** **Example:** Switch(config-if)# **cdp enable**	Enables CDP on the port. By default, CDP is enabled.

Step 5	Use one of the following:
	   • **mls qos trust cos**
	   • **mls qos trust dscp**
	Example:
	Switch(config-if)# **mls qos trust cos**
		Configures the switch port to trust the CoS value in traffic received from the Cisco IP Phone.
		or
		Configures the routed port to trust the DSCP value in traffic received from the Cisco IP Phone.
		By default, the port is not trusted.

Step 6	**mls qos trust device cisco-phone** **Example:** Switch(config-if)# **mls qos trust device cisco-phone**	Specifies that the Cisco IP Phone is a trusted device. You cannot enable both trusted boundary and auto-QoS (**auto qos voip** interface configuration command) at the same time; they are mutually exclusive. **Note** To disable the trusted boundary feature, use the **no mls qos trust device** interface configuration command.
Step 7	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 8	**show mls qos interface** **Example:** Switch# **show mls qos interface**	Verifies your entries.
Step 9	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Enabling DSCP Transparency Mode

The switch supports the DSCP transparency feature. It affects only the DSCP field of a packet at egress. By default, DSCP transparency is disabled. The switch modifies the DSCP field in an incoming packet, and the

DSCP field in the outgoing packet is based on the quality of service (QoS) configuration, including the port trust setting, policing and marking, and the DSCP-to-DSCP mutation map.

If DSCP transparency is enabled by using the **no mls qos rewrite ip dscp** command, the switch does not modify the DSCP field in the incoming packet, and the DSCP field in the outgoing packet is the same as that in the incoming packet.

Regardless of the DSCP transparency configuration, the switch modifies the internal DSCP value of the packet, which the switch uses to generate a class of service (CoS) value that represents the priority of the traffic. The switch also uses the internal DSCP value to select an egress queue and threshold.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos**
3. **no mls qos rewrite ip dscp**
4. **end**
5. **show mls qos interface** [*interface-id*]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**mls qos** **Example:** Switch(config)# **mls qos**	Enables QoS globally.
Step 3	**no mls qos rewrite ip dscp** **Example:** Switch(config)# **no mls qos rewrite ip dscp**	Enables DSCP transparency. The switch is configured to not modify the DSCP field of the IP packet.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show mls qos interface** [*interface-id*] **Example:**	Verifies your entries.

	Command or Action	Purpose
	Switch# **show mls qos interface gigabitethernet 2/1/1**	
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

DSCP Transparency Mode

To configure the switch to modify the DSCP value based on the trust setting or on an ACL by disabling DSCP transparency, use the **mls qos rewrite ip dscp** global configuration command.

If you disable QoS by using the **no mls qos** global configuration command, the CoS and DSCP values are not changed (the default QoS setting).

If you enter the **no mls qos rewrite ip dscp** global configuration command to enable DSCP transparency and then enter the **mls qos trust** [**cos** | **dscp**] interface configuration command, DSCP transparency is still enabled.

Note For Catalyst 2960-L switches, DSCP transparency is enabled by default.

Configuring the DSCP Trust State on a Port Bordering Another QoS Domain

If you are administering two separate QoS domains between which you want to implement QoS features for IP traffic, you can configure the switch ports bordering the domains to a DSCP-trusted state. The receiving port accepts the DSCP-trusted value and avoids the classification stage of QoS. If the two domains use different DSCP values, you can configure the DSCP-to-DSCP-mutation map to translate a set of DSCP values to match the definition in the other domain.

Figure 65: DSCP-Trusted State on a Port Bordering Another QoS Domain

QoS Domain 1

QoS Domain 2

IP traffic

Set interface to the DSCP-trusted state.
Configure the DSCP-to-DSCP-mutation map.
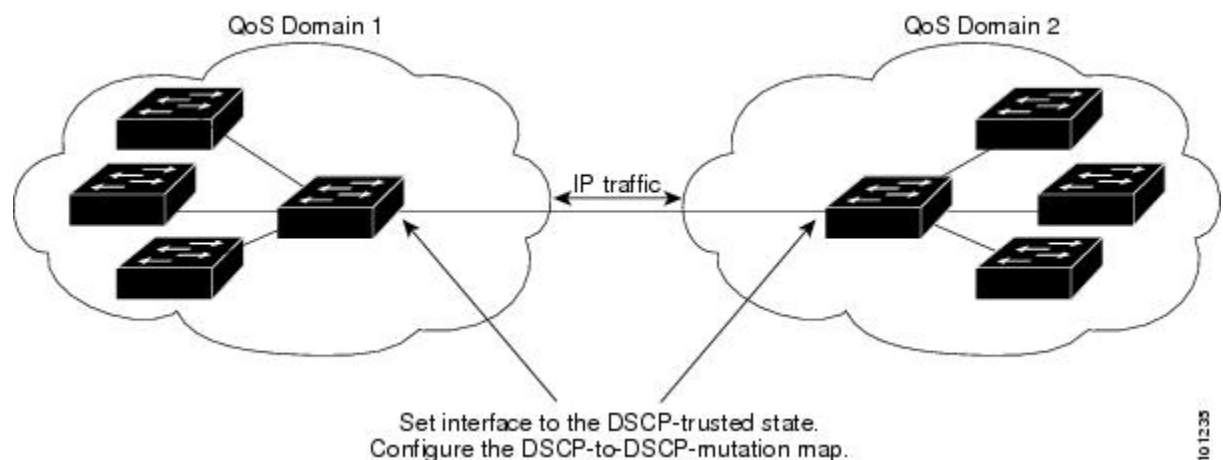
Beginning in privileged EXEC mode, follow these steps to configure the DSCP-trusted state on a port and modify the DSCP-to-DSCP-mutation map. To ensure a consistent mapping strategy across both QoS domains, you must perform this procedure on the ports in both domains.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos map dscp-mutation** *dscp-mutation-name in-dscp* **to** *out-dscp*
3. **interface** *interface-id*
4. **mls qos trust dscp**
5. **mls qos dscp-mutation** *dscp-mutation-name*
6. **end**
7. **show mls qos maps dscp-mutation**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**mls qos map dscp-mutation** *dscp-mutation-name in-dscp* **to** *out-dscp* **Example:** Switch(config)# **mls qos map dscp-mutation gigabitethernet1/0/2-mutation 10 11 12 13 to 30**	Modifies the DSCP-to-DSCP-mutation map. The default DSCP-to-DSCP-mutation map is a null map, which maps an incoming DSCP value to the same DSCP value. • For *dscp-mutation-name*, enter the mutation map name. You can create more than one map by specifying a new name. • For *in-dscp*, enter up to eight DSCP values separated by spaces. Then enter the **to** keyword. • For *out-dscp*, enter a single DSCP value. The DSCP range is 0 to 63.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/2**	Specifies the port to be trusted, and enter interface configuration mode. Valid interfaces include physical ports.
Step 4	**mls qos trust dscp** **Example:**	Configures the ingress port as a DSCP-trusted port. By default, the port is not trusted.

	Command or Action	Purpose
	Switch(config-if)# **mls qos trust dscp**	**Note** To return a port to its non-trusted state, use the **no mls qos trust** interface configuration command.
Step 5	**mls qos dscp-mutation** *dscp-mutation-name* **Example:** Switch(config-if)# **mls qos dscp-mutation gigabitethernet1/0/2-mutation**	Applies the map to the specified ingress DSCP-trusted port. For *dscp-mutation-name*, specify the mutation map name created in Step 2. You can configure multiple DSCP-to-DSCP-mutation maps on an ingress port. **Note** To return to the default DSCP-to-DSCP-mutation map values, use the **no mls qos map dscp-mutation** *dscp-mutation-name* global configuration command.
Step 6	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 7	**show mls qos maps dscp-mutation** **Example:** Switch# **show mls qos maps dscp-mutation**	Verifies your entries.
Step 8	**copy running-config startup-config** **Example:** Switch# **copy-running-config startup-config**	(Optional) Saves your entries in the configuration file. **Note** To return a port to its non-trusted state, use the **no mls qos trust interface** configuration command. To return to the default DSCP-to-DSCP-mutation map values, use the **no mls qos map dscp-mutation** *dscp-mutation-name* global configuration command.

Configuring a QoS Policy

Configuring a QoS policy typically requires the following tasks:

- Classifying traffic into classes

- Configuring policies applied to those traffic classes

- Attaching policies to ports

These sections describe how to classify, police, and mark traffic. Depending on your network configuration, you must perform one or more of the modules in this section.

Classifying Traffic by Using ACLs

You can classify IP traffic by using IPv4 standard ACLS, IPv4 extended ACLs, or IPv6 ACLs.

You can classify non-IP traffic by using Layer 2 MAC ACLs.

Creating an IP Standard ACL for IPv4 Traffic

Before you begin

Before you perform this task, determine which access lists you will be using for your QoS configuration.

SUMMARY STEPS

1. **configure terminal**
2. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
3. **end**
4. **show access-lists**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**access-list** *access-list-number* {**deny**	**permit**} *source* [*source-wildcard*] **Example:** Switch(config)# **access-list 1 permit 192.2.255.0 10.1.1.255**	Creates 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. • Use the **permit** keyword to permit a certain type of traffic if the conditions are matched. Use the **deny** keyword to deny a certain type of traffic if conditions are matched. • 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. Place ones in the bit positions that you want to ignore.

	Command or Action	Purpose
		When you create an access list, remember that by default the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.
		Note To delete an access list, use the **no access-list** *access-list-number* global configuration command.
Step 3	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 4	**show access-lists** **Example:** Switch# **show access-lists**	Verifies your entries.
Step 5	**copy running-config startup-config** **Example:** Switch# **copy-running-config startup-config**	(Optional) Saves your entries in the configuration file.

Creating an IP Extended ACL for IPv4 Traffic

Before you begin

Before you perform this task, determine which access lists you will be using for your QoS configuration.

SUMMARY STEPS

1. **configure terminal**
2. **access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard*
3. **end**
4. **show access-lists**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:**	Enters global configuration mode.

	Command or Action	**Purpose**
	Switch# **configure terminal**	
Step 2	**access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard* **Example:** Switch(config)# **access-list 100 permit ip any any dscp 32**	Creates an IP extended ACL, repeating the command as many times as necessary. • For *access-list-number*, enter the access list number. The range is 100 to 199 and 2000 to 2699. • Use the **permit** keyword to permit a certain type of traffic if the conditions are matched. Use the **deny** keyword to deny a certain type of traffic if conditions are matched. • For *protocol*, enter the name or number of an IP protocol. Use the question mark (?) to see a list of available protocol keywords. • For *source*, enter the network or host from which the packet is being sent. You specify this by using dotted decimal notation, by using the **any** keyword as an abbreviation for *source* 0.0.0.0 *source-wildcard* 255.255.255.255, or by using the **host** keyword for *source* 0.0.0.0. • For *source-wildcard*, enter the wildcard bits by placing ones in the bit positions that you want to ignore. You specify the wildcard by using dotted decimal notation, by using the **any** keyword as an abbreviation for *source* 0.0.0.0 *source-wildcard* 255.255.255.255, or by using the **host** keyword for *source* 0.0.0.0. • For *destination*, enter the network or host to which the packet is being sent. You have the same options for specifying the *destination and destination-wildcard* as those described by *source* and *source-wildcard*. When creating an access list, remember that, by default, the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end. **Note** To delete an access list, use the **no access-list** *access-list-number* global configuration command.
Step 3	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

		Command or Action	**Purpose**
Step 4		show access-lists	Verifies your entries.
		Example:	
		Switch# **show access-lists**	
Step 5		**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.
		Example:	
		Switch# **copy-running-config startup-config**	

Creating an IPv6 ACL for IPv6 Traffic

Before you begin

Before you perform this task, determine which access lists you will be using for your QoS configuration.

SUMMARY STEPS

1. **configure terminal**
2. **ipv6 access-list** *access-list-name*
3. {**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*]
4. **end**
5. **show ipv6 access-list**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal**	Enters global configuration mode.
	Example:	
	Switch# **configure terminal**	
Step 2	**ipv6 access-list** *access-list-name*	Creates an IPv6 ACL and enters IPv6 access-list configuration mode.
	Example:	Accesses list names cannot contain a space or quotation mark or begin with a numeric.
	Switch(config)# **ipv6 access-list ipv6_Name_ACL**	**Note** To delete an access list, use the **no ipv6 access-list** *access-list-number* global configuration command.

	Command or Action	**Purpose**
Step 3	{**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*] **Example:** ```Switch(config-ipv6-acl)# permit ip host 10::1 host 11::2 host```	Enters **deny** or **permit** to specify whether to deny or permit the packet if conditions are matched. These are the conditions: For *protocol*, enter the 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. • The *source-ipv6-prefix/prefix-length* or *destination-ipv6-prefix/ prefix-length* is the 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 (see RFC 2373). • Enter **any** as an abbreviation for the IPv6 prefix ::/0. • For **host** *source-ipv6-address* or *destination-ipv6-address*, enter the source or destination IPv6 host address for which to set deny or permit conditions, specified in hexadecimal using 16-bit values between colons. • (Optional) For *operator*, specify an operand that compares the source or destination ports of the specified protocol. Operands are **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. • (Optional) The *port-number* is a decimal number from 0 to 65535 or the name of a TCP or UDP port. You can use TCP port names only when filtering TCP. You can use UDP port names only when filtering UDP. • (Optional) Enter **dscp** *value* to 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) Enter **fragments** to check noninitial fragments. This keyword is visible only if the protocol is IPv6. • (Optional) Enter **log** to cause a logging message to be sent to the console about the packet that matches the entry. Enter **log-input** to include the input interface in

	Command or Action	**Purpose**
		the log entry. Logging is supported only for router ACLs.
		• (Optional) Enter **routing** to specify that IPv6 packets be routed.
		• (Optional) Enter **sequence** *value* to specify the sequence number for the access list statement. The acceptable range is from 1 to 4294967295.
		• (Optional) Enter **time-range** *name* to specify the time range that applies to the deny or permit statement.
Step 4	**end** **Example:** Switch(config-ipv6-acl)# **end**	Returns to privileged EXEC mode.
Step 5	**show ipv6 access-list** **Example:** Switch# **show ipv6 access-list**	Verifies the access list configuration.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy-running-config startup-config**	(Optional) Saves your entries in the configuration file.

Creating a Layer 2 MAC ACL for Non-IP Traffic

Before you begin

Before you perform this task, determine that Layer 2 MAC access lists are required for your QoS configuration.

SUMMARY STEPS

1. **configure terminal**
2. **mac access-list extended** *name*
3. {**permit** | **deny**} {**host** *src-MAC-addr mask* | **any** | **host** *dst-MAC-addr* | *dst-MAC-addr mask*} [*type mask*]
4. **end**
5. **show access-lists** [*access-list-number* | *access-list-name*]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**mac access-list extended** *name* **Example:** Switch(config)# **mac access-list extended maclist1**	Creates a Layer 2 MAC ACL by specifying the name of the list. After entering this command, the mode changes to extended MAC ACL configuration. **Note** To delete an access list, use the **no mac access-list extended** *access-list-name* global configuration command.
Step 3	{**permit** | **deny**} {**host** *src-MAC-addr mask* | **any** | **host** *dst-MAC-addr* | *dst-MAC-addr mask*} [*type mask*] **Example:** Switch(config-ext-mac1) # **permit 0001.0000.0001 0.0.0 0002.0000.0001 0.0.0** Switch(config-ext-mac1) # **permit 0001.0000.0002 0.0.0 0002.0000.0002 0.0.0 xns-idp**	Specifies the type of traffic to permit or deny if the conditions are matched, entering the command as many times as necessary. • For *src-MAC-addr*, enter the MAC address of the host from which the packet is being sent. You specify this by using the hexadecimal format (H.H.H), by using the **any** keyword as an abbreviation for *source* 0.0.0, *source-wildcard* ffff.ffff.ffff, or by using 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 specify this by using the hexadecimal format (H.H.H), by using the **any** keyword as an abbreviation for *source* 0.0.0, *source-wildcard* ffff.ffff.ffff, or by using 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. When creating an access list, remember that, by default, the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.

	Command or Action	**Purpose**
Step 4	**end** **Example:** `Switch(config-ext-mac1)# **end**`	Returns to privileged EXEC mode.
Step 5	**show access-lists** [*access-list-number* | *access-list-name*] **Example:** `Switch# **show access-lists**`	Verifies your entries.

Step 6	**copy running-config startup-config**
	
	Example:
	
	`Switch# **copy-running-config**`
	`**startup-config**`
		(Optional) Saves your entries in the configuration file.

Classifying Traffic by Using Class Maps

You use the **class-map** global configuration command to name and to isolate a specific traffic flow (or class) from all other traffic. The 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, IP precedence values, or DSCP values. The match criterion is defined with one match statement entered within the class-map configuration mode.

Note   You can also create class maps during policy map creation by using the **class** policy-map configuration command.

SUMMARY STEPS

1. **configure terminal**
2. Use one of the following:

    - **access-list** *access-list-number* {**deny** \| **permit**} *source* [*source-wildcard*]
    - **access-list** *access-list-number* {**deny** \| **permit**} *protocol source* [*source-wildcard*] *destination* [*destination-wildcard*]
    - **ipv6 access-list** *access-list-name* {**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*]
    - **mac access-list extended** *name* {**permit** \| **deny**} {**host** *src-MAC-addr mask* \| **any** \| **host** *dst-MAC-addr* \| *dst-MAC-addr mask*} [*type mask*]

3. **class-map** [**match-all** \| **match-any**] *class-map-name*
4. **match** {**access-group** *acl-index-or-name* \| **ip dscp** *dscp-list* \| **ip precedence** *ip-precedence-list*}
5. **end**

6. **show class-map**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	Use one of the following: • **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] • **access-list** *access-list-number* {**deny** | **permit**} *protocol source* [*source-wildcard*] *destination* [*destination-wildcard*] • **ipv6 access-list** *access-list-name* {**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*] • **mac access-list extended** *name* {**permit** | **deny**} {**host** *src-MAC-addr mask* | **any** | **host** *dst-MAC-addr* | *dst-MAC-addr mask*} [*type mask*] **Example:** Switch(config)# **access-list 103 permit ip any any dscp 10**	Creates an IP standard or extended ACL, an IPv6 ACL for IP traffic, or a Layer 2 MAC ACL for non-IP traffic, repeating the command as many times as necessary. When creating an access list, remember that, by default, the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.
Step 3	**class-map** [**match-all** | **match-any**] *class-map-name* **Example:** Switch(config)# **class-map class1**	Creates a class map, and enters 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.

	Command or Action	**Purpose**
		If neither the **match-all** or **match-any** keyword is specified, the default is **match-all**.
		Note To delete an existing class map, use the **no class-map** [**match-all** | **match-any**] *class-map-name* global configuration command.
Step 4	**match** {**access-group** *acl-index-or-name* | **ip dscp** *dscp-list* | **ip precedence** *ip-precedence-list*} **Example:** Switch(config-cmap)# **match ip dscp 10 11 12**	Defines the match criterion to classify traffic. By default, no match criterion is defined. Only one match criterion 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 the ACL created in Step 2. • To filter IPv6 traffic with the **match access-group** command, create an IPv6 ACL, as described in Step 2. • For **ip dscp** *dscp-list*, enter a list of up to eight IP DSCP values to match against incoming packets. Separate each value with a space. The range is 0 to 63. • For **ip precedence** *ip-precedence-list*, enter a list of up to eight IP-precedence values to match against incoming packets. Separate each value with a space. The range is 0 to 7. **Note** To remove a match criterion, use the **no match** {**access-group** *acl-index-or-name* | **ip dscp** | **ip precedence**} class-map configuration command.
Step 5	**end** **Example:** Switch(config-cmap)# **end**	Returns to privileged EXEC mode.
Step 6	**show class-map** **Example:** Switch# **show class-map**	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy-running-config startup-config**	(Optional) Saves your entries in the configuration file.

Classifying Traffic by Using Class Maps and Filtering IPv6 Traffic

To apply the primary match criteria to only IPv4 traffic, use the **match protocol** command with the **ip** keyword. To apply the primary match criteria to only IPv6 traffic, use the **match protocol** command with the **ipv6** keyword.

SUMMARY STEPS

1. **configure terminal**
2. **class-map** {**match-all**} *class-map-name*
3. **match protocol** [*ip* / *ipv6*]
4. **match** {**ip dscp** *dscp-list* | **ip precedence** *ip-precedence-list*}
5. **end**
6. **show class-map**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**class-map** {**match-all**} *class-map-name* **Example:** Switch(config)# **class-map cm-1**	Creates a class map, and enters class-map configuration mode. By default, no class maps are defined. When you use the **match protocol** command, only the **match-all** keyword is supported. • For *class-map-name*, specify the name of the class map. If neither the **match-all** or **match-any** keyword is specified, the default is **match-all**. **Note** To delete an existing class map, use the **no class-map** [**match-all**	**match-any**] *class-map-name* global configuration command.
Step 3	**match protocol** [*ip* / *ipv6*] **Example:** Switch(config-cmap)# **match protocol ip**	(Optional) Specifies the IP protocol to which the class map applies: • Use the argument *ip* to specify IPv4 traffic and *ipv6* to specify IPv6 traffic. • When you use the **match protocol** command, only the **match-all** keyword is supported for the **class-map** command.

	Command or Action	Purpose
		Note You can also match the following protocols: ARP, CDP, and HTTP.
Step 4	**match** {**ip dscp** *dscp-list*	**ip precedence** *ip-precedence-list*} **Example:** Switch(config-cmap)# **match ip dscp 10**	Defines the match criterion to classify traffic. By default, no match criterion is defined. • For **ip dscp** *dscp-list*, enter a list of up to eight IP DSCP values to match against incoming packets. Separate each value with a space. The range is 0 to 63. • For **ip precedence** *ip-precedence-list*, enter a list of up to eight IP-precedence values to match against incoming packets. Separate each value with a space. The range is 0 to 7. **Note** To remove a match criterion, use the no match {**access-group** *acl-index-or-name*	**ip dscp**	**ip precedence**} class-map configuration command.
Step 5	**end** **Example:** Switch(config-cmap)# **end**	Returns to privileged EXEC mode.
Step 6	**show class-map** **Example:** Switch# **show class-map**	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy-running-config startup-config**	(Optional) Saves your entries in the configuration file.

Classifying, Policing, and Marking Traffic on Physical Ports by Using Policy Maps

You can configure a policy map on a physical port that specifies which traffic class to act on. Actions can include trusting the CoS, DSCP, or IP precedence values in the traffic class; setting a specific DSCP or IP precedence value in the traffic class; and specifying the traffic bandwidth limitations for each matched traffic class (policer) and the action to take when the traffic is out of profile (marking).

A policy map also has these characteristics:

• A policy map can contain multiple class statements, each with different match criteria and policers.

• A policy map can contain a predefined default traffic class explicitly placed at the end of the map.

• A separate policy-map class can exist for each type of traffic received through a port.

Follow these guidelines when configuring policy maps on physical ports:

• You can attach only one policy map per ingress port.

• If you configure the IP-precedence-to-DSCP map by using the **mls qos map ip-prec-dscp** *dscp1...dscp8* global configuration command, the settings only affect packets on ingress interfaces that are configured to trust the IP precedence value. In a policy map, if you set the packet IP precedence value to a new value by using the **set ip precedence** *new-precedence* policy-map class configuration command, the egress DSCP value is not affected by the IP-precedence-to-DSCP map. If you want the egress DSCP value to be different than the ingress value, use the **set dscp new-dscp** policy-map class configuration command.

• If you enter or have used the **set ip dscp** command, the switch changes this command to **set dscp** in its configuration.

• You can use the **set ip precedence** or the **set precedence** policy-map class configuration command to change the packet IP precedence value. This setting appears as set ip precedence in the switch configuration.

• You can configure a separate second-level policy map for each class defined for the port. The second-level policy map specifies the police action to take for each traffic class.

• A policy-map and a port trust state can both run on a physical interface. The policy-map is applied before the port trust state.

• When you configure a default traffic class by using the **class class-default** policy-map configuration command, unclassified traffic (traffic that does not meet the match criteria specified in the traffic classes) is treated as the default traffic class (class-default).

SUMMARY STEPS

1.   **configure terminal**
2.   **class-map** [**match-all** | **match-any**] *class-map-name*
3.   **policy-map** *policy-map-name*
4.   **class** [*class-map-name* | **class-default**]
5.   **trust** [**cos** | **dscp** | **ip-precedence**]
6.   **set** {**dscp** *new-dscp* | **ip precedence** *new-precedence*}
7.   **police** *rate-bps burst-byte* [**exceed-action** {**drop** | **policed-dscp-transmit**}]
8.   **exit**
9.   **exit**
10.   **interface** *interface-id*
11.   **service-policy input** *policy-map-name*
12.   **end**
13.   **show policy-map** [*policy-map-name* [**class** *class-map-name*]]
14.   **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 2	**class-map** [**match-all** | **match-any**] *class-map-name* Example: Switch(config)# **class-map ipclass1**	Creates a class map, and enters 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 neither the **match-all** or **match-any** keyword is specified, the default is **match-all**.
Step 3	**policy-map** *policy-map-name* Example: Switch(config-cmap)# **policy-map flowit**	Creates a policy map by entering the policy map name, and enters policy-map configuration mode. By default, no policy maps are defined. The default behavior of a policy map is to set the DSCP to 0 if the packet is an IP packet and to set the CoS to 0 if the packet is tagged. No policing is performed. **Note** To delete an existing policy map, use the **no policy-map** *policy-map-name* global configuration command.
Step 4	**class** [*class-map-name* | **class-default**] Example: Switch(config-pmap)# **class ipclass1**	Defines a traffic classification, and enters policy-map class configuration mode. By default, no policy map class-maps are defined. If a traffic class has already been defined by using the **class-map** global configuration command, specify its name for *class-map-name* in this command. A **class-default** traffic class is pre-defined and can be added to any policy. It is always placed at the end of a policy map. With an implied **match any** included in the

	Command or Action	**Purpose**
		class-default class, all packets that have not already matched the other traffic classes will match **class-default**.
		Note To delete an existing class map, use the **no class** *class-map-name* policy-map configuration command.
Step 5	**trust** [**cos** | **dscp** | **ip-precedence**] **Example:** Switch(config-pmap-c)# **trust dscp**	Configures the trust state, which QoS uses to generate a CoS-based or DSCP-based QoS label. This command is mutually exclusive with the **set** command within the same policy map. If you enter the **trust** command, go to Step 6. By default, the port is not trusted. If no keyword is specified when the command is entered, the default is **dscp**. The keywords have these meanings: • **cos**—QoS derives the DSCP value by using the received or default port CoS value and the CoS-to-DSCP map. • **dscp**—QoS derives the DSCP value by using the DSCP value from the ingress packet. For non-IP packets that are tagged, QoS derives the DSCP value by using the received CoS value; for non-IP packets that are untagged, QoS derives the DSCP value by using the default port CoS value. In either case, the DSCP value is derived from the CoS-to-DSCP map. • **ip-precedence**—QoS derives the DSCP value by using the IP precedence value from the ingress packet and the IP-precedence-to-DSCP map. For non-IP packets that are tagged, QoS derives the DSCP value by using the received CoS value; for non-IP packets that are untagged, QoS derives the DSCP value by using the default port CoS value. In either case, the DSCP value is derived from the CoS-to-DSCP map. **Note** To return to the untrusted state, use the **no trust** policy-map configuration command
Step 6	**set** {**dscp** *new-dscp* | **ip precedence** *new-precedence*} **Example:** Switch(config-pmap-c)# **set dscp 45**	Classifies IP traffic by setting a new value in the packet. • For **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.

	Command or Action	Purpose
		Note To remove an assigned DSCP or IP precedence value, use the **no set** {**dscp** *new-dscp* | **ip** precedence *new-precedence*} policy-map configuration command.
Step 7	**police** *rate-bps burst-byte* [**exceed-action** {**drop** | **policed-dscp-transmit**}] **Example:** Switch(config-pmap-c)# **police 100000 80000 drop**	Defines a policer for the classified traffic. By default, no policer is defined. • For *rate-bps,* specify average traffic rate in bits per second (b/s). The range is 8000 to 10000000000. • For *burst-byte,* specify the normal burst size in bytes. The range is 8000 to 1000000. • (Optional) Specifies the action to take when the rates are exceeded. Use the **exceed-action drop** keywords to drop the packet. Use the **exceed-action policed-dscp-transmit** keywords to mark down the DSCP value (by using the policed-DSCP map) and to send the packet. **Note** To remove an existing policer, use the **no police** *rate-bps burst-byte* [**exceed-action** {**drop** | **policed-dscp-transmit**}] policy-map configuration command.
Step 8	**exit** **Example:** Switch(config-pmap-c)# **exit**	Returns to policy map configuration mode.
Step 9	**exit** **Example:** Switch(config-pmap)# **exit**	Returns to global configuration mode.
Step 10	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet 2/0/1**	Specifies the port to attach to the policy map, and enters interface configuration mode. Valid interfaces include physical ports.
Step 11	**service-policy input** *policy-map-name* **Example:** Switch(config-if)# **service-policy**	Specifies the policy-map name, and applies it to an ingress port. Only one policy map per ingress port is supported.

	Command or Action	Purpose
	`input flowit`	**Note** To remove the policy map and port association, use the **no service-policy** *input policy-map-name* interface configuration command.
Step 12	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 13	**show policy-map** [*policy-map-name* [**class** *class-map-name*]] **Example:** Switch# **show policy-map**	Verifies your entries.
Step 14	**copy running-config startup-config** **Example:** Switch# **copy-running-config startup-config**	(Optional) Saves your entries in the configuration file.

Classifying, Policing, and Marking Traffic on SVIs by Using Hierarchical Policy Maps

You can configure hierarchical policy maps on SVIs, but not on other types of interfaces. Hierarchical policing combines the VLAN- and interface-level policy maps to create a single policy map.

You can configure hierarchical policy maps on SVIs, but not on other types of interfaces. Hierarchical policing combines the VLAN- and interface-level policy maps to create a single policy map.

On an SVI, the VLAN-level policy map specifies which traffic class to act on. Actions can include trusting the CoS, DSCP, or IP precedence values or setting a specific DSCP or IP precedence value in the traffic class. Use the interface-level policy map to specify the physical ports that are affected by individual policers.

You can configure hierarchical policy maps that filter IPv4 and IPv6 traffic.

Follow these guidelines when configuring hierarchical policy maps:

- Before configuring a hierarchical policy map, you must enable VLAN-based QoS on the physical ports that are to be specified at the interface level of the policy map.

- You can attach only one policy map per ingress port or SVI.

- A policy map can contain multiple class statements, each with different match criteria and actions.

- A separate policy-map class can exist for each type of traffic received on the SVI.

- In a switch stack, you cannot use the **match input-interface** class-map configuration command to specify interfaces across stack members in a policy-map class.

- A policy-map and a port trust state can both run on a physical interface. The policy-map is applied before the port trust state.

- If you configure the IP-precedence-to-DSCP map by using the **mls qos map ip-prec-dscp** *dscp1...dscp8* global configuration command, the settings only affect packets on ingress interfaces that are configured to trust the IP precedence value. In a policy map, if you set the packet IP precedence value to a new value by using the **set ip precedence** *new-precedence* policy-map class configuration command, the egress DSCP value is not affected by the IP-precedence-to-DSCP map. If you want the egress DSCP value to be different than the ingress value, use the **set dscp** *new-dscp* policy-map class configuration command.

- If you enter or have used the **set ip dscp** command, the switch changes this command to **set dscp** in its configuration. If you enter the **set ip dscp** command, this setting appears as **set dscp** in the switch configuration.

- You can use the **set ip precedence** or the **set precedence** policy-map class configuration command to change the packet IP precedence value. This setting appears as **set ip precedence** in the switch configuration.

- If VLAN-based QoS is enabled, the hierarchical policy map supersedes the previously configured port-based policy map.

- The hierarchical policy map is attached to the SVI and affects all traffic in the VLAN. The actions specified in the VLAN-level policy map affect the traffic belonging to the SVI. The police action on the port-level policy map affects the ingress traffic on the affected physical interfaces.

- When configuring a hierarchical policy map on trunk ports, the VLAN ranges must not overlap. If the ranges overlap, the actions specified in the policy map affect the incoming and outgoing traffic on the overlapped VLANs.

- Aggregate policers are not supported in hierarchical policy maps.

- When VLAN-based QoS is enabled, the switch supports VLAN-based features, such as the VLAN map.

- You can configure a hierarchical policy map only on the primary VLAN of a private VLAN.

- When you enable VLAN-based QoS and configure a hierarchical policy map in a switch stack, these automatic actions occur when the stack configuration changes:

  - When a new active stack is selected, the active stack reenables and reconfigures these features on all applicable interfaces on the active stack.

  - When a stack member is added, the active stack reenables and reconfigures these features on all applicable ports on the stack member.

  - When you merge switch stacks, the new active stack reenables and reconfigures these features on the switches in the new stack.

  - When the switch stack divides into two or more switch stacks, the active stack in each switch stack re-enables and reconfigures these features on all applicable interfaces on the stack members, including the active stack.

  - When you configure a default traffic class by using the **class class-default** policy-map configuration command, unclassified traffic (traffic that does not meet the match criteria specified in the traffic classes) is treated as default traffic class (**class-default**).

SUMMARY STEPS

1. **configure terminal**
2. **class-map** [**match-all** | **match-any**] *class-map-name*
3. **match** {**access-group** *acl-index-or-name* | **ip dscp** *dscp-list* | **ip precedence** *ip-precedence-list*}
4. **match protocol** [*ip* / *ipv6*]
5. **exit**
6. **exit**
7. **class-map** [**match-all** | **match-any**] *class-map-name*
8. **match input-interface** *interface-id-list*
9. **exit**
10. **exit**
11. **policy-map** *policy-map-name*
12. **class-map** *class-map-name*
13. **police** *rate-bps burst-byte* [**exceed-action** {**drop** | **policed-dscp-transmit**}]
14. **exit**
15. **exit**
16. **policy-map** *policy-map-name*
17. **class** [*class-map-name* | **class-default**]
18. **trust** [**cos** | **dscp** | **ip-precedence**]
19. **set** {**dscp** *new-dscp* | **ip precedence** *new-precedence*}
20. **service-policy** *policy-map-name*
21. **exit**
22. **exit**
23. **interface** *interface-id*
24. **service-policy input** *policy-map-name*
25. **end**
26. **show policy-map** [*policy-map-name* [**class** *class-map-name*]] or **show mls qos vlan-based**
27. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**class-map** [**match-all**	**match-any**] *class-map-name* **Example:** Switch(config)# **class-map cm-1**	Creates a VLAN-level class map, and enters 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.

	Command or Action	**Purpose**
		• (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 neither the **match-all** or **match-any** keyword is specified, the default is **match-all**.
		Because only one **match** command per class map is supported, the **match-all** and **match-any** keywords function the same.
Step 3	**match** {**access-group** *acl-index-or-name* | **ip dscp** *dscp-list* | **ip precedence** *ip-precedence-list*} **Example:** Switch(config-cmap)# **match ip dscp 10**	Defines the match criterion to classify traffic. By default, no match criterion is defined. Only one match criterion 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 the ACL. • For **ip dscp** *dscp-list*, enter a list of up to eight IP DSCP values to match against incoming packets. Separate each value with a space. The range is 0 to 63. • For **ip precedence** *ip-precedence-list*, enter a list of up to eight IP-precedence values to match against incoming packets. Separate each value with a space. The range is 0 to 7.
Step 4	**match protocol** [*ip* | *ipv6*] **Example:** Switch(config-cmap)# **match protocol ipv6**	(Optional) Specifies the IP protocol to which the class map applies. • Use the argument *ip* to specify IPv4 traffic, and *ipv6* to specify IPv6 traffic. • When you use the **match protocol** command, only the **match-all** keyword is supported for the first level class map. You can use the **match protocol** command with the **match ip dscp** or **match precedence** commands, but not with the **match access-group** command.
Step 5	**exit** **Example:**	Returns to class-map configuration mode.

	Command or Action	**Purpose**
	Switch(config-cmap)# **exit**	
Step 6	**exit** **Example:** Switch(config)# **exit**	Returns to global configuration mode.
Step 7	**class-map** [**match-all** | **match-any**] *class-map-name* **Example:** Switch(config)# **class-map match-all cm-2**	Creates an interface-level class map, and enters 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 neither the **match-all** or **match-any** keyword is specified, the default is **match-all**. Because only one **match** command per class map is supported, the **match-all** and **match-any** keywords function the same.
Step 8	**match input-interface** *interface-id-list* **Example:** Switch(config-cmap)# **match input-interface gigabitethernet 3/0/1-3/0/2**	Specifies the physical ports on which the interface-level class map acts. You can specify up to six ports as follows: - A single port (counts as one entry) - A list of ports separated by a space (each port counts as an entry) - A range of ports separated by a hyphen (counts as two entries) This command can only be used in the child-level policy map and must be the only match condition in the child-level policy map.
Step 9	**exit** **Example:**	Returns to class-map configuration mode.

	Command or Action	**Purpose**
	Switch(config-cmap)# **exit**	
Step 10	**exit** **Example:** Switch(config)# **exit**	Returns to global configuration mode.
Step 11	**policy-map** *policy-map-name* **Example:** Switch# **policy-map port-plcmap**	Creates an interface-level policy map by entering the policy-map name, and enters policy-map configuration mode. By default, no policy maps are defined, and no policing is performed. To delete an existing policy map, use the **no policy-map** *policy-map-name* global configuration command. To delete an existing class map, use the **no class** *class-map-name* policy-map configuration command. To return to the untrusted state in a policy map, use the **no trust** policy-map configuration command. To remove an assigned DSCP or IP precedence value, use the **no set** {**dscp** *new-dscp*	**ip precedence** *new-precedence*} policy-map configuration command.
Step 12	**class-map** *class-map-name* **Example:** Switch(config-pmap)# **class cm-interface-1**	Defines an interface-level traffic classification, and enters policy-map configuration mode. By default, no policy-map class-maps are defined. If a traffic class has already been defined by using the **class-map** global configuration command, specify its name for *class-map-name* in this command.
Step 13	**police** *rate-bps burst-byte* [**exceed-action** {**drop**	**policed-dscp-transmit**}] **Example:** Switch(config-pmap-c)# **police 900000 9000 exceed-action policed-dscp-transmit**	Defines an individual policer for the classified traffic. By default, no policer is defined. • For *rate-bps,* specify average traffic rate in bits per second (b/s). The range is 8000 to 10000000000. • For *burst-byte,* specify the normal burst size in bytes. The range is 8000 to 1000000. • (Optional) Specifies the action to take when the rates are exceeded. Use the **exceed-action drop** keywords to drop the packet. Use the **exceed-action policed-dscp-transmit** keywords to mark down the DSCP value (by using the policed-DSCP map) and to send the packet.

	Command or Action	Purpose
		To remove an existing policer in an interface-level policy map, use the **no police** *rate-bps burst-byte* [**exceed-action** {**drop**	**policed-dscp-transmit**}] policy-map configuration command. To remove the hierarchical policy map and port associations, use the **no service-policy input** *policy-map-name* interface configuration command.
Step 14	**exit** **Example:** Switch(config-pmap-c)# **exit**	Returns to policy-map configuration mode.
Step 15	**exit** **Example:** Switch(config-pmap)# **exit**	Returns to global configuration mode.
Step 16	**policy-map** *policy-map-name* **Example:** Switch(config)# **policy-map vlan-plcmap**	Creates a VLAN-level policy map by entering the policy-map name, and enters policy-map configuration mode. By default, no policy maps are defined. The default behavior of a policy map is to set the DSCP to 0 if the packet is an IP packet and to set the CoS to 0 if the packet is tagged. No policing is performed.
Step 17	**class** [*class-map-name*	**class-default**] **Example:** Switch(config-pmap)# **class cm-1**	Defines a VLAN-level traffic classification, and enters policy-map class configuration mode. By default, no policy-map class-maps are defined. If a traffic class has already been defined by using the **class-map** global configuration command, specify its name for **class-map-name** in this command. A **class-default** traffic class is pre-defined and can be added to any policy. It is always placed at the end of a policy map. With an implied **match any** included in the **class-default** class, all packets that have not already matched the other traffic classes will match **class-default**.
Step 18	**trust** [**cos**	**dscp**	**ip-precedence**] **Example:** Switch(config-pmap-c)# **trust dscp**	Configures the trust state, which QoS uses to generate a CoS-based or DSCP-based QoS label. This command is mutually exclusive with the **set** command within the same policy map. If you enter the **trust** command, omit Step 18. By default, the port is not trusted. If no keyword is specified when the command is entered, the default is **dscp**.

	Command or Action	**Purpose**
		The keywords have these meanings:
		• **cos**—QoS derives the DSCP value by using the received or default port CoS value and the CoS-to-DSCP map.
		• **dscp**—QoS derives the DSCP value by using the DSCP value from the ingress packet. For non-IP packets that are tagged, QoS derives the DSCP value by using the received CoS value; for non-IP packets that are untagged, QoS derives the DSCP value by using the default port CoS value. In either case, the DSCP value is derived from the CoS-to-DSCP map.
		• **ip-precedence**—QoS derives the DSCP value by using the IP precedence value from the ingress packet and the IP-precedence-to-DSCP map. For non-IP packets that are tagged, QoS derives the DSCP value by using the received CoS value; for non-IP packets that are untagged, QoS derives the DSCP value by using the default port CoS value. In either case, the DSCP value is derived from the CoS-to-DSCP map.
Step 19	**set** {**dscp** *new-dscp* | **ip precedence** *new-precedence*} **Example:** `Switch(config-pmap-c)# set dscp 10`	Classifies IP traffic by setting a new value in the packet. • For **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.
Step 20	**service-policy** *policy-map-name* **Example:** `Switch(config-pmap-c)# service-policy port-plcmap-1`	Specifies the interface-level policy-map name (from Step 10) and associate it with the VLAN-level policy map. If the VLAN-level policy map specifies more than one class, each class can have a different **service-policy** *policy-map-name* command.
Step 21	**exit** **Example:** `Switch(config-pmap-c)# exit`	Returns to policy-map configuration mode.
Step 22	**exit** **Example:** `Switch(config-pmap)# exit`	Returns to global configuration mode.

	Command or Action	Purpose
Step 23	**interface** *interface-id* **Example:** Switch(config)# **interface vlan 10**	Specifies the SVI to which to attach the hierarchical policy map, and enters interface configuration mode.
Step 24	**service-policy input** *policy-map-name* **Example:** Switch(config-if)# **service-policy input vlan-plcmap**	Specifies the VLAN-level policy-map name, and applies it to the SVI. Repeat the previous step and this command to apply the policy map to other SVIs. If the hierarchical VLAN-level policy map has more than one interface-level policy map, all class maps must be configured to the same VLAN-level policy map specified in the **service-policy** *policy-map-name* command. **Note** To remove the hierarchical policy map and port associations, use the **no service-policy input** *policy-map-name* interface configuration command.
Step 25	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 26	**show policy-map** [*policy-map-name* [**class** *class-map-name*]] or **show mls qos vlan-based** **Example:** Switch# **show mls qos vlan-based**	Verifies your entries.
Step 27	**copy running-config startup-config** **Example:** Switch# **copy-running-config startup-config**	(Optional) Saves your entries in the configuration file.

Classifying, Policing, and Marking Traffic by Using Aggregate Policers

By using an aggregate policer, you can create a policer that is shared by multiple traffic classes within the same policy map. However, you cannot use the aggregate policer across different policy maps or ports.

You can configure aggregate policers only in nonhierarchical policy maps on physical ports.

SUMMARY STEPS

    1.   **configure terminal**

> **2.** **mls qos aggregate-policer** *aggregate-policer-name rate-bps burst-byte* **exceed-action** {**drop** |
> **policed-dscp-transmit**}
> **3.** **class-map** [**match-all** | **match-any**] *class-map-name*
> **4.** **policy-map** *policy-map-name*
> **5.** **class** [*class-map-name* | **class-default**]
> **6.** **police aggregate** *aggregate-policer-name*
> **7.** **exit**
> **8.** **interface** *interface-id*
> **9.** **service-policy input** *policy-map-name*
> **10.** **end**
> **11.** **show mls qos aggregate-policer** [*aggregate-policer-name*]
> **12.** **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**mls qos aggregate-policer** *aggregate-policer-name rate-bps burst-byte* **exceed-action** {**drop**	**policed-dscp-transmit**} **Example:** Switch(config)# **mls qos aggregate-police transmit1 48000 8000 exceed-action policed-dscp-transmit**	Defines the policer parameters that can be applied to multiple traffic classes within the same policy map. By default, no aggregate policer is defined. • 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 10000000000. • For *burst-byte,* specify the normal burst size in bytes. The range is 8000 to 1000000. • Specifies the action to take when the rates are exceeded. Use the **exceed-action drop** keywords to drop the packet. Use the **exceed-action policed-dscp-transmit** keywords to mark down the DSCP value (by using the policed-DSCP map) and to send the packet.
Step 3	**class-map** [**match-all**	**match-any**] *class-map-name* **Example:** Switch(config)# **class-map ipclass1**	Creates a class map to classify traffic as necessary.

	Command or Action	**Purpose**
Step 4	**policy-map** *policy-map-name* **Example:** Switch(config-cmap)# **policy-map aggflow1**	Creates a policy map by entering the policy map name, and enters policy-map configuration mode.
Step 5	**class** [*class-map-name* | **class-default**] **Example:** Switch(config-cmap-p)# **class ipclass1**	Defines a traffic classification, and enters policy-map class configuration mode.
Step 6	**police aggregate** *aggregate-policer-name* **Example:** Switch(configure-cmap-p)# **police aggregate transmit1**	Applies an aggregate policer to multiple classes in the same policy map. For *aggregate-policer-name*, enter the name specified in Step 2. To remove the specified aggregate policer from a policy map, use the **no police aggregate** *aggregate-policer-name* policy map configuration command. To delete an aggregate policer and its parameters, use the **no mls qos aggregate-policer** *aggregate-policer-name* global configuration command.
Step 7	**exit** **Example:** Switch(configure-cmap-p)# **exit**	Returns to global configuration mode.
Step 8	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet 2/0/1**	Specifies the port to attach to the policy map, and enters interface configuration mode. Valid interfaces include physical ports.
Step 9	**service-policy input** *policy-map-name* **Example:** Switch(config-if)# service-policy input aggflow1	Specifies the policy-map name, and applies it to an ingress port. Only one policy map per ingress port is supported.
Step 10	**end** **Example:** Switch(configure-if)# **end**	Returns to privileged EXEC mode.

	Command or Action	**Purpose**
Step 11	**show mls qos aggregate-policer** [*aggregate-policer-name*] **Example:** Switch# **show mls qos aggregate-policer transmit1**	Verifies your entries.
Step 12	**copy running-config startup-config** **Example:** Switch# **copy-running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring DSCP Maps

Configuring the CoS-to-DSCP Map

You use the CoS-to-DSCP map to map CoS values in incoming packets to a DSCP value that QoS uses internally to represent the priority of the traffic.

Beginning in privileged EXEC mode, follow these steps to modify the CoS-to-DSCP map. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos map cos-dscp** *dscp1...dscp8*
3. **end**
4. **show mls qos maps cos-dscp**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**mls qos map cos-dscp** *dscp1...dscp8* **Example:** Switch(config)# **mls qos map cos-dscp 10 15 20 25 30 35 40 45**	Modifies the CoS-to-DSCP map. For *dscp1...dscp8*, enter eight DSCP values that correspond to CoS values 0 to 7. Separate each DSCP value with a space. The DSCP range is 0 to 63. **Note** To return to the default map, use the **no mls qos cos-dscp** global configuration command.

	Command or Action	Purpose
Step 3	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 4	**show mls qos maps cos-dscp** **Example:** Switch# **show mls qos maps cos-dscp**	Verifies your entries.
Step 5	**copy running-config startup-config** **Example:** Switch# **copy-running-config** **startup-config**	(Optional) Saves your entries in the configuration file.

Configuring the IP-Precedence-to-DSCP Map

You use the IP-precedence-to-DSCP map to map IP precedence values in incoming packets to a DSCP value that QoS uses internally to represent the priority of the traffic.

Beginning in privileged EXEC mode, follow these steps to modify the IP-precedence-to-DSCP map. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos map ip-prec-dscp** *dscp1...dscp8*
3. **end**
4. **show mls qos maps ip-prec-dscp**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**mls qos map ip-prec-dscp** *dscp1...dscp8* **Example:** Switch(config)# **mls qos map**	Modifies the IP-precedence-to-DSCP map. For *dscp1...dscp8*, enter eight DSCP values that correspond to the IP precedence values 0 to 7. Separate each DSCP value with a space.

	Command or Action	Purpose
	`ip-prec-dscp 10 15 20 25 30 35 40 45`	The DSCP range is 0 to 63. **Note** To return to the default map, use the **no mls qos ip-prec-dscp** global configuration command.
Step 3	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 4	**show mls qos maps ip-prec-dscp** **Example:** Switch# **show mls qos maps ip-prec-dscp**	Verifies your entries.
Step 5	**copy running-config startup-config** **Example:** Switch# **copy-running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring the Policed-DSCP Map

You use the policed-DSCP map to mark down a DSCP value to a new value as the result of a policing and marking action.

The default policed-DSCP map is a null map, which maps an incoming DSCP value to the same DSCP value.

Beginning in privileged EXEC mode, follow these steps to modify the policed-DSCP map. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos map policed-dscp** *dscp-list* **to** *mark-down-dscp*
3. **end**
4. **show mls qos maps policed-dscp**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**mls qos map policed-dscp** *dscp-list* **to** *mark-down-dscp* **Example:** Switch(config)# **mls qos map policed-dscp 50 51 52 53 54 55 56 57 to 0**	Modifies the policed-DSCP map. • For *dscp-list*, enter up to eight DSCP values separated by spaces. Then enter the **to** keyword. • For *mark-down-dscp*, enter the corresponding policed (marked down) DSCP value. **Note** To return to the default map, use the **no mls qos policed-dscp** global configuration command.
Step 3	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 4	**show mls qos maps policed-dscp** **Example:** Switch(config)# **show mls qos maps policed-dscp**	Verifies your entries.
Step 5	**copy running-config startup-config** **Example:** Switch#	(Optional) Saves your entries in the configuration file.

Configuring the DSCP-to-CoS Map

You use the DSCP-to-CoS map to generate a CoS value, which is used to select one of the four egress queues.

Beginning in privileged EXEC mode, follow these steps to modify the DSCP-to-CoS map. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos map dscp-cos** *dscp-list* **to** *cos*
3. **end**
4. **show mls qos maps dscp-to-cos**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal**	Enters global configuration mode.
	Example:	
	Switch# **configure terminal**	
Step 2	**mls qos map dscp-cos** *dscp-list* **to** *cos*	Modifies the DSCP-to-CoS map.
	Example:	• For *dscp-list*, enter up to eight DSCP values separated by spaces. Then enter the **to** keyword.
	Switch# **mls qos map dscp-cos 0 8 16 24 32 40 48 50 to 0**	• For *cos*, enter the CoS value to which the DSCP values correspond.
		The DSCP range is 0 to 63; the CoS range is 0 to 7.
		Note To return to the default map, use the **no mls qos dscp-cos** global configuration command.
Step 3	**end**	Returns to privileged EXEC mode.
	Example:	
	Switch(config)# **end**	
Step 4	**show mls qos maps dscp-to-cos**	Verifies your entries.
	Example:	
	Switch# **show mls qos maps dscp-to-cos**	
Step 5	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.
	Example:	
	Switch# **copy-running-config startup-config**	

Configuring the DSCP-to-DSCP-Mutation Map

If two QoS domains have different DSCP definitions, use the DSCP-to-DSCP-mutation map to translate one set of DSCP values to match the definition of another domain. You apply the DSCP-to-DSCP-mutation map to the receiving port (ingress mutation) at the boundary of a QoS administrative domain.

With ingress mutation, the new DSCP value overwrites the one in the packet, and QoS applies the new value to the packet. The switch sends the packet out the port with the new DSCP value.

You can configure multiple DSCP-to-DSCP-mutation maps on an ingress port. The default DSCP-to-DSCP-mutation map is a null map, which maps an incoming DSCP value to the same DSCP value.

Beginning in privileged EXEC mode, follow these steps to modify the DSCP-to-DSCP-mutation map. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos map dscp-mutation** *dscp-mutation-name in-dscp* **to** *out-dscp*
3. **interface** *interface-id*
4. **mls qos trust dscp**
5. **mls qos dscp-mutation** *dscp-mutation-name*
6. **end**
7. **show mls qos maps dscp-mutation**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**mls qos map dscp-mutation** *dscp-mutation-name in-dscp* **to** *out-dscp* **Example:** Switch(config)# **mls qos map dscp-mutation mutation1 1 2 3 4 5 6 7 to 0**	Modifies the DSCP-to-DSCP-mutation map. • For *dscp-mutation-name*, enter the mutation map name. You can create more than one map by specifying a new name. • For *in-dscp*, enter up to eight DSCP values separated by spaces. Then enter the **to** keyword. • For *out-dscp*, enter a single DSCP value. The DSCP range is 0 to 63. **Note** To return to the default map, use the **no mls qos dscp-mutation** *dscp-mutation-name* global configuration command.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	Specifies the port to which to attach the map, and enters interface configuration mode. Valid interfaces include physical ports.
Step 4	**mls qos trust dscp** **Example:** Switch(config-if)# **mls qos trust dscp**	Configures the ingress port as a DSCP-trusted port. By default, the port is not trusted.

	Command or Action	**Purpose**
Step 5	**mls qos dscp-mutation** *dscp-mutation-name* **Example:** Switch(config-if)# **mls qos dscp-mutation mutation1**	Applies the map to the specified ingress DSCP-trusted port. For *dscp-mutation-name*, enter the mutation map name specified in Step 2.
Step 6	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 7	**show mls qos maps dscp-mutation** **Example:** Switch# **show mls qos maps dscp-mutation**	Verifies your entries.
Step 8	**copy running-config startup-config** **Example:** Switch# **copy-running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Ingress Queue Characteristics

Depending on the complexity of your network and your QoS solution, you might need to perform all of the tasks in the next modules. You need to make decisions about these characteristics:

- Which packets are assigned (by DSCP or CoS value) to each queue?

- What drop percentage thresholds apply to each queue, and which CoS or DSCP values map to each threshold?

- How much of the available buffer space is allocated between the queues?

- How much of the available bandwidth is allocated between the queues?

- Is there traffic (such as voice) that should be given high priority?

Configuration Guidelines

Follow these guidelines when the expedite queue is enabled or the egress queues are serviced based on their SRR weights:

- If the egress expedite queue is enabled, it overrides the SRR shaped and shared weights for queue 1.

- If the egress expedite queue is disabled and the SRR shaped and shared weights are configured, the shaped mode overrides the shared mode for queue 1, and SRR services this queue in shaped mode.

• If the egress expedite queue is disabled and the SRR shaped weights are not configured, SRR services this queue in shared mode.

Mapping DSCP or CoS Values to an Ingress Queue and Setting WTD Thresholds

You can prioritize traffic by placing packets with particular DSCPs or CoSs into certain queues and adjusting the queue thresholds so that packets with lower priorities are dropped.

Beginning in privileged EXEC mode, follow these steps to map DSCP or CoS values to an ingress queue and to set WTD thresholds. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. Use one of the following:
   - **mls qos srr-queue input dscp-map queue** *queue-id* **threshold** *threshold-id dscp1...dscp8*
   - **mls qos srr-queue input cos-map queue** *queue-id* **threshold** *threshold-id cos1...cos8*
3. **mls qos srr-queue input threshold** *queue-id threshold-percentage1 threshold-percentage2*
4. **end**
5. **show mls qos maps**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	Use one of the following: • **mls qos srr-queue input dscp-map queue** *queue-id* **threshold** *threshold-id dscp1...dscp8* • **mls qos srr-queue input cos-map queue** *queue-id* **threshold** *threshold-id cos1...cos8* **Example:** Switch(config)# **mls qos srr-queue input dscp-map queue 1 threshold 2 20 21 22 23 24 25 26**	Maps DSCP or CoS values to an ingress queue and to a threshold ID. By default, DSCP values 0–39 and 48–63 are mapped to queue 1 and threshold 1. DSCP values 40–47 are mapped to queue 2 and threshold 1. By default, CoS values 0–4, 6, and 7 are mapped to queue 1 and threshold 1. CoS value 5 is mapped to queue 2 and threshold 1. • For *queue-id*, the range is 1 to 2. • For *threshold-id*, the range is 1 to 3. The drop-threshold percentage for threshold 3 is predefined. It is set to the queue-full state. • For *dscp1...dscp8*, enter up to eight values, and separate each value with a space. The range is 0 to 63.

	Command or Action	Purpose
		• For *cos1...cos8*, enter up to eight values, and separate each value with a space. The range is 0 to 7.
Step 3	**mls qos srr-queue input threshold** *queue-id threshold-percentage1 threshold-percentage2* **Example:** Switch(config)# **mls qos srr-queue input threshold 1 50 70**	Assigns the two WTD threshold percentages for (threshold 1 and 2) to an ingress queue. The default, both thresholds are set to 100 percent. • For *queue-id*, the range is 1 to 2. • For *threshold-percentage1 threshold-percentage2*, the range is 1 to 100. Separate each value with a space. Each threshold value is a percentage of the total number of queue descriptors allocated for the queue.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show mls qos maps** **Example:** Switch# **show mls qos maps**	Verifies your entries. The DSCP input queue threshold map appears as a matrix. The d1 column specifies the most-significant digit of the DSCP number; the d2 row specifies the least-significant digit in the DSCP number. The intersection of the d1 and the d2 values provides the queue ID and threshold ID; for example, queue 2 and threshold 1 (02-01). The CoS input queue threshold map shows the CoS value in the top row and the corresponding queue ID and threshold ID in the second row; for example, queue 2 and threshold 2 (2-2).
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file. To return to the default CoS input queue threshold map or the default DSCP input queue threshold map, use the **no mls qos srr-queue input cos-map** or the **no mls qos srr-queue input dscp-map** global configuration command. To return to the default WTD threshold percentages, use the **no mls qos srr-queue input threshold queue-id** global configuration command

Allocating Buffer Space Between the Ingress Queues

You define the ratio (allocate the amount of space) with which to divide the ingress buffers between the two queues. The buffer and the bandwidth allocation control how much data can be buffered before packets are dropped.

Beginning in privileged EXEC mode, follow these steps to allocate the buffers between the ingress queues. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos srr-queue input buffers** *percentage1 percentage2*
3. **end**
4. Use one of the following:
    - **show mls qos interface buffer**
    - **show mls qos input-queue**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal**	Enters global configuration mode.
	Example:	
	Switch# **configure terminal**	
Step 2	**mls qos srr-queue input buffers** *percentage1 percentage2*	Allocates the buffers between the ingress queues
	Example:	By default 90 percent of the buffers are allocated to queue 1, and 10 percent of the buffers are allocated to queue 2.
	Switch(config)# **mls qos srr-queue input buffers 60 40**	For *percentage1 percentage2,* the range is 0 to 100. Separate each value with a space.
		You should allocate the buffers so that the queues can handle any incoming bursty traffic.
Step 3	**end**	Returns to privileged EXEC mode.
	Example:	
	Switch(config)# **end**	
Step 4	Use one of the following:	Verifies your entries.
	• **show mls qos interface buffer**	
	• **show mls qos input-queue**	
	Example:	
	Switch# **show mls qos interface buffer**	
	or	
	Switch# **show mls qos input-queue**	

	Command or Action	**Purpose**
Step 5	**copy running-config startup-config** **Example:** Switch# **copy-running-config startup-config**	(Optional) Saves your entries in the configuration file. To return to the default setting, use the **no mls qos srr-queue input buffers** global configuration command.

Allocating Bandwidth Between the Ingress Queues

You need to specify how much of the available bandwidth is allocated between the ingress queues. The ratio of the weights is the ratio of the frequency in which the SRR scheduler sends packets from each queue. The bandwidth and the buffer allocation control how much data can be buffered before packets are dropped. On ingress queues, SRR operates only in shared mode.

> **Note** SRR bandwidth limit works in both mls qos enabled and disabled states.

Beginning in privileged EXEC mode, follow these steps to allocate bandwidth between the ingress queues. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos srr-queue input bandwidth** *weight1 weight2*
3. **end**
4. Use one of the following:
   - **show mls qos interface queueing**
   - **show mls qos input-queue**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**mls qos srr-queue input bandwidth** *weight1 weight2* **Example:** Switch(config)# **mls qos srr-queue input bandwidth 25 75**	Assigns shared round robin weights to the ingress queues. The default setting for *weight1* and *weight2* is 4 (1/2 of the bandwidth is equally shared between the two queues). For *weight1* and *weight2*, the range is 1 to 100. Separate each value with a space.

	Command or Action	**Purpose**
		SRR services the priority queue for its configured weight as specified by the **bandwidth** keyword in the **mls qos srr-queue input priority-queue** *queue-id* **bandwidth** *weight* global configuration command. Then, SRR shares the remaining bandwidth with both ingress queues and services them as specified by the weights configured with the **mls qos srr-queue input bandwidth** *weight1 weight2* global configuration command.
Step 3	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Step 4	Use one of the following:
	
	    • **show mls qos interface queueing**
	    • **show mls qos input-queue**
	
	Example:
	
	Switch# **show mls qos interface queueing**
	
	or
	
	Switch# **show mls qos input-queue**
		Verifies your entries.

Step 5	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file. To return to the default setting, use the **no mls qos srr-queue input bandwidth** global configuration command.

Configuring Egress Queue Characteristics

Depending on the complexity of your network and your QoS solution, you might need to perform all of the tasks in the following modules. You need to make decisions about these characteristics:

- Which packets are mapped by DSCP or CoS value to each queue and threshold ID?

- What drop percentage thresholds apply to the queue-set (four egress queues per port), and how much reserved and maximum memory is needed for the traffic type?

- How much of the fixed buffer space is allocated to the queue-set?

- Does the bandwidth of the port need to be rate limited?

- How often should the egress queues be serviced and which technique (shaped, shared, or both) should be used?

Configuration Guidelines

Follow these guidelines when the expedite queue is enabled or the egress queues are serviced based on their SRR weights:

- If the egress expedite queue is enabled, it overrides the SRR shaped and shared weights for queue 1.

- If the egress expedite queue is disabled and the SRR shaped and shared weights are configured, the shaped mode overrides the shared mode for queue 1, and SRR services this queue in shaped mode.

- If the egress expedite queue is disabled and the SRR shaped weights are not configured, SRR services this queue in shared mode.

Allocating Buffer Space to and Setting WTD Thresholds for an Egress Queue Set

You can guarantee the availability of buffers, set WTD thresholds, and configure the maximum allocation for a queue set by using the **mls qos queue-set output** *qset-id* **threshold** *queue-id drop-threshold1 drop-threshold2 reserved-threshold maximum-threshold* global configuration command.

Each threshold value is a percentage of the queue's allocated buffers, which you specify by using the **mls qos queue-set output** *qset-id* **buffers** *allocation1 ... allocation4* global configuration command. The queues use WTD to support distinct drop percentages for different traffic classes.

Note The egress queue default settings are suitable for most situations. You should change them only when you have a thorough understanding of the egress queues and if these settings do not meet your QoS solution.

Beginning in privileged EXEC mode, follow these steps to configure the memory allocation and to drop thresholds for a queue set. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos srr-queue output queues 8**
3. **mls qos queue-set output** *qset-id* **buffers** *allocation1 ... allocation8*
4. **mls qos queue-set output** *qset-id* **threshold** *queue-id drop-threshold1 drop-threshold2 reserved-threshold maximum-threshold*
5. **interface** *interface-id*
6. **queue-set** *qset-id*
7. **end**
8. **show mls qos interface** [*interface-id*] **buffers**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:**	Enters global configuration mode.

	Command or Action	**Purpose**
	Switch# **configure terminal**	
Step 2	**mls qos srr-queue output queues 8** **Example:** Switch(config)# **mls qos srr-queue output queues 8**	(Optional) The switch supports 4 egress queues by default, although you can enable a total of 8 egress queues. Use the optional **mls qos srr-queue output queues 8** command to enable the additional 4 egress queues. Once 8 queue support is enabled, you can then proceed to configure the additional 4 queues. Any existing egress queue configuration commands are then modified to support the additional queue parameters. **Note** The option to enable 8 queues is only available on a standalone switch. If the switch is within a stack, then only an additional 2 queues can be enabled for a total of 6 egress queues.
Step 3	**mls qos queue-set output** *qset-id* **buffers** *allocation1 ... allocation8* **Example:** Switch(config)# **mls qos queue-set output 2 buffers 40 20 20 20 10 10 10 10**	Allocates buffers to a queue set. By default, all allocation values are equally mapped among the four queues (25, 25, 25, 25). Each queue has 1/4 of the buffer space. When eight egress queues are configured, then by default 30 percent of the total buffer space is allocated to queue 2 and 10 percent (each) to queues 1,3,4,5,6,7, and 8. If you enabled 8 egress queues as described in Step 2 above, then the following applies: • For *qset-id,* enter the ID of the queue set. The range is 1 to 2. Each port belongs to a queue set, which defines all the characteristics of the four egress queues per port. • For *allocation1 ... allocation8*, specify eight percentages, one for each queue in the queue set. For *allocation1*, *allocation3*, and *allocation4* to *allocation8*, the range is 0 to 99. For *allocation2*, the range is 1 to 100 (including the CPU buffer). Allocate buffers according to the importance of the traffic; for example, give a large percentage of the buffer to the queue with the highest-priority traffic. **Note** To return to the default setting, use the **no mls qos queue-set output** *qset-id* **buffers** global configuration command.
Step 4	**mls qos queue-set output** *qset-id* **threshold** *queue-id drop-threshold1 drop-threshold2 reserved-threshold maximum-threshold*	Configures the WTD thresholds, guarantee the availability of buffers, and configures the maximum memory allocation for the queue set (four egress queues per port).

Command or Action	Purpose
Example: Switch(config)# **mls qos queue-set output 2 threshold 2 40 60 80 200**	By default, the WTD thresholds for queues 1, 3, and 4 are set to 100 percent. The thresholds for queue 2 are set to 200 percent. The reserved thresholds for queues 1, 2, 3, and 4 are set to 50 percent. The maximum thresholds for all queues are set to 400 percent by default.
	If you enabled 8 egress queues as described in Step 2 above, then the following applies:
	• For *qset-id*, enter the ID of the queue set specified in Step 3. The range is 1 to 2.
	• For *queue-id*, enter the specific queue in the queue set on which the command is performed. The queue-id range is 1-4 by default and 1-8 when 8 queues are enabled.
	• For *drop-threshold1 drop-threshold2*, specify the two WTD thresholds expressed as a percentage of the queue's allocated memory. The range is 1 to 3200 percent.
	• For *reserved-threshold*, enter the amount of memory to be guaranteed (reserved) for the queue expressed as a percentage of the allocated memory. The range is 1 to 100 percent.
	• For *maximum-threshold*, enable a queue in the full condition to obtain more buffers than are reserved for it. This is the maximum memory the queue can have before the packets are dropped if the common pool is not empty. The range is 1 to 3200 percent.
	Note To return to the default WTD threshold percentages, use the **no mls qos queue-set output** *qset-id* **threshold** [*queue-id*] global configuration command.
Step 5 **interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	Specifies the port of the outbound traffic, and enters interface configuration mode.
Step 6 **queue-set** *qset-id* **Example:** Switch(config-id)# **queue-set 2**	Maps the port to a queue set. For *qset-id*, enter the ID of the queue set specified in Step 2. The range is 1 to 2. The default is 1.

	Command or Action	**Purpose**
Step 7	**end** **Example:** Switch(config-id)# **end**	Returns to privileged EXEC mode.
Step 8	**show mls qos interface** [*interface-id*] **buffers** **Example:** Switch# **show mls qos interface buffers**	Verifies your entries.
Step 9	**copy running-config startup-config** **Example:** Switch# **copy-running-config startup-config**	(Optional) Saves your entries in the configuration file. To return to the default setting, use the **no mls qos queue-set output** *qset-id* **buffers** global configuration command. To return to the default WTD threshold percentages, use the **no mls qos queue-set output** *qset-id* **threshold** [*queue-id*] global configuration command.

Mapping DSCP or CoS Values to an Egress Queue and to a Threshold ID

You can prioritize traffic by placing packets with particular DSCPs or costs of service into certain queues and adjusting the queue thresholds so that packets with lower priorities are dropped.

Note   The egress queue default settings are suitable for most situations. You should change them only when you have a thorough understanding of egress queues and if these settings do not meet your QoS solution.

Beginning in privileged EXEC mode, follow these steps to map DSCP or CoS values to an egress queue and to a threshold ID. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. Use one of the following:

    • **mls qos srr-queue output dscp-map queue** *queue-id* **threshold** *threshold-id dscp1...dscp8*
    • **mls qos srr-queue output cos-map queue** *queue-id* **threshold** *threshold-id cos1...cos8*

3. **mls qos srr-queue output cos-map queue** *queue-id* **threshold** *threshold-id cos1...cos8*
4. **end**
5. **show mls qos maps**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	Use one of the following: • **mls qos srr-queue output dscp-map queue** *queue-id* **threshold** *threshold-id dscp1...dscp8* • **mls qos srr-queue output cos-map queue** *queue-id* **threshold** *threshold-id cos1...cos8* **Example:** Switch(config)# **mls qos srr-queue output dscp-map queue 1 threshold 2 10 11**	Maps DSCP or CoS values to an egress queue and to a threshold ID. By default, DSCP values 0–15 are mapped to queue 2 and threshold 1. DSCP values 16–31 are mapped to queue 3 and threshold 1. DSCP values 32–39 and 48–63 are mapped to queue 4 and threshold 1. DSCP values 40–47 are mapped to queue 1 and threshold 1. By default, CoS values 0 and 1 are mapped to queue 2 and threshold 1. CoS values 2 and 3 are mapped to queue 3 and threshold 1. CoS values 4, 6, and 7 are mapped to queue 4 and threshold 1. CoS value 5 is mapped to queue 1 and threshold 1. • For *queue-id*, the range is 1 to 4. • For *threshold-id*, the range is 1 to 3. The drop-threshold percentage for threshold 3 is predefined. It is set to the queue-full state. • For *dscp1...dscp8*, enter up to eight values, and separate each value with a space. The range is 0 to 63. • For *cos1...cos8*, enter up to eight values, and separate each value with a space. The range is 0 to 7. **Note**　To return to the default DSCP output queue threshold map or the default CoS output queue threshold map, use the **no mls qos srr-queue output dscp-map** or the **no mls qos srr-queue output cos-map** global configuration command.
Step 3	**mls qos srr-queue output cos-map queue** *queue-id* **threshold** *threshold-id cos1...cos8* **Example:** Switch(config)# **mls qos srr-queue output cos-map queue 3 threshold 1 2 3**	Maps CoS values to an egress queue and to a threshold ID. By default, CoS values 0 and 1 are mapped to queue 2 and threshold 1. CoS values 2 and 3 are mapped to queue 3 and threshold 1. CoS values 4, 6, and 7 are mapped to queue 4 and threshold 1. CoS value 5 is mapped to queue 1 and threshold 1. • For *queue-id*, the range is 1 to 4. • For *threshold-id*, the range is 1 to 3. The drop-threshold percentage for threshold 3 is predefined. It is set to the queue-full state.

	Command or Action	**Purpose**
		• For *cos1...cos8*, enter up to eight values, and separate each value with a space. The range is 0 to 7.
		Note To return to the default CoS output queue threshold map, use the **no mls qos srr-queue output cos-map** global configuration command.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show mls qos maps** **Example:** Switch# **show mls qos maps**	Verifies your entries. The DSCP output queue threshold map appears as a matrix. The d1 column specifies the most-significant digit of the DSCP number; the d2 row specifies the least-significant digit in the DSCP number. The intersection of the d1 and the d2 values provides the queue ID and threshold ID; for example, queue 2 and threshold 1 (02-01). The CoS output queue threshold map shows the CoS value in the top row and the corresponding queue ID and threshold ID in the second row; for example, queue 2 and threshold 2 (2-2).
Step 6	**copy running-config startup-config** **Example:** Switch# **copy-running-config startup-config**	(Optional) Saves your entries in the configuration file. To return to the default DSCP output queue threshold map or the default CoS output queue threshold map, use the **no mls qos srr-queue output dscp-map** or the **no mls qos srr-queue output cos-map** global configuration command.

Configuring SRR Shaped Weights on Egress Queues

You can specify how much of the available bandwidth is allocated to each queue. The ratio of the weights is the ratio of frequency in which the SRR scheduler sends packets from each queue.

You can configure the egress queues for shaped or shared weights, or both. Use shaping to smooth bursty traffic or to provide a smoother output over time.

Beginning in privileged EXEC mode, follow these steps to assign the shaped weights and to enable bandwidth shaping on the four egress queues mapped to a port. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **srr-queue bandwidth shape** *weight1 weight2 weight3 weight4*
4. **end**

5. **show mls qos interface** *interface-id* **queueing**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* Example: Switch(config)# **interface gigabitethernet2/0/1**	Specifies the port of the outbound traffic, and enters interface configuration mode.
Step 3	**srr-queue bandwidth shape** *weight1 weight2 weight3 weight4* Example: Switch(config-if)# **srr-queue bandwidth shape 8 0 0 0**	Assigns SRR weights to the egress queues. By default, weight1 is set to 25; weight2, weight3, and weight4 are set to 0, and these queues are in shared mode. For *weight1 weight2 weight3 weight4*, enter the weights to control the percentage of the port that is shaped. The inverse ratio (1/weight) controls the shaping bandwidth for this queue. Separate each value with a space. The range is 0 to 65535. If you configure a weight of 0, the corresponding queue operates in shared mode. The weight specified with the **srr-queue bandwidth shape** command is ignored, and the weights specified with the **srr-queue bandwidth share** interface configuration command for a queue come into effect. When configuring queues in the same queue-set for both shaping and sharing, make sure that you configure the lowest number queue for shaping. The shaped mode overrides the shared mode. To return to the default setting, use the **no srr-queue bandwidth shape** interface configuration command.
Step 4	**end** Example: Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 5	**show mls qos interface** *interface-id* **queueing** Example:	Verifies your entries.

	Command or Action	Purpose
	Switch# **show mls qos interface** **interface-id queuing**	
Step 6	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.
	Example:	To return to the default setting, use the **no srr-queue bandwidth shape** interface configuration command.
	Switch# **copy running-config** **startup-config**	

Configuring SRR Shared Weights on Egress Queues

In shared mode, the queues share the bandwidth among them according to the configured weights. The bandwidth is guaranteed at this level but not limited to it. For example, if a queue empties and does not require a share of the link, the remaining queues can expand into the unused bandwidth and share it among them. With sharing, the ratio of the weights controls the frequency of dequeuing; the absolute values are meaningless.

> **Note** The egress queue default settings are suitable for most situations. You should change them only when you have a thorough understanding of the egress queues and if these settings do not meet your QoS solution.

Beginning in privileged EXEC mode, follow these steps to assign the shared weights and to enable bandwidth sharing on the four egress queues mapped to a port. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **srr-queue bandwidth share** *weight1 weight2 weight3 weight4*
4. **end**
5. **show mls qos interface** *interface-id* **queueing**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
	Example:	
	Switch# **configure terminal**	
Step 2	**interface** *interface-id*	Specifies the port of the outbound traffic, and enters interface configuration mode.
	Example:	
	Switch(config)# **interface**	

	Command or Action	Purpose
	`gigabitethernet2/0/1`	
Step 3	**srr-queue bandwidth share** *weight1 weight2 weight3 weight4* **Example:** `Switch(config-id)# srr-queue bandwidth share 1 2 3 4`	Assigns SRR weights to the egress queues. By default, all four weights are 25 (1/4 of the bandwidth is allocated to each queue). For *weight1 weight2 weight3 weight4*, enter the weights to control the ratio of the frequency in which the SRR scheduler sends packets. Separate each value with a space. The range is 1 to 255. To return to the default setting, use the **no srr-queue bandwidth share** interface configuration command.
Step 4	**end** **Example:** `Switch(config-id)# end`	Returns to privileged EXEC mode.
Step 5	**show mls qos interface** *interface-id* **queueing** **Example:** `Switch# show mls qos interface interface_id queuing`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy-running-config startup-config`	(Optional) Saves your entries in the configuration file. To return to the default setting, use the **no srr-queue bandwidth share** interface configuration command.

Configuring the Egress Expedite Queue

You can ensure that certain packets have priority over all others by queuing them in the egress expedite queue. SRR services this queue until it is empty before servicing the other queues.

Beginning in privileged EXEC mode, follow these steps to enable the egress expedite queue. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos**
3. **interface** *interface-id*
4. **priority-queue out**
5. **end**
6. **show running-config**

7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**mls qos** **Example:** Switch(config)# **mls qos**	Enables QoS on a switch.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	Specifies the egress port, and enters interface configuration mode.
Step 4	**priority-queue out** **Example:** Switch(config-if)# **priority-queue out**	Enables the egress expedite queue, which is disabled by default. When you configure this command, the SRR weight and queue size ratios are affected because there is one fewer queue participating in SRR. This means that *weight1* in the **srr-queue bandwidth shape** or the **srr-queue bandwidth share** command is ignored (not used in the ratio calculation). **Note** To disable the egress expedite queue, use the **no priority-queue out** interface configuration command.
Step 5	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 7	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Command or Action	Purpose
Example: Switch# **copy running-config startup-config**	To disable the egress expedite queue, use the **no priority-queue out** interface configuration command.

Limiting the Bandwidth on an Egress Interface

You can limit the bandwidth on an egress port. For example, if a customer pays only for a small percentage of a high-speed link, you can limit the bandwidth to that amount.

> **Note** The egress queue default settings are suitable for most situations. You should change them only when you have a thorough understanding of the egress queues and if these settings do not meet your QoS solution.

Beginning in privileged EXEC mode, follow these steps to limit the bandwidth on an egress port. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **srr-queue bandwidth limit** *weight1*
4. **end**
5. **show mls qos interface** [*interface-id*] **queueing**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet2/0/1**	Specifies the port to be rate-limited, and enters interface configuration mode.
Step 3	**srr-queue bandwidth limit** *weight1* **Example:** Switch(config-if)# **srr-queue**	Specifies the percentage of the port speed to which the port should be limited. The range is 10 to 90. By default, the port is not rate-limited and is set to 100 percent.

	Command or Action	Purpose
	`bandwidth limit 80`	**Note** To return to the default setting, use the **no srr-queue bandwidth limit** interface configuration command.
Step 4	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show mls qos interface** [*interface-id*] **queueing** **Example:** `Switch# show mls qos interface interface_id queueing`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Switch# copy-running-config startup-config`	(Optional) Saves your entries in the configuration file. To return to the default setting, use the **no srr-queue bandwidth limit** interface configuration command.

Monitoring Standard QoS

Table 59: Commands for Monitoring Standard QoS on the Switch

Command	Description
show class-map [*class-map-name*]	Displays QoS class maps, which define the match criteria to classify traffic.
show mls qos	Displays global QoS configuration information.
show mls qos aggregate-policer [*aggregate-policer-name*]	Displays the aggregate policer configuration.
show mls qos interface [*interface-id*] [**buffers** | **policers** | **queueing** | **statistics**]	Displays QoS information at the port level, including the buffer allocation, which ports have configured policers, the queueing strategy, and the ingress and egress statistics.
show mls qos maps [**cos-dscp** | |**cos-output-q** | **dscp-cos** | |**dscp-mutation** *dscp-mutation-name* | **dscp-output-q** | **ip-prec-dscp** | **policed-dscp**]	Displays QoS mapping information.
show mls qos queue-set [*qset-id*]	Displays QoS settings for the egress queues.

Command	Description
show mls qos vlan *vlan-id*	Displays the policy maps attached to the specified SVI.
show policy-map [*policy-map-name* [**class** *class-map-name*]]	Displays QoS policy maps, which define classification criteria for incoming traffic. Do not use the **show policy-map interface** privileged EXEC command to display classification information for incoming traffic. The **control-plane** and **interface** keywords are not supported, and the statistics shown in the display should be ignored.
**show running-config	include rewrite**	Displays the DSCP transparency setting.

Configuration Examples for QoS

Example: Configuring Port to the DSCP-Trusted State and Modifying the DSCP-to-DSCP-Mutation Map

This example shows how to configure a port to the DSCP-trusted state and to modify the DSCP-to-DSCP-mutation map (named *gi1/0/2-mutation*) so that incoming DSCP values 10 to 13 are mapped to DSCP 30:

```
Switch(config)# mls qos map dscp-mutation gigabitethernet1/0/2-mutation
10 11 12 13 to 30
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# mls qos trust dscp
Switch(config-if)# mls qos dscp-mutation gigabitethernet1/0/2-mutation
Switch(config-if)# end
```

Examples: Classifying Traffic by Using ACLs

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. Any host with a source address that does not match the access list statements is rejected.

```
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
! (Note: all other access implicitly denied)
```

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
```

This example shows how to create an ACL that permits PIM traffic from any source to a destination group address of 224.0.0.2 with a DSCP set to 32:

```
Switch(config)# access-list 102 permit pim any 224.0.0.2 dscp 32
```

This example shows how to create an ACL that permits IPv6 traffic from any source to any destination that has the DSCP value set to 32:

```
Switch(config)# ipv6 access-list 100 permit ip any any dscp 32
```

This example shows how to create an ACL that permits IPv6 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)# ipv6 access-list ipv6_Name_ACL permit ip host 10::1 host 10.1.1.2
precedence 5
```

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
! (Note: all other access implicitly denied)
```

Examples: Classifying Traffic by Using Class Maps

This example shows how to 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 ip any any dscp 10
Switch(config)# class-map class1
Switch(config-cmap)# match access-group 103
Switch(config-cmap)# end
Switch#
```

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 class2
Switch(config-cmap)# match ip dscp 10 11 12
Switch(config-cmap)# end
Switch#
```

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 class3
Switch(config-cmap)# match ip precedence 5 6 7
Switch(config-cmap)# end
Switch#
```

This example shows how to configure a class map to match IP DSCP and IPv6:

```
Switch(config)# Class-map cm-1
Switch(config-cmap)# match ip dscp 10
Switch(config-cmap)# match protocol ipv6
Switch(config-cmap)# exit
Switch(config)# Class-map cm-2
Switch(config-cmap)# match ip dscp 20
Switch(config-cmap)# match protocol ip
Switch(config-cmap)# exit
Switch(config)# Policy-map pm1
Switch(config-pmap)# class cm-1
Switch(config-pmap-c)# set dscp 4
Switch(config-pmap-c)# exit
Switch(config-pmap)# class cm-2
Switch(config-pmap-c)# set dscp 6
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface G1/0/1
Switch(config-if)# service-policy input pm1
```

This example shows how to configure a class map that applies to both IPv4 and IPv6 traffic:

```
Switch(config)# ip access-list 101 permit ip any any
Switch(config)# ipv6 access-list ipv6-any permit ip any any
Switch(config)# Class-map cm-1
Switch(config-cmap)# match access-group 101
Switch(config-cmap)# exit
Switch(config)# class-map cm-2
Switch(config-cmap)# match access-group name ipv6-any
Switch(config-cmap)# exit
Switch(config)# Policy-map pm1
Switch(config-pmap)# class cm-1
Switch(config-pmap-c)# set dscp 4
Switch(config-pmap-c)# exit
Switch(config-pmap)# class cm-2
Switch(config-pmap-c)# set dscp 6
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface G0/1
Switch(config-if)# switch mode access
Switch(config-if)# service-policy input pm1
```

Examples: Classifying, Policing, and Marking Traffic on Physical Ports Using Policy Maps

This example shows how to create a policy map and attach it to an ingress port. In the configuration, the IP standard ACL permits traffic from network 10.1.0.0. For traffic matching this classification, the DSCP value in the incoming packet is trusted. If the matched traffic exceeds an average traffic rate of 48000 b/s and a normal burst size of 8000 bytes, its DSCP is marked down (based on the policed-DSCP map) and sent:

```
Switch(config)# access-list 1 permit 10.1.0.0 0.0.255.255
Switch(config)# class-map ipclass1
Switch(config-cmap)# match access-group 1
Switch(config-cmap)# exit
Switch(config)# policy-map flow1t
Switch(config-pmap)# class ipclass1
Switch(config-pmap-c)# trust dscp
Switch(config-pmap-c)# police 1000000 8000 exceed-action policed-dscp-transmit
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# service-policy input flow1t
```

This example shows how to create a Layer 2 MAC ACL with two permit statements and attach it to an ingress port. The first permit statement allows traffic from the host with MAC address 0001.0000.0001 destined for the host with MAC address 0002.0000.0001. The second permit statement allows only Ethertype XNS-IDP traffic from the host with MAC address 0001.0000.0002 destined for the host with MAC address 0002.0000.0002.

```
Switch(config)# mac access-list extended maclist1
Switch(config-ext-mac)# permit 0001.0000.0001 0.0.0 0002.0000.0001 0.0.0
Switch(config-ext-mac)# permit 0001.0000.0002 0.0.0 0002.0000.0002 0.0.0 xns-idp
Switch(config-ext-mac)# exit
Switch(config)# mac access-list extended maclist2
Switch(config-ext-mac)# permit 0001.0000.0003 0.0.0 0002.0000.0003 0.0.0
Switch(config-ext-mac)# permit 0001.0000.0004 0.0.0 0002.0000.0004 0.0.0 aarp
Switch(config-ext-mac)# exit
Switch(config)# class-map macclass1
Switch(config-cmap)# match access-group maclist1
Switch(config-cmap)# exit
Switch(config)# policy-map macpolicy1
Switch(config-pmap)# class macclass1
Switch(config-pmap-c)# set dscp 63
Switch(config-pmap-c)# exit
Switch(config-pmap)# class macclass2 maclist2
Switch(config-pmap-c)# set dscp 45
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# mls qos trust cos
Switch(config-if)# service-policy input macpolicy1
```

This example shows how to create a class map that applies to both IPv4 and IPv6 traffic with the default class applied to unclassified traffic:

```
Switch(config)# ip access-list 101 permit ip any any
Switch(config)# ipv6 access-list ipv6-any permit ip any any
```

```
Switch(config)# class-map cm-1
Switch(config-cmap)# match access-group 101
Switch(config-cmap)# exit
Switch(config)# class-map cm-2
Switch(config-cmap)# match access-group name ipv6-any
Switch(config-cmap)# exit
Switch(config)# policy-map pm1
Switch(config-pmap)# class cm-1
Switch(config-pmap-c)# set dscp 4
Switch(config-pmap-c)# exit
Switch(config-pmap)# class cm-2
Switch(config-pmap-c)# set dscp 6
Switch(config-pmap-c)# exit
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# set dscp 10
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface G0/1
Switch(config-if)# switch mode access
Switch(config-if)# service-policy input pm1
```

Examples: Classifying, Policing, and Marking Traffic on SVIs by Using Hierarchical Policy Maps

This example shows how to create a hierarchical policy map:

```
Switch> enable
Switch#  configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# access-list 101 permit ip any any
Switch(config)# class-map cm-1
Switch(config-cmap)# match access 101
Switch(config-cmap)# exit
Switch(config)# exit
Switch#
Switch#
```

This example shows how to attach the new map to an SVI:

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# class-map cm-interface-1
Switch(config-cmap)# match input gigabitethernet3/0/1 - gigabitethernet3/0/2
Switch(config-cmap)# exit
Switch(config)# policy-map port-plcmap
Switch(config-pmap)# class cm-interface-1
Switch(config-pmap-c)# police 900000 9000 exc policed-dscp-transmit
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# policy-map vlan-plcmap
Switch(config-pmap)# class cm-1
Switch(config-pmap-c)# set dscp 7
Switch(config-pmap-c)# service-policy port-plcmap-1
Switch(config-pmap-c)# exit
Switch(config-pmap)# class cm-2
Switch(config-pmap-c)# service-policy port-plcmap-1
Switch(config-pmap-c)# set dscp 10
```

```
Switch(config-pmap)# exit
Switch(config-pmap)# class cm-3
Switch(config-pmap-c)# service-policy port-plcmap-2
Switch(config-pmap-c)# set dscp 20
Switch(config-pmap)# exit
Switch(config-pmap)# class cm-4
Switch(config-pmap-c)# trust dscp
Switch(config-pmap)# exit
Switch(config)# interface vlan 10
Switch(config-if)# service-policy input vlan-plcmap
Switch(config-if)# exit
Switch(config)# exit
Switch#
```

This example shows that when a child-level policy map is attached below a class, an action must be specified for the class:

```
Switch(config)# policy-map vlan-plcmap
Switch(config-pmap)# class cm-5
Switch(config-pmap-c)# set dscp 7
Switch(config-pmap-c)# service-policy port-plcmap-1
```

This example shows how to configure a class map to match IP DSCP and IPv6:

```
Switch(config)# class-map cm-1
Switch(config-cmap)# match ip dscp 10
Switch(config-cmap)# match protocol ipv6
Switch(config-cmap)# exit
Switch(config)# class-map cm-2
Switch(config-cmap)# match ip dscp 20
Switch(config-cmap)# match protocol ip
Switch(config-cmap)# exit
Switch(config)# policy-map pm1
Switch(config-pmap)# class cm-1
Switch(config-pmap-c)# set dscp 4
Switch(config-pmap-c)# exit
Switch(config-pmap)# class cm-2
Switch(config-pmap-c)# set dscp 6
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface G1/0/1
Switch(config-if)# service-policy input pm1
```

This example shows how to configure default traffic class to a policy map:

```
Switch# configure terminal
Switch(config)# class-map cm-3
Switch(config-cmap)# match ip dscp 30
Switch(config-cmap)# match protocol ipv6
Switch(config-cmap)# exit
Switch(config)# class-map cm-4
Switch(config-cmap)# match ip dscp 40
Switch(config-cmap)# match protocol ip
Switch(config-cmap)# exit
Switch(config)# policy-map pm3
Switch(config-pmap)# class class-default
Switch(config-pmap)# set dscp 10
Switch(config-pmap-c)# exit
```

```
Switch(config-pmap)# class cm-3
Switch(config-pmap-c) set dscp 4
Switch(config-pmap-c)# exit
Switch(config-pmap)# class cm-4
Switch(config-pmap-c)# trust cos
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
```

This example shows how the default traffic class is automatically placed at the end of policy-map pm3 even though class-default was configured first:

```
Switch# show policy-map pm3
  Policy Map pm3
    Class cm-3
      set dscp 4
    Class cm-4
      trust cos
    Class class-default
      police 8000 80000 exceed-action drop
Switch#
```

Examples: Classifying, Policing, and Marking Traffic by Using Aggregate Policers

This example shows how to create an aggregate policer and attach it to multiple classes within a policy map. In the configuration, the IP ACLs permit traffic from network 10.1.0.0 and from host 11.3.1.1. For traffic coming from network 10.1.0.0, the DSCP in the incoming packets is trusted. For traffic coming from host 11.3.1.1, the DSCP in the packet is changed to 56. The traffic rate from the 10.1.0.0 network and from host 11.3.1.1 is policed. If the traffic exceeds an average rate of 48000 b/s and a normal burst size of 8000 bytes, its DSCP is marked down (based on the policed-DSCP map) and sent. The policy map is attached to an ingress port.

```
Switch(config)# access-list 1 permit 10.1.0.0 0.0.255.255
Switch(config)# access-list 2 permit 11.3.1.1
Switch(config)# mls qos aggregate-police transmit1 48000 8000 exceed-action
policed-dscp-transmit
Switch(config)# class-map ipclass1
Switch(config-cmap)# match access-group 1
Switch(config-cmap)# exit
Switch(config)# class-map ipclass2
Switch(config-cmap)# match access-group 2
Switch(config-cmap)# exit
Switch(config)# policy-map aggflow1
Switch(config-pmap)# class ipclass1
Switch(config-pmap-c)# trust dscp
Switch(config-pmap-c)# police aggregate transmit1
Switch(config-pmap-c)# exit
Switch(config-pmap)# class ipclass2
Switch(config-pmap-c)# set dscp 56
Switch(config-pmap-c)# police aggregate transmit1
Switch(config-pmap-c)# exit
Switch(config-pmap)# class class-default
Switch(config-pmap-c)# set dscp 10
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
```

```
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# service-policy input aggflow1
Switch(config-if)# exit
```

Examples: Configuring DSCP Maps

This example shows how to modify and display the CoS-to-DSCP map:

```
Switch(config)# mls qos map cos-dscp 10 15 20 25 30 35 40 45
Switch(config)# end
Switch# show mls qos maps cos-dscp

Cos-dscp map:
        cos:   0  1  2  3  4  5  6  7
      --------------------------------
       dscp:   10 15 20 25 30 35 40 45
```

This example shows how to modify and display the IP-precedence-to-DSCP map:

```
Switch(config)# mls qos map ip-prec-dscp 10 15 20 25 30 35 40 45
Switch(config)# end
Switch# show mls qos maps ip-prec-dscp

IpPrecedence-dscp map:
     ipprec:   0  1  2  3  4  5  6  7
      -------------------------------
       dscp:   10 15 20 25 30 35 40 45
```

This example shows how to map DSCP 50 to 57 to a marked-down DSCP value of 0:

```
Switch(config)# mls qos map policed-dscp 50 51 52 53 54 55 56 57 to 0
Switch(config)# end
Switch# show mls qos maps policed-dscp
Policed-dscp map:
     d1 :  d2 0  1  2  3  4  5  6  7  8  9
     --------------------------------------
      0 :    00 01 02 03 04 05 06 07 08 09
      1 :    10 11 12 13 14 15 16 17 18 19
      2 :    20 21 22 23 24 25 26 27 28 29
      3 :    30 31 32 33 34 35 36 37 38 39
      4 :    40 41 42 43 44 45 46 47 48 49
      5 :    00 00 00 00 00 00 00 00 58 59
      6 :    60 61 62 63
```

Note In this policed-DSCP map, the marked-down DSCP values are shown in the body of the matrix. The d1 column specifies the most-significant digit of the original DSCP; the d2 row specifies the least-significant digit of the original DSCP. The intersection of the d1 and d2 values provides the marked-down value. For example, an original DSCP value of 53 corresponds to a marked-down DSCP value of 0.

This example shows how to map DSCP values 0, 8, 16, 24, 32, 40, 48, and 50 to CoS value 0 and to display the map:

```
Switch(config)# mls qos map dscp-cos 0 8 16 24 32 40 48 50 to 0
Switch(config)# end
Switch# show mls qos maps dscp-cos
Dscp-cos map:
     d1 :  d2 0   1   2   3   4   5   6   7   8   9
     ---------------------------------------
      0 :     00  00  00  00  00  00  00  00  00  01
      1 :     01  01  01  01  01  01  00  02  02  02
      2 :     02  02  02  02  00  03  03  03  03  03
      3 :     03  03  00  04  04  04  04  04  04  04
      4 :     00  05  05  05  05  05  05  05  00  06
      5 :     00  06  06  06  06  06  07  07  07  07
      6 :     07  07  07  07
```

Note In the above DSCP-to-CoS map, the CoS values are shown in the body of the matrix. The d1 column specifies the most-significant digit of the DSCP; the d2 row specifies the least-significant digit of the DSCP. The intersection of the d1 and d2 values provides the CoS value. For example, in the DSCP-to-CoS map, a DSCP value of 08 corresponds to a CoS value of 0.

This example shows how to define the DSCP-to-DSCP-mutation map. All the entries that are not explicitly configured are not modified (remains as specified in the null map):

```
Switch(config)# mls qos map dscp-mutation mutation1 1 2 3 4 5 6 7 to 0
Switch(config)# mls qos map dscp-mutation mutation1 8 9 10 11 12 13 to 10
Switch(config)# mls qos map dscp-mutation mutation1 20 21 22 to 20
Switch(config)# mls qos map dscp-mutation mutation1 30 31 32 33 34 to 30
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# mls qos trust dscp
Switch(config-if)# mls qos dscp-mutation mutation1
Switch(config-if)# end
Switch# show mls qos maps dscp-mutation mutation1
Dscp-dscp mutation map:
   mutation1:
     d1 :  d2 0   1   2   3   4   5   6   7   8   9
     ---------------------------------------
      0 :     00  00  00  00  00  00  00  00  10  10
      1 :     10  10  10  10  14  15  16  17  18  19
      2 :     20  20  20  23  24  25  26  27  28  29
      3 :     30  30  30  30  30  35  36  37  38  39
      4 :     40  41  42  43  44  45  46  47  48  49
      5 :     50  51  52  53  54  55  56  57  58  59
      6 :     60  61  62  63
```

Note In the above DSCP-to-DSCP-mutation map, the mutated values are shown in the body of the matrix. The d1 column specifies the most-significant digit of the original DSCP; the d2 row specifies the least-significant digit of the original DSCP. The intersection of the d1 and d2 values provides the mutated value. For example, a DSCP value of 12 corresponds to a mutated value of 10.

Examples: Configuring Egress Queue Characteristics

This example shows how to map DSCP values 10 and 11 to egress queue 1 and to threshold 2:

```
Switch(config)# mls qos srr-queue output dscp-map queue 1 threshold 2 10 11
```

This example shows how to configure bandwidth shaping on queue 1. Because the weight ratios for queues 2, 3, and 4 are set to 0, these queues operate in shared mode. The bandwidth weight for queue 1 is 1/8, which is 12.5 percent:

```
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# srr-queue bandwidth shape 8 0 0 0
```

This example shows how to configure the weight ratio of the SRR scheduler running on an egress port. Four queues are used, and the bandwidth ratio allocated for each queue in shared mode is 1/(1+2+3+4), 2/(1+2+3+4), 3/(1+2+3+4), and 4/(1+2+3+4), which is 10 percent, 20 percent, 30 percent, and 40 percent for queues 1, 2, 3, and 4. This means that queue 4 has four times the bandwidth of queue 1, twice the bandwidth of queue 2, and one-and-a-third times the bandwidth of queue 3.

```
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# srr-queue bandwidth share 1 2 3 4
```

This example shows how to enable the egress expedite queue when the SRR weights are configured. The egress expedite queue overrides the configured SRR weights.

```
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# srr-queue bandwidth shape 25 0 0 0
Switch(config-if)# srr-queue bandwidth share 30 20 25 25
Switch(config-if)# priority-queue out
Switch(config-if)# end
```

This example shows how to limit the bandwidth on a port to 80 percent:

```
Switch(config)# interface gigabitethernet2/0/1
Switch(config-if)# srr-queue bandwidth limit 80
```

When you configure this command to 80 percent, the port is idle 20 percent of the time. The line rate drops to 80 percent of the connected speed, which is 800 Mb/s. These values are not exact because the hardware adjusts the line rate in increments of six.

Where to Go Next

Review the auto-QoS documentation to see if you can use these automated capabilities for your QoS configuration.

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this book.	*Catalyst 2960-XR Switch Quality of Service Command Reference*

Standards and RFCs

Standard/RFC	Title
—	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for QoS

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Configuring Auto-QoS

Prerequisites for Auto-QoS

Before configuring standard QoS or auto-QoS, you must have a thorough understanding of these items:

- 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.

Auto-QoS VoIP Considerations

Before configuring auto-QoS for VoIP, you should be aware of this information:

- Auto-QoS configures the switch for VoIP with Cisco IP Phones on nonrouted and routed ports. Auto-QoS also configures the switch for VoIP with devices running the Cisco SoftPhone application.

> **Note** When a device running Cisco SoftPhone is connected to a nonrouted or routed port, the switch supports only one Cisco SoftPhone application per port.

- When enabling auto-QoS with a Cisco IP Phone on a routed port, you must assign a static IP address to the IP phone.

- This release supports only Cisco IP SoftPhone Version 1.3(3) or later.

- Connected devices must use Cisco Call Manager Version 4 or later.

- Auto-Qos VoIP uses the **priority-queue** interface configuration command for an egress interface. You can also configure a policy-map and trust device on the same interface for Cisco IP phones.

Auto-QoS Enhanced Considerations

Auto-QoS is enhanced to support video. Automatic configurations are generated that classify and trust traffic from Cisco TelePresence systems and Cisco IP cameras.

Before configuring auto-QoS enhanced, you should be aware of this information:

- The **auto qos srnd4** global configuration command is generated as a result of enhanced auto-QoS configuration.

Restrictions for Auto-QoS

The following are restrictions for automatic QoS (auto-QoS):

- After auto-QoS is enabled, do not modify a policy map or aggregate policer that includes *AutoQoS* in its name. If you need to modify the policy map or aggregate policer, make a copy of it, and change the copied policy map or policer. To use this new policy map instead of the generated one, remove the generated policy map from the interface, and apply the new policy map to the interface.

- To take advantage of the auto-QoS defaults, you should enable auto-QoS before you configure other QoS commands. If necessary, you can fine-tune the QoS configuration, but we recommend that you do so only after the auto-QoS configuration is completed.

- By default, the CDP is enabled on all ports. For auto-QoS to function properly, do not disable CDP.

Note You can enable auto-QoS on static, dynamic-access, voice VLAN access, and trunk ports.

Information About Configuring Auto-QoS

Auto-QoS Overview

You can use the auto-QoS feature to simplify the deployment of QoS features. Auto-QoS determines the network design and enables QoS configurations so that the switch can prioritize different traffic flows. It uses the egress queues instead of using the default (disabled) QoS behavior. The switch offers best-effort service to each packet, regardless of the packet contents or size, and sends it from a single queue.

When you enable auto-QoS, it automatically classifies traffic based on the traffic type and ingress packet label. The switch uses the classification results to choose the appropriate egress queue.

You can use auto-QoS commands to identify ports connected to the following Cisco devices:

- Cisco IP Phones

- Devices running the Cisco SoftPhone application

- Cisco TelePresence

- Cisco IP Camera

- Cisco digital media player

You also use the auto-QoS commands to identify ports that receive trusted traffic through an uplink. Auto-QoS then performs these functions:

- Detects the presence or absence of auto-QoS devices through conditional trusted interfaces.

- Configures QoS classification

- Configures egress queues

Auto-QoS Compact Overview

When you enter an auto-QoS command, the switch displays all the generated commands as if the commands were entered from the CLI. You can use the auto-QoS compact feature to hide the auto-QoS generated commands from the running configuration. This would make it easier to comprehend the running-configuration and also help to increase efficient usage of memory.

Generated Auto-QoS Configuration

By default, auto-QoS is disabled on all ports. Packets are not modified--the CoS, DSCP and IP precedence values in the packet are not changed.

When you enable the auto-QoS feature on the first port of the interface:

- Ingress packet label is used to categorize traffic, to assign packet labels, and to configure the ingress and egress queues.

- QoS is globally enabled (**mls qos** global configuration command), and other global configuration commands are automatically generated. (See Examples: Global Auto-QoS Configuration, on page 589).

- Switch enables the trusted boundary feature and uses the Cisco Discovery Protocol (CDP) to detect the presence of a supported device.

- Policing is used to determine whether a packet is in or out of profile and specifies the action on the packet.

VoIP Device Specifics

The following actions occur when you issue these auto-QoS commands on a port:

- **auto qos voip cisco-phone**—When you enter this command on a port at the network edge connected to a Cisco IP Phone, the switch enables the trusted boundary feature. If the packet does not have a DSCP value of 24, 26, or 46 or is out of profile, the switch changes the DSCP value to 0. When there is no

Cisco IP Phone, the ingress classification is set to not trust the QoS label in the packet. The policing is applied to the traffic matching the policy-map classification before the switch enables the trust boundary feature.

- **auto qos voip cisco-softphone** —When you enter this interface configuration command on a port at the network edge that is connected to a device running the Cisco SoftPhone, the switch uses policing to determine whether a packet is in or out of profile and to specify the action on the packet. If the packet does not have a DSCP value of 24, 26, or 46 or is out of profile, the switch changes the DSCP value to 0.

- **auto qos voip trust**—When you enter this interface configuration command on a port connected to the network interior, the switch trusts the CoS value for nonrouted ports or the DSCP value for routed ports in ingress packets (the assumption is that traffic has already been classified by other edge devices).

The switch configures egress queues on the port according to the settings in the following tables.

Table 60: Traffic Types, Packet Labels, and Queues

	VoIP Data Traffic	VoIP Control Traffic	Routing Protocol Traffic	STP BPDU Traffic	Real-Time Video Traffic	All Other Traffic
DSCP value	46	24, 26	48	56	34	–
CoS value	5	3	6	7	3	–
CoS-to-Egress queue map	4, 5 (queue 1)	2, 3, 6, 7 (queue 2)			0 (queue 3)	2 (queue 3)	0, 1 (queue 4)

The following table shows the generated auto-QoS configuration for the egress queues.

Table 61: Auto-QoS Configuration for the Egress Queues

Egress Queue	Queue Number	CoS-to-Queue Map	Queue Weight (Bandwidth)	Queue (Buffer) Size for Gigabit-Capable Ports	Queue (Buffer) Size for 10/100 Ethernet Ports
Priority	1	4, 5	Up to 100 percent	25 percent	15 percent
SRR shared	2	2, 3, 6, 7	10 percent	25 percent	25 percent
SRR shared	3	0	60 percent	25 percent	40 percent
SRR shared	4	1	20 percent	25 percent	20 percent

- When you enable auto-QoS by using the **auto qos voip cisco-phone**, the **auto qos voip cisco-softphone**, or the **auto qos voip trust** interface configuration command, the switch automatically generates a QoS configuration based on the traffic type and ingress packet label and applies the commands listed in Examples: Global Auto-QoS Configuration, on page 589 to the port.

Effects of Auto-QoS on Running Configuration

When auto-QoS is enabled, the **auto qos** interface configuration commands and the generated global configuration are added to the running configuration.

The switch applies the auto-QoS-generated commands as if the commands were entered from the CLI. An existing user configuration can cause the application of the generated commands to fail or to be overridden by the generated commands. These actions may occur without warning. If all the generated commands are successfully applied, any user-entered configuration that was not overridden remains in the running configuration. Any user-entered configuration that was overridden can be retrieved by reloading the switch without saving the current configuration to memory. If the generated commands are not applied, the previous running configuration is restored.

How to Configure Auto-QoS

Configuring Auto-QoS

Enabling Auto-QoS

For optimum QoS performance, enable auto-QoS on all the devices in your network.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. Use one of the following:

   - **auto qos voip** {**cisco-phone** | **cisco-softphone** | **trust**}
   - **auto qos video** {**cts** | **ip-camera** | **media-player**}
   - **auto qos classify** [**police**]
   - **auto qos trust** {**cos** | **dscp**}

4. **exit**
5. **interface** *interface-id*
6. **auto qos trust**
7. **end**
8. **show auto qos interface** *interface-id*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.

	Command or Action	**Purpose**
Step 2	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet 3/0/1**	Specifies the port that is connected to a video device or the uplink port that is connected to another trusted switch or router in the network interior, and enters interface configuration mode.
Step 3	Use one of the following: • **auto qos voip** {**cisco-phone** | **cisco-softphone** | **trust**} • **auto qos video** {**cts** | **ip-camera** | **media-player**} • **auto qos classify** [**police**] • **auto qos trust** {**cos** | **dscp**} **Example:** Switch(config-if)# **auto qos trust dscp**	Enables auto-QoS for VoIP. • **cisco-phone**—If the port is connected to a Cisco IP Phone, the QoS labels of incoming packets are trusted only when the telephone is detected. • **cisco-softphone**—The port is connected to device running the Cisco SoftPhone feature. • **trust**—The uplink port is connected to a trusted switch or router, and the VoIP traffic classification in the ingress packet is trusted. Enables auto-QoS for a video device. • **cts**—A port connected to a Cisco Telepresence system. • **ip-camera**—A port connected to a Cisco video surveillance camera. • **media-player**—A port connected to a CDP-capable Cisco digital media player. QoS labels of incoming packets are trusted only when the system is detected. Enables auto-QoS for classification. • **police**—Policing is set up by defining the QoS policy maps and applying them to ports (port-based QoS). Enables auto-QoS for trusted interfaces. • **cos**—Class of service. • **dscp**—Differentiated Services Code Point. • <cr>—Trust interface. **Note** To view a list of commands that are automatically generated by issuing one of the auto-QoS commands listed here, you need to be in debug mode. Refer to the *Catalyst 2960-XR Switch QoS Command Reference Guide* for examples of how to run the appropriate debug command to view a list of these commands.

		Command or Action	Purpose
Step 4		**exit** **Example:** Switch(config-if)# **exit**	Returns to global configuration mode.
Step 5		**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet 2/0/1**	Specifies the switch port identified as connected to a trusted switch or router, and enters interface configuration mode.
Step 6		**auto qos trust** **Example:** Switch(config-if)# **auto qos trust**	Enables auto-QoS on the port, and specifies that the port is connected to a trusted router or switch.
Step 7		**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 8		**show auto qos interface** *interface-id* **Example:** Switch# **show auto qos interface gigabitethernet 2/0/1**	Verifies your entries. This command displays the auto-QoS command on the interface on which auto-QoS was enabled. You can use the **show running-config** privileged EXEC command to display the auto-QoS configuration and the user modifications.

Troubleshooting Auto-QoS

To display the QoS commands that are automatically generated when auto-QoS is enabled or disabled, enter the **debug auto qos** privileged EXEC command before you enable auto-QoS. For more information, see the **debug auto qos** command in the command reference for this release.

To disable auto-QoS on a port, use the **no** form of the **auto qos** command interface configuration command, such as **no auto qos voip**.

Note Auto-QoS generated global commands can also be removed manually if desired.

Only the auto-QoS-generated interface configuration commands for this port are removed. If this is the last port on which auto-QoS is enabled and you enter the **no auto qos voip** command, auto-QoS is considered disabled even though the auto-QoS-generated global configuration commands remain (to avoid disrupting traffic on other ports affected by the global configuration).

You can use the **no mls qos** global configuration command to disable the auto-QoS-generated global configuration commands. With QoS disabled, there is no concept of trusted or untrusted ports because 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 (packets are switched without any rewrites and classified as best effort without any policing).

Monitoring Auto-QoS

Table 62: Commands for Monitoring Auto-QoS

Command	Description
show auto qos [**interface** [*interface-type*]]	Displays the initial auto-QoS configuration. You can compare the **show auto qos** and the **show running-config** command output to identify the user-defined QoS settings.
show mls qos [**aggregate policer** | **interface** | **maps** | **queue-set** | **stack-port** | **stack-qset** | **vlan**]	Displays information about the QoS configuration that might be affected by auto-QoS.
show mls qos aggregate policer *policer_name*	Displays information about the QoS aggregate policer configuration that might be affected by auto-QoS.
show mls qos interface [*interface-type* | **buffers** | **policers** | **queueing** | **statistics**]	Displays information about the QoS interface configuration that might be affected by auto-QoS.
show mls qos maps [**cos-dscp** | **cos-output-q** | **dscp-cos** | **dscp-mutation** | **dscp-output-q** | **ip-prec-dscp** | **policed-dscp**]	Displays information about the QoS maps configuration that might be affected by auto-QoS.
show mls qos queue-set *queue-set ID*	Displays information about the QoS queue-set configuration that might be affected by auto-QoS.
show mls qos stack-port buffers	Displays information about the QoS stack port buffer configuration that might be affected by auto-QoS.
show mls qos stack-qset	Displays information about the QoS stack queue set configuration that might be affected by auto-QoS.
show running-config	Displays information about the QoS configuration that might be affected by auto-QoS. You can compare the **show auto qos** and the **show running-config** command output to identify the user-defined QoS settings.

Configuration Examples for Auto-Qos

Examples: Global Auto-QoS Configuration

The following table describes the automatically generated commands for auto-QoS and enhanced auto-QoS by the switch.

Table 63: Generated Auto-QoS Configuration

Description	Automatically Generated Command {voip}	Enhanced Automatically Generated Command {Video|Trust|Classify}

The switch automatically enables standard QoS and configures the CoS-to-DSCP map (maps CoS values in incoming packets to a DSCP value).	```Switch(config)# mls qos```
	```Switch(config)# mls qos map```
	```cos-dscp```
	```0 8 16 26 32 46 48 56```
		```Switch(config)# mls qos```
		```Switch(config)# mls qos map```
		```cos-dscp```
		```0 8 16 24 32 46 48 56```

The switch automatically maps CoS values to an egress queue and to a threshold ID.	```Switch(config)# no mls qos```
	```srr-queue```
	```output cos-map```
	```Switch(config)# mls qos```
	```srr-queue```
	```output cos-map queue 1```
	```threshold 3 5```
	```Switch(config)# mls qos```
	```srr-queue```
	```output cos-map queue 2```
	```threshold 3 3```
	```6 7```
	```Switch(config)# mls qos```
	```srr-queue```
	```output cos-map queue 3```
	```threshold 3 2```
	```4```
	```Switch(config)# mls qos```
	```srr-queue```
	```output cos-map queue 4```
	```threshold 2 1```
	```Switch(config)# mls qos```
	```srr-queue```
	```output cos-map queue 4```
	```threshold 3 0```
		```Switch(config)# no mls qos```
		```srr-queue```
		```output cos-map```
		```Switch(config)# mls qos```
		```srr-queue```
		```output cos-map queue 1```
		```threshold 3 4 5```
		```Switch(config)# mls qos```
		```srr-queue```
		```output cos-map queue 2```
		```threshold 3 6 7```
		```Switch(config)# mls qos```
		```srr-queue```
		```output cos-map queue 2```
		```threshold 1 2```
		```Switch(config)# mls qos```
		```srr-queue```
		```output cos-map queue 2```
		```threshold 2 3```
		```Switch(config)# mls qos```
		```srr-queue```
		```output cos-map queue 3```
		```threshold 3 0```
		
		```Switch(config)# mls qos```
		```srr-queue```
		```output cos-map queue 4```
		```threshold 3 1```

Description	Automatically Generated Command {voip}	Enhanced Automatically Generated Command {Video|Trust|Classify}
The switch automatically maps DSCP values to an egress queue and to a threshold ID.	Switch(config)# **no mls qos srr-queue output dscp-map** Switch(config)# **mls qos srr-queue output dscp-map queue 1 threshold 3** **40 41 42 43 44 45 46 47** Switch(config)# **mls qos srr-queue output dscp-map queue 2 threshold 3** **24 25 26 27 28 29 30 31** Switch(config)# **mls qos srr-queue output dscp-map queue 2 threshold 3** **48 49 50 51 52 53 54 55** Switch(config)# **mls qos srr-queue output dscp-map queue 2 threshold 3** **56 57 58 59 60 61 62 63** Switch(config)# **mls qos srr-queue output dscp-map queue 3 threshold 3** **16 17 18 19 20 21 22 23** Switch(config)# **mls qos srr-queue output dscp-map queue 3 threshold 3** **32 33 34 35 36 37 38 39** Switch(config)# **mls qos srr-queue output dscp-map queue 4 threshold 1 8** Switch(config)# **mls qos srr-queue output dscp-map queue 4 threshold 2 9** **10 11 12 13 14 15** Switch(config)# **mls qos srr-queue output dscp-map queue 4 threshold 3 0 1 2 3 4 5 6 7**	Switch(config)# **no mls qos srr-queue output dscp-map** Switch(config)# **mls qos srr-queue output dscp-map queue 1 threshold 3 32** **33 40 41 42 43 44 45 46 47** Switch(config)# **mls qos srr-queue output dscp-map queue 2 threshold 1 16** **17 18 19 20 21 22 23** Switch(config)# **mls qos srr-queue output dscp-map queue 2 threshold 1 26** **27 28 29 30 31 34 35 36 37 38 39** Switch(config)# **mls qos srr-queue output dscp-map queue 2 threshold 2 24** Switch(config)# **mls qos srr-queue output dscp-map queue 2 threshold 3 48** **49 50 51 52 53 54 55 56** Switch(config)# **mls qos srr-queue output dscp-map queue 2 threshold 3 57** **58 59 60 61 62 63** Switch(config)# **mls qos srr-queue output dscp-map queue 3 threshold 3 0** **1 2 3 4 5 6 7** Switch(config)# **mls qos srr-queue output dscp-map queue 4 threshold 1 8** **9 11 13 15** Switch(config)# **mls qos srr-queue output dscp-map queue 4 threshold 2 10** **12 14**

Description	Automatically Generated Command {voip}	Enhanced Automatically Generated Command {Video|Trust|Classify}
The switch automatically configures the egress queue buffer sizes. It configures the bandwidth and the SRR mode (shaped or shared) on the egress queues mapped to the port.	Switch(config)# **mls qos queue-set output 1 threshold 1 138 138 92 138** Switch(config)# **mls qos queue-set output 1 threshold 2 138 138 92 400** Switch(config)# **mls qos queue-set output 1 threshold 3 36 77 100 318** Switch(config)# **mls qos queue-set output 1 threshold 4 20 50 67 400** Switch(config)# **mls qos queue-set output 2 threshold 1 149 149 100 149** Switch(config)# **mls qos queue-set output 2 threshold 2 118 118 100 235** Switch(config)# **mls qos queue-set output 2 threshold 3 41 68 100 272** Switch(config)# **mls qos queue-set output 2 threshold 4 42 72 100 242** Switch(config)# **mls qos queue-set output 1 buffers 10 10 26 54** Switch(config)# **mls qos queue-set output 2 buffers 16 6 17 61** Switch(config-if)# **priority-queue out** Switch(config-if)# **srr-queue bandwidth share 10 10 60 20**	Switch(config)# **mls qos queue-set output 1 threshold 2 100 100 50 200** Switch(config)# **mls qos queue-set output 1 threshold 2 125 125 100 400** Switch(config)# **mls qos queue-set output 1 threshold 3 100 100 100 400** Switch(config)# **mls qos queue-set output 1 threshold 4 60 150 50 200** Switch(config)# **mls qos queue-set output 1 buffers 15 25 40 20**

Examples: Auto-QoS Generated Configuration for VoIP Devices

If you entered the **auto qos voip cisco-phone** command, the switch automatically enables the trusted boundary feature, which uses the CDP to detect the presence or absence of a Cisco IP Phone.

```
Switch(config-if)# mls qos trust device cisco-phone
```

If you entered the **auto qos voip cisco-softphone** command, the switch automatically creates class maps and policy maps.

```
Switch(config)# mls qos map policed-dscp 24 26 46 to 0
Switch(config)# class-map match-all AutoQoS-VoIP-RTP-Trust
Switch(config-cmap)# match ip dscp ef
Switch(config)# class-map match-all AutoQoS-VoIP-Control-Trust
Switch(config-cmap)# match ip dscp cs3 af31
Switch(config)# policy-map AutoQoS-Police-SoftPhone
Switch(config-pmap)# class AutoQoS-VoIP-RTP-Trust
Switch(config-pmap-c)# set dscp ef
Switch(config-pmap-c)# police 320000 8000 exceed-action policed-dscp-transmit
Switch(config-pmap)# class AutoQoS-VoIP-Control-Trust
Switch(config-pmap-c)# set dscp cs3
Switch(config-pmap-c)# police 32000 8000 exceed-action policed-dscp-transmit
```

After creating the class maps and policy maps, the switch automatically applies the policy map called *AutoQoS-Police-SoftPhone* to an ingress interface on which auto-QoS with the Cisco SoftPhone feature is enabled.

```
Switch(config-if)#service-policy input AutoQoS-Police-SoftPhone
```

If you entered the **auto qos voip cisco-phone** command, the switch automatically creates class maps and policy maps.

```
Switch(config-if)# mls qos trust device cisco-phone
```

If you entered the **auto qos voip cisco-softphone** command, the switch automatically creates class maps and policy maps.

```
Switch(config)# mls qos map policed-dscp 24 26 46 to 0
Switch(config)# class-map match-all AutoQoS-VoIP-RTP-Trust
Switch(config-cmap)# match ip dscp ef
Switch(config)# class-map match-all AutoQoS-VoIP-Control-Trust
Switch(config-cmap)# match ip dscp cs3 af31
Switch(config)# policy-map AutoQoS-Police-CiscoPhone
Switch(config-pmap)# class AutoQoS-VoIP-RTP-Trust
Switch(config-pmap-c)# set dscp ef
Switch(config-pmap-c)# police 320000 8000 exceed-action policed-dscp-transmit
Switch(config-pmap)# class AutoQoS-VoIP-Control-Trust
Switch(config-pmap-c)# set dscp cs3
Switch(config-pmap-c)# police 32000 8000 exceed-action policed-dscp-transmit
```

After creating the class maps and policy maps, the switch automatically applies the policy map called *AutoQoS-Police-SoftPhone* to an ingress interface on which auto-QoS with the Cisco SoftPhone feature is enabled.

```
Switch(config-if)# service-policy input AutoQoS-Police-SoftPhone
```

Examples: Auto-QoS Generated Configuration For Enhanced Video, Trust, and Classify Devices

If you entered the following enhanced auto-QoS commands, the switch configures a CoS-to-DSCP map (maps CoS values in incoming packets to a DSCP value):

- **auto qos video cts**

- **auto qos video ip-camera**

- **auto qos video media-player**

- **auto qos trust**

- **auto qos trust cos**

- **auto qos trust dscp**

  The following command is initiated after entering one of the above auto-QoS commands:

  ```
  Switch(config)# mls qos map cos-dscp 0 8 16 24 32 46 48 56
  ```

  Note No class maps and policy maps are configured.

If you entered the **auto qos classify** command, the switch automatically creates class maps and policy maps (as shown below).

```
Switch(config)# mls qos map policed-dscp 0 10 18 24 26 46 to 8
Switch(config)# mls qos map cos-dscp 0 8 16 24 32 46 48 56
Switch(config)# class-map match-all AUTOQOS_MULTIENHANCED_CONF_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-MULTIENHANCED-CONF
Switch(config)# class-map match-all AUTOQOS_DEFAULT_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-DEFAULT
Switch(config)# class-map match-all AUTOQOS_TRANSACTION_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-TRANSACTIONAL-DATA
Switch(config)# class-map match-all AUTOQOS_SIGNALING_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-SIGNALING
Switch(config)# class-map match-all AUTOQOS_BULK_DATA_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-BULK-DATA
Switch(config)# class-map match-all AUTOQOS_SCAVANGER_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-SCAVANGER
Switch(config)# policy-map AUTOQOS-SRND4-CLASSIFY-POLICY
Switch(config-pmap)# class AUTOQOS_MULTIENHANCED_CONF_CLASS
Switch(config-pmap-c)# set dscp af41
Switch(config-pmap)# class AUTOQOS_BULK_DATA_CLASS
Switch(config-pmap-c)# set dscp af11
Switch(config-pmap)# class AUTOQOS_TRANSACTION_CLASS
Switch(config-pmap-c)# set dscp af21
Switch(config-pmap)# class AUTOQOS_SCAVANGER_CLASS
Switch(config-pmap-c)# set dscp cs1
Switch(config-pmap)# class AUTOQOS_SIGNALING_CLASS
Switch(config-pmap-c)# set dscp cs3
Switch(config-pmap)# class AUTOQOS_DEFAULT_CLASS
Switch(config-pmap-c)# set dscp default
;
```

```
Switch(config-if)# service-policy input AUTOQOS-SRND4-CLASSIFY-POLICY
```

If you entered the **auto qos classify police** command, the switch automatically creates class maps and policy maps (as shown below).

```
Switch(config)# mls qos map policed-dscp 0 10 18 24 26 46 to 8
Switch(config)# mls qos map cos-dscp 0 8 16 24 32 46 48 56
Switch(config)# class-map match-all AUTOQOS_MULTIENHANCED_CONF_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-MULTIENHANCED-CONF
Switch(config)# class-map match-all AUTOQOS_DEFAULT_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-DEFAULT
Switch(config)# class-map match-all AUTOQOS_TRANSACTION_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-TRANSACTIONAL-DATA
Switch(config)# class-map match-all AUTOQOS_SIGNALING_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-SIGNALING
Switch(config)# class-map match-all AUTOQOS_BULK_DATA_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-BULK-DATA
Switch(config)# class-map match-all AUTOQOS_SCAVANGER_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-SCAVANGER
Switch(config)# policy-map AUTOQOS-SRND4-CLASSIFY-POLICE-POLICY
Switch(config-pmap)# class AUTOQOS_MULTIENHANCED_CONF_CLASS
Switch(config-pmap-c)# set dscp af41
Switch(config-pmap-c)# police 5000000 8000 exceed-action drop
Switch(config-pmap)# class AUTOQOS_BULK_DATA_CLASS
Switch(config-pmap-c)# set dscp af11
Switch(config-pmap-c)# police 10000000 8000 exceed-action policed-dscp-transmit
Switch(config-pmap)# class AUTOQOS_TRANSACTION_CLASS
Switch(config-pmap-c)# set dscp af21
Switch(config-pmap-c)# police 10000000 8000 exceed-action policed-dscp-transmit
Switch(config-pmap)# class AUTOQOS_SCAVANGER_CLASS
Switch(config-pmap-c)# set dscp cs1
Switch(config-pmap-c)# police 10000000 8000 exceed-action drop
Switch(config-pmap)# class AUTOQOS_SIGNALING_CLASS
Switch(config-pmap-c)# set dscp cs3
Switch(config-pmap-c)# police 32000 8000 exceed-action drop
Switch(config-pmap)# class AUTOQOS_DEFAULT_CLASS
Switch(config-pmap-c)# set dscp default
Switch(config-pmap-c)# police 10000000 8000 exceed-action policed-dscp-transmit
;
Switch(config-if)# service-policy input AUTOQOS-SRND4-CLASSIFY-POLICE-POLICY
```

This is the enhanced configuration for the **auto qos voip cisco-phone** command:

```
Switch(config)# mls qos map policed-dscp 0 10 18 24 26 46 to 8
Switch(config)# mls qos map cos-dscp 0 8 16 24 32 46 48 56
Switch(config)# class-map match-all AUTOQOS_VOIP_DATA_CLASS
Switch(config-cmap)# match ip dscp ef
Switch(config)# class-map match-all AUTOQOS_DEFAULT_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-DEFAULT
Switch(config)# class-map match-all AUTOQOS_VOIP_SIGNAL_CLASS
Switch(config-cmap)# match ip dscp cs3
Switch(config)# policy-map AUTOQOS-SRND4-CISCOPHONE-POLICY
Switch(config-pmap)# class AUTOQOS_VOIP_DATA_CLASS
Switch(config-pmap-c)# set dscp ef
Switch(config-pmap-c)# police 128000 8000 exceed-action policed-dscp-transmit
Switch(config-pmap)# class AUTOQOS_VOIP_SIGNAL_CLASS
Switch(config-pmap-c)# set dscp cs3
Switch(config-pmap-c)# police 32000 8000 exceed-action policed-dscp-transmit
Switch(config-pmap)# class AUTOQOS_DEFAULT_CLASS
```

```
Switch(config-pmap-c)# set dscp default
Switch(config-pmap-c)# police 10000000 8000 exceed-action policed-dscp-transmit
;
Switch(config-if)# service-policy input AUTOQOS-SRND4-CISCOPHONE-POLICY
```

This is the enhanced configuration for the **auto qos voip cisco-softphone** command:

```
Switch(config)# mls qos map policed-dscp 0 10 18 24 26 46 to 8
Switch(config)# mls qos map cos-dscp 0 8 16 24 32 46 48 56
Switch(config)# class-map match-all AUTOQOS_MULTIENHANCED_CONF_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-MULTIENHANCED-CONF
Switch(config)# class-map match-all AUTOQOS_VOIP_DATA_CLASS
Switch(config-cmap)# match ip dscp ef
Switch(config)# class-map match-all AUTOQOS_DEFAULT_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-DEFAULT
Switch(config)# class-map match-all AUTOQOS_TRANSACTION_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-TRANSACTIONAL-DATA
Switch(config)# class-map match-all AUTOQOS_VOIP_SIGNAL_CLASS
Switch(config-cmap)# match ip dscp cs3
Switch(config)# class-map match-all AUTOQOS_SIGNALING_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-SIGNALING
Switch(config)# class-map match-all AUTOQOS_BULK_DATA_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-BULK-DATA
Switch(config)# class-map match-all AUTOQOS_SCAVANGER_CLASS
Switch(config-cmap)# match access-group name AUTOQOS-ACL-SCAVANGER

Switch(config)# policy-map AUTOQOS-SRND4-SOFTPHONE-POLICY
Switch(config-pmap)# class AUTOQOS_VOIP_DATA_CLASS
Switch(config-pmap-c)# set dscp ef
Switch(config-pmap-c)# police 128000 8000 exceed-action policed-dscp-transmit
Switch(config-pmap)# class AUTOQOS_VOIP_SIGNAL_CLASS
Switch(config-pmap-c)# set dscp cs3
Switch(config-pmap-c)# police 32000 8000 exceed-action policed-dscp-transmit
Switch(config-pmap)#class AUTOQOS_MULTIENHANCED_CONF_CLASS
Switch(config-pmap-c)#set dscp af41
Switch(config-pmap-c)# police 5000000 8000 exceed-action drop
Switch(config-pmap)# class AUTOQOS_BULK_DATA_CLASS
Switch(config-pmap-c)# set dscp af11
Switch(config-pmap-c)# police 10000000 8000 exceed-action policed-dscp-transmit
Switch(config-pmap)# class AUTOQOS_TRANSACTION_CLASS
Switch(config-pmap-c)# set dscp af21
Switch(config-pmap-c)# police 10000000 8000 exceed-action policed-dscp-transmit
Switch(config-pmap)# class AUTOQOS_SCAVANGER_CLASS
Switch(config-pmap-c)# set dscp cs1
Switch(config-pmap-c)# police 10000000 8000 exceed-action drop
Switch(config-pmap)# class AUTOQOS_SIGNALING_CLASS
Switch(config-pmap-c)# set dscp cs3
Switch(config-pmap-c)# police 32000 8000 exceed-action drop
Switch(config-pmap)# class AUTOQOS_DEFAULT_CLASS
Switch(config-pmap-c)# set dscp default
;
Switch(config-if)# service-policy input AUTOQOS-SRND4-SOFTPHONE-POLICY
```

Where to Go Next for Auto-QoS

Review the QoS documentation if you require any specific QoS changes to your auto-QoS configuration.

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this book.	*Catalyst 2960-XR Switch Quality of Service Command Reference*

Standards and RFCs

Standard/RFC	Title
—	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Auto-QoS

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

PART VII

Routing

Configuring IP Unicast Routing

Information About Configuring IP Unicast Routing

This module describes how to configure IP Version 4 (IPv4) unicast routing on the switch.

A switch stack operates and appears as a single router to the rest of the routers in the network. Basic routing functions like static routing are available with IP Lite .

Note In addition to IPv4 traffic, you can also enable IP Version 6 (IPv6) unicast routing and configure interfaces to forward IPv6 traffic.

Information About 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 (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 66: Routing Topology Example

This figure shows a basic routing topology. Switch A is in VLAN 10, and Switch B is in VLAN 20. The router
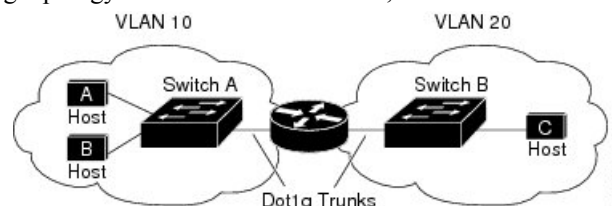
has an interface in each VLAN.

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 these ways:

- By using default routing

- By using preprogrammed static routes for the traffic

- By dynamically calculating routes by using a routing protocol

Default routing refers to sending traffic with a destination unknown to the router to a default outlet or destination.

Static unicast routing forwards packets from predetermined ports through a single path into and out of a network. Static routing is secure and uses little bandwidth, but does not automatically respond to changes in the network, such as link failures, and therefore, might result in unreachable destinations. As networks grow, static routing becomes a labor-intensive liability.

Switches running the LAN base feature set support 16 user-configured static routes, in addition to any default routes used for the management interface. The LAN base image supports static routing only on SVIs.

Dynamic routing protocols are used by routers to dynamically calculate the best route for forwarding traffic. There are two types of dynamic routing protocols:

- Routers using distance-vector protocols maintain routing tables with distance values of networked resources, and periodically pass these tables to their neighbors. Distance-vector protocols use one or a series of metrics for calculating the best routes. These protocols are easy to configure and use.

- Routers using link-state protocols maintain a complex database of network topology, based on the exchange of link-state advertisements (LSAs) between routers. LSAs are triggered by an event in the network, which speeds up the convergence time or time required to respond to these changes. Link-state protocols respond quickly to topology changes, but require greater bandwidth and more resources than distance-vector protocols.

Distance-vector protocols supported by the switch are Routing Information Protocol (RIP), which uses a single distance metric (cost) to determine the best path. The switch also supports the Open Shortest Path First (OSPF) link-state protocol, which adds some link-state routing features to traditional Interior Gateway Routing Protocol (IGRP) to improve efficiency.

IP Routing and Switch Stacks

A switch stack appears to the network as a single switch, regardless of which switch in the stack is connected to a routing peer.

The active switch performs these functions:

- It initializes and configures the routing protocols.

- It sends routing protocol messages and updates to other routers.

- It processes routing protocol messages and updates received from peer routers.

- It generates, maintains, and distributes the distributed Cisco Express Forwarding (dCEF) database to all stack members. The routes are programmed on all switches in the stack bases on this database.

- The MAC address of the active switch is used as the router MAC address for the whole stack, and all outside devices use this address to send IP packets to the stack.

- All IP packets that require software forwarding or processing go through the CPU of the active switch.

Stack members perform these functions:

- They act as routing standby switches, ready to take over in case they are elected as the new active switch if the active switch fails.

- They program the routes into hardware.

If a active switch fails, the stack detects that the active switch is down and elects one of the stack members to be the new active switch. During this period, except for a momentary interruption, the hardware continues to forward packets with no active protocols.

However, even though the switch stack maintains the hardware identification after a failure, the routing protocols on the router neighbors might flap during the brief interruption before the active switch restarts. Routing protocols such as OSPF and EIGRP need to recognize neighbor transitions.

Upon election, the new active switch performs these functions:

- It starts generating, receiving, and processing routing updates.

- It builds routing tables, generates the CEF database, and distributes it to stack members.

- It uses its MAC address as the router MAC address. To notify its network peers of the new MAC address, it periodically (every few seconds for 5 minutes) sends a gratuitous ARP reply with the new router MAC address.

Note If you configure the persistent MAC address feature on the stack and the active switch changes, the stack MAC address does not change for the configured time period. If the previous active switch rejoins the stack as a member switch during that time period, the stack MAC address remains the MAC address of the previous active switch.

• It attempts to determine the reachability of every proxy ARP entry by sending an ARP request to the proxy ARP IP address and receiving an ARP reply. For each reachable proxy ARP IP address, it generates a gratuitous ARP reply with the new router MAC address. This process is repeated for 5 minutes after a new active switch election.

Caution Partitioning of the switch stack into two or more stacks might lead to undesirable behavior in the network.

If the switch is reloaded, then all the ports on that switch go down and there is a loss of traffic for the interfaces involved in routing.

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 the figure, 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 67: IP Classless Routing
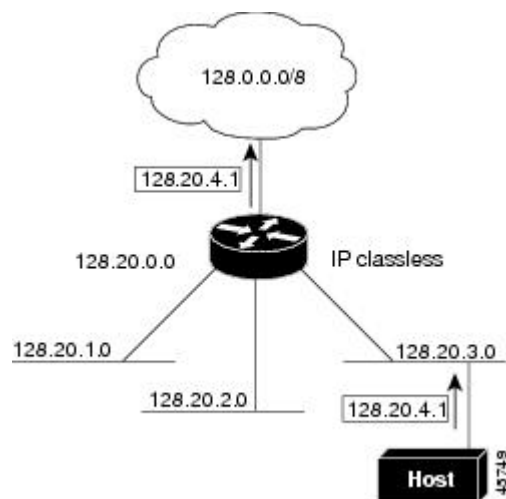
In the figure , the router in network 128.20.0.0 is connected to subnets 128.20.1.0, 128.20.2.0, and 128.20.3.0. If the host sends a packet to 120.20.4.1, because there is no network default route, the router discards the packet.

Figure 68: No IP Classless Routing
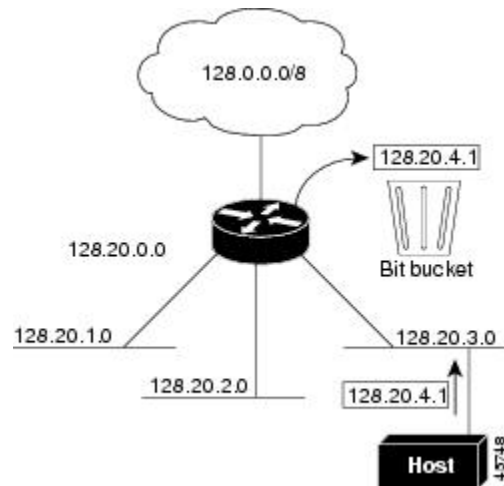
To prevent the Switch from forwarding packets destined for unrecognized subnets to the best supernet route possible, you can disable classless routing behavior.

Address Resolution

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.

Note In a switch stack, network communication uses a single MAC address and the IP address of the stack.

The local address or MAC address is known as a data link address because it is contained in the data link layer (Layer 2) section of the packet header and is read by data link (Layer 2) devices. 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 the *Cisco IOS Configuration Fundamentals Configuration Guide*

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.

ICMP Router Discovery Protocol

Router discovery allows the Switch to dynamically learn about routes to other networks using ICMP router discovery protocol (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.

UDP Broadcast Packets and Protocols

User Datagram Protocol (UDP) is an IP host-to-host layer protocol, as is TCP. UDP 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 remedy this situation by configuring 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. The description for the **ip forward-protocol** interface configuration command in the *Cisco IOS IP Command Reference, Volume 1 of 3: Addressing and Services* lists the ports that are forwarded by default if you do not specify any UDP ports.

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. Many implementations, including the one in the Switch, support several addressing schemes for forwarding broadcast messages.

IP Broadcast Flooding

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, it still can receive broadcasts. However, the interface never forwards 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. Thus, the destination address 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.

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.

How to Configure IP Routing

By default, IP routing is disabled on the Switch, and you must enable it before routing can take place. For detailed IP routing configuration information, see the *Cisco IOS IP Configuration Guide*.

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. For more information, see the "Configuring Layer 3 EtherChannels" chapter in the Layer 2 Configuration Guide.

> **Note** 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.

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

Configuring routing consists of several main procedures:

- To support VLAN interfaces, create and configure VLANs on the Switch or switch stack, and assign VLAN membership to Layer 2 interfaces. For more information, see the "Configuring VLANs" chapter in the VLAN Configuration Guide.

- Configure Layer 3 interfaces.

- Enable IP routing on the switch.

- Assign IP addresses to the Layer 3 interfaces.

- Enable selected routing protocols on the switch.

- Configure routing protocol parameters (optional).

How to Configure IP Addressing

A required task for configuring IP routing is to assign IP addresses to Layer 3 network interfaces to enable the interfaces and allow communication with the hosts on those interfaces that use IP. The following 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

- Assigning IP Addresses to Network Interfaces

- Configuring Address Resolution Methods

- Routing Assistance When IP Routing is Disabled

- Configuring Broadcast Packet Handling

- Monitoring and Maintaining IP Addressing

Default IP Addressing Configuration

Table 64: 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.

Feature	Default Setting
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.
	Spanning Tree Protocol (STP): 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. 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 IP addresses.

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. To receive an assigned network number, contact your Internet service provider.

Procedure

	Command or Action	**Purpose**
Step 1	**enable**	Enables privileged EXEC mode.
	Example:	• Enter your password if prompted.
	Switch> **enable**	

	Command or Action	**Purpose**
Step 2	**configure terminal**	Enters global configuration mode.
	Example:	
	Switch# **configure terminal**	
Step 3	**interface** *interface-id*	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
	Example:	
	Switch(config)# interface gigabitethernet 1/0/1	
Step 4	**no switchport**	Removes the interface from Layer 2 configuration mode (if it is a physical interface).
	Example:	
	Switch(config-if)# no switchport	
Step 5	**ip address** *ip-address subnet-mask*	Configures the IP address and IP subnet mask.
	Example:	
	Switch(config-if)# ip address 10.1.5.1 255.255.255.0	
Step 6	**no shutdown**	Enables the physical interface.
	Example:	
	Switch(config-if)# no shutdown	
Step 7	**end**	Returns to privileged EXEC mode.
	Example:	
	Switch(config)# **end**	
Step 8	**show ip route**	Verifies your entries.
	Example:	
	Switch# show ip route	
Step 9	**show ip interface** [*interface-id*]	Verifies your entries.
	Example:	
	Switch# show ip interface gigabitethernet 1/0/1	
Step 10	**show running-config**	Verifies your entries.
	Example:	
	Switch# **show running-config**	

	Command or Action	**Purpose**
Step 11	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Using Subnet Zero

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.

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.

Use the **no ip subnet-zero** global configuration command to restore the default and disable the use of subnet zero.

Procedure

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**ip subnet-zero** **Example:** Switch(config)# ip subnet-zero	Enables the use of subnet zero for interface addresses and routing updates.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.

	Command or Action	**Purpose**
Step 6	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.
	Example:	
	Switch# **copy running-config startup-config**	

Disabling 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.

Procedure

	Command or Action	**Purpose**
Step 1	**enable**	Enables privileged EXEC mode.
	Example:	• Enter your password if prompted.
	Switch> **enable**	
Step 2	**configure terminal**	Enters global configuration mode.
	Example:	
	Switch# **configure terminal**	
Step 3	**no ip classless**	Disables classless routing behavior.
	Example:	
	Switch(config)#no ip classless	
Step 4	**end**	Returns to privileged EXEC mode.
	Example:	
	Switch(config)# **end**	
Step 5	**show running-config**	Verifies your entries.
	Example:	
	Switch# **show running-config**	
Step 6	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.
	Example:	

Command or Action	Purpose
Switch# **copy running-config startup-config**	

Configuring Address Resolution Methods

You can perform the following tasks to configure address resolution.

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.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**arp** *ip-address hardware-address type* **Example:** Switch(config)# ip 10.1.5.1 c2f3.220a.12f4 arpa	Associates an IP address with a MAC (hardware) address in the ARP cache, and specifies 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
Step 4	**arp** *ip-address hardware-address type* [**alias**] **Example:** Switch(config)# ip 10.1.5.3 d7f3.220d.12f5 arpa alias	(Optional) Specifies that the switch respond to ARP requests as if it were the owner of the specified IP address.

	Command or Action	Purpose
Step 5	**interface** *interface-id* **Example:** Switch(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the interface to configure.
Step 6	**arp** *timeout seconds* **Example:** Switch(config-if)# arp 20000	(Optional) Sets 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.
Step 7	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 8	**show interfaces** [*interface-id*] **Example:** Switch# show interfaces gigabitethernet 1/0/1	Verifies the type of ARP and the timeout value used on all interfaces or a specific interface.
Step 9	**show arp** **Example:** Switch# show arp	Views the contents of the ARP cache.
Step 10	**show ip arp** **Example:** Switch# show ip arp	Views the contents of the ARP cache.
Step 11	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

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.

To disable an encapsulation type, use the **no arp arpa** or **no arp snap** interface configuration command.

Procedure

	Command or Action	Purpose
Step 1	**enable** Example: Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* Example: Switch(config)# interface gigabitethernet 1/0/2	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 4	**arp** {**arpa** | **snap**} Example: Switch(config-if)# arp arpa	Specifies the ARP encapsulation method: • **arpa**—Address Resolution Protocol • **snap**—Subnetwork Address Protocol
Step 5	**end** Example: Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**show interfaces** [*interface-id*] Example: Switch# show interfaces	Verifies ARP encapsulation configuration on all interfaces or the specified interface.
Step 7	**copy running-config startup-config** Example: Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Enabling Proxy ARP

By default, the Switch uses proxy ARP to help hosts learn MAC addresses of hosts on other networks or subnets.

Procedure

	Command or Action	Purpose
Step 1	**enable** Example: Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* Example: Switch(config)# interface gigabitethernet 1/0/2	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 4	**ip proxy-arp** Example: Switch(config-if)# ip proxy-arp	Enables proxy ARP on the interface.
Step 5	**end** Example: Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**show ip interface** [*interface-id*] Example: Switch# show ip interface gigabitethernet 1/0/2	Verifies the configuration on the interface or all interfaces.
Step 7	**copy running-config startup-config** Example: Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

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
- Default Gateway

• ICMP Router Discovery Protocol (IRDP)

Proxy ARP

Proxy ARP is enabled by default. To enable it after it has been disabled, see the "Enabling Proxy ARP" section. 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 non-local 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.

Procedure

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**ip default-gateway** *ip-address* **Example:** Switch(config)# ip default gateway 10.1.5.1	Sets up a default gateway (router).
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show ip redirects** **Example:** Switch# show ip redirects	Displays the address of the default gateway router to verify the setting.
Step 6	**copy running-config startup-config** **Example:**	(Optional) Saves your entries in the configuration file.

Command or Action	Purpose
Switch# **copy running-config startup-config**	

ICMP Router Discovery Protocol (IRDP)

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. 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.

Procedure

	Command or Action	Purpose
Step 1	**enable** Example: Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* Example: Switch(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 4	**ip irdp** Example: Switch(config-if)# ip irdp	Enables IRDP processing on the interface.
Step 5	**ip irdp multicast** Example: Switch(config-if)# ip irdp multicast	(Optional) Sends IRDP advertisements to the multicast address (224.0.0.1) instead of IP broadcasts. **Note** This 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.

	Command or Action	**Purpose**
Step 6	**ip irdp holdtime** *seconds* **Example:** Switch(config-if)# ip irdp holdtime 1000	(Optional) Sets 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.
Step 7	**ip irdp maxadvertinterval** *seconds* **Example:** Switch(config-if)# ip irdp maxadvertinterval 650	(Optional) Sets the IRDP maximum interval between advertisements. The default is 600 seconds.
Step 8	**ip irdp minadvertinterval** *seconds* **Example:** Switch(config-if)# ip irdp minadvertinterval 500	(Optional) Sets 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**).
Step 9	**ip irdp preference** *number* **Example:** Switch(config-if)# ip irdp preference 2	(Optional) Sets a device IRDP preference level. The allowed range is –231 to 231. The default is 0. A higher value increases the router preference level.
Step 10	**ip irdp address** *address* [*number*] **Example:** Switch(config-if)# ip irdp address 10.1.10.10	(Optional) Specifies an IRDP address and preference to proxy-advertise.
Step 11	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 12	**show ip irdp** **Example:** Switch# show ip irdp	Verifies settings by displaying IRDP values.
Step 13	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Broadcast Packet Handling

Perform the tasks in these sections to enable these schemes:

- Enabling Directed Broadcast-to-Physical Broadcast Translation

- Forwarding UDP Broadcast Packets and Protocols

- Establishing an IP Broadcast Address

- Flooding IP Broadcasts

Enabling Directed Broadcast-to-Physical Broadcast Translation

By default, IP directed broadcasts are dropped; they are not forwarded. Dropping IP-directed broadcasts makes 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.

You can specify an access list to control which broadcasts are forwarded. When an access list is specified, only those IP packets permitted by the access list are eligible to be translated from directed broadcasts to physical broadcasts. For more information on access lists, see the "Information about Network Security with ACLs" section in the Security Configuration Guide.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. - Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# interface gigabitethernet 1/0/2	Enters interface configuration mode, and specifies the interface to configure.
Step 4	**ip directed-broadcast** [*access-list-number*] **Example:** Switch(config-if)# ip directed-broadcast 103	Enables 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, only IP packets permitted by the access list can be translated.
Step 5	**exit** **Example:** Switch(config-if)# exit	Returns to global configuration mode.

	Command or Action	Purpose
Step 6	**ip forward-protocol** {**udp** [*port*]	**nd**	**sdns**} **Example:** Switch(config)# ip forward-protocol nd	Specifies 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
Step 7	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 8	**show ip interface** [*interface-id*] **Example:** Switch# show ip interface	Verifies the configuration on the interface or all interfaces
Step 9	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 10	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Forwarding UDP Broadcast Packets and Protocols

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.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.

		Command or Action	Purpose
Step 2		**configure terminal**	Enters global configuration mode.
		Example:	
		Switch# **configure terminal**	
Step 3		**interface** *interface-id*	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
		Example:	
		Switch(config)# interface gigabitethernet 1/0/1	
Step 4		**ip helper-address** *address*	Enables forwarding and specifies the destination address for forwarding UDP broadcast packets, including BOOTP.
		Example:	
		Switch(config-if)# ip helper address 10.1.10.1	
Step 5		**exit**	Returns to global configuration mode.
		Example:	
		Switch(config-if)# exit	
Step 6		**ip forward-protocol** {**udp** [*port*] | **nd** | **sdns**}	Specifies which protocols the router forwards when forwarding broadcast packets.
		Example:	
		Switch(config)# ip forward-protocol sdns	
Step 7		**end**	Returns to privileged EXEC mode.
		Example:	
		Switch(config)# **end**	
Step 8		**show ip interface** [*interface-id*]	Verifies the configuration on the interface or all interfaces.
		Example:	
		Switch# show ip interface gigabitethernet 1/0/1	
Step 9		**show running-config**	Verifies your entries.
		Example:	
		Switch# **show running-config**	
Step 10		**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.
		Example:	
		Switch# **copy running-config startup-config**	

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.

Procedure

	Command or Action	**Purpose**
Step 1	**enable**	Enables privileged EXEC mode.
	Example:	• Enter your password if prompted.
	Switch> **enable**	
Step 2	**configure terminal**	Enters global configuration mode.
	Example:	
	Switch# **configure terminal**	
Step 3	**interface** *interface-id*	Enters interface configuration mode, and specifies the interface to configure.
	Example:	
	Switch(config)# interface gigabitethernet 1/0/1	
Step 4	**ip broadcast-address** *ip-address*	Enters a broadcast address different from the default, for example 128.1.255.255.
	Example:	
	Switch(config-if)# ip broadcast-address 128.1.255.255	
Step 5	**end**	Returns to privileged EXEC mode.
	Example:	
	Switch(config)# **end**	
Step 6	**show ip interface** [*interface-id*]	Verifies the broadcast address on the interface or all interfaces.
	Example:	
	Switch# show ip interface	
Step 7	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.
	Example:	
	Switch# **copy running-config startup-config**	

Flooding IP Broadcasts

Procedure

	Command or Action	Purpose
Step 1	**enable** Example: Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 3	**ip forward-protocol spanning-tree** Example: Switch(config)# ip forward-protocol spanning-tree	Uses the bridging spanning-tree database to flood UDP datagrams.
Step 4	**end** Example: Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show running-config** Example: Switch# **show running-config**	Verifies your entries.
Step 6	**copy running-config startup-config** Example: Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.
Step 7	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 8	**ip forward-protocol turbo-flood** Example:	Uses the spanning-tree database to speed up flooding of UDP datagrams.

	Command or Action	Purpose
	Switch(config)# ip forward-protocol turbo-flood	
Step 9	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 10	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 11	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

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. The Table lists the commands for clearing contents.

Table 65: Commands to Clear Caches, Tables, and Databases

clear arp-cache	Clears the IP ARP cache and the fast-switching cache.
clear host {*name*	*****}	Removes one or all entries from the hostname and the address cache.
clear ip route {**network** [*mask*]	*****}	Removes 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. The Table lists the privileged EXEC commands for displaying IP statistics.

Table 66: Commands to Display Caches, Tables, and Databases

show arp	Displays the entries in the ARP table.
show hosts	Displays the default domain name, style of lookup service, name server hosts, and the cached list of hostnames and addresses.
show ip aliases	Displays IP addresses mapped to TCP ports (aliases).

show ip arp	Displays the IP ARP cache.
show ip interface [*interface-id*]	Displays the IP status of interfaces.
show ip irdp	Displays IRDP values.
show ip masks *address*	Displays the masks used for network addresses and the number of subnets using each mask.
show ip redirects	Displays the address of a default gateway.
show ip route [*address* [*mask*]]	[*protocol*]	Displays the current state of the routing table.
show ip route summary	Displays the current state of the routing table in summary form.

How to Configure IP Unicast Routing

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, you must enable IP routing.

Procedure

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**ip routing** **Example:** Switch(config)# ip routing	Enables IP routing.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Example of Enabling IP Routing

This example shows how to enable IP routing :

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

Switch(config-router)# end
```

What to Do Next

You can now set up parameters for the selected routing protocols as described in these sections:

- RIP
- OSPF,
- EIGRP
- Unicast Reverse Path Forwarding
- Protocol-Independent Features (optional)

Information About RIP

The Routing Information Protocol (RIP) is an interior gateway protocol (IGP) created for use in small, homogeneous networks. It is a distance-vector routing protocol that uses broadcast User Datagram Protocol (UDP) data packets to exchange routing information. The protocol is documented in RFC 1058. You can find detailed information about RIP in *IP Routing Fundamentals,* published by Cisco Press.

Note   RIP is supported in the IP Lite.

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; it is treated by RIP as a network to implement the default routing feature. 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.

Summary Addresses and 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 usually optimizes communication among multiple routers, especially when links are broken.

How to Configure RIP

Default RIP Configuration

Table 67: 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 triggered	Disabled
IP split horizon	Varies with media.
Neighbor	None defined.
Network	None specified.
Offset list	Disabled.
Output delay	0 milliseconds.

Feature	Default Setting
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. On the Switch, RIP configuration commands are ignored until you configure the network number.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**ip routing** **Example:** Switch(config)# **ip routing**	Enables IP routing. (Required only if IP routing is disabled.)
Step 4	**router rip** **Example:** Switch(config)# **router rip**	Enables a RIP routing process, and enter router configuration mode.
Step 5	**network** *network number* **Example:** Switch(config-router)# **network 12.0.0.0**	Associates 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.

	Command or Action	**Purpose**
		Note You must configure a network number for the RIP commands to take effect.
Step 6	**neighbor** *ip-address* **Example:** Switch(config-router)# **neighbor 10.2.5.1**	(Optional) Defines a neighboring router with which to exchange routing information. This step allows routing updates from RIP (normally a broadcast protocol) to reach nonbroadcast networks.
Step 7	**offset-list** [*access-list number* | *name*] {**in** | **out**} *offset* [*type number*] **Example:** Switch(config-router)# **offset-list 103 in 10**	(Optional) Applies 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.
Step 8	**timers basic** *update invalid holddown flush* **Example:** Switch(config-router)# **timers basic 45 360 400 300**	(Optional) Adjusts 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.
Step 9	**version** {**1** | **2**} **Example:** Switch(config-router)# **version 2**	(Optional) Configures 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.
Step 10	**no auto-summary** **Example:** Switch(config-router)# **no auto-summary**	(Optional) Disables 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.
Step 11	**output-delay** *delay* **Example:** Switch(config-router)# **output-delay 8**	(Optional) Adds 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.
Step 12	**end** **Example:**	Returns to privileged EXEC mode.

	Command or Action	**Purpose**
	`Switch(config-router)# end`	
Step 13	**show ip protocols**	Verifies your entries.
	Example:	
	`Switch# show ip protocols`	
Step 14	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.
	Example:	
	`Switch# copy running-config startup-config`	

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.

The Switch supports two modes of authentication on interfaces for which RIP authentication is enabled: plain text and MD5. The default is plain text.

Procedure

	Command or Action	**Purpose**
Step 1	**enable**	Enables privileged EXEC mode.
	Example:	• Enter your password if prompted.
	`Switch> enable`	
Step 2	**configure terminal**	Enters global configuration mode.
	Example:	
	`Switch# configure terminal`	
Step 3	**interface** *interface-id*	Enters interface configuration mode, and specifies the interface to configure.
	Example:	
	`Switch(config)# interface gigabitethernet 1/0/1`	
Step 4	**ip rip authentication key-chain** *name-of-chain*	Enables RIP authentication.
	Example:	
	`Switch(config-if)# ip rip authentication key-chain trees`	

	Command or Action	**Purpose**
Step 5	**ip rip authentication mode** {**text** | **md5**} **Example:** Switch(config-if)# **ip rip authentication mode md5**	Configures the interface to use plain text authentication (the default) or MD5 digest authentication.
Step 6	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 7	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 8	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Summary Addresses and Split Horizon

> **Note**   In general, disabling split horizon is not recommended unless you are certain that your application requires it to properly advertise routes.

If you want 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.

Procedure

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet 1/0/1**	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 4	**ip address** *ip-address subnet-mask* **Example:** Switch(config-if)# **ip address 10.1.1.10 255.255.255.0**	Configures the IP address and IP subnet.
Step 5	**ip summary-address rip** *ip-address ip-network mask* **Example:** Switch(config-if)# **ip summary-address rip 10.1.1.30 255.255.255.0**	Configures the IP address to be summarized and the IP network mask.
Step 6	**no ip split-horizon** **Example:** Switch(config-if)# **no ip split-horizon**	Disables split horizon on the interface.
Step 7	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 8	**show ip interface** *interface-id* **Example:** Switch# **show ip interface gigabitethernet 1/0/1**	Verifies your entries.
Step 9	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

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, especially when links are broken.

✎

Note In general, we do not recommend disabling split horizon unless you are certain that your application requires it to properly advertise routes.

Procedure

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the interface to configure.
Step 4	**ip address** *ip-address subnet-mask* **Example:** Switch(config-if)# ip address 10.1.1.10 255.255.255.0	Configures the IP address and IP subnet.
Step 5	**no ip split-horizon** **Example:** Switch(config-if)# no ip split-horizon	Disables split horizon on the interface.
Step 6	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

	Command or Action	**Purpose**
Step 7	**show ip interface** *interface-id* **Example:** `Switch# show ip interface gigabitethernet 1/0/1`	Verifies your entries.
Step 8	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuration Example for Summary Addresses and Split Horizon

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. In the example, if the interface is still in Layer 2 mode (the default), you must enter a **no switchport** interface configuration command before entering the **ip address** interface configuration command.

> **Note** If split horizon is enabled, neither autosummary nor interface summary addresses (those configured with the **ip summary-address rip** router configuration command) are advertised.

```
Switch(config)# router rip
Switch(config-router)# interface gigabitethernet1/0/2
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
```

Information About OSPF

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. The Cisco implementation supports RFC 1253, OSPF management information base (MIB).

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 1587are supported.

✎

Note OSPF-Non Stop Routing is not supported in this release.

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.

OSPF for Routed Access

✎

Note OSPF is supported in IP Lite. OSPF for Routed Access supports only one OSPFv2 and one OSPFv3 instance with a combined total of 1000 dynamically learned routes. The IP Lite image provides OSPF for routed access. However, these restrictions are not enforced in this release.

With the typical topology (hub and spoke) in a campus environment, where the wiring closets (spokes) are connected to the distribution switch (hub) that forwards all nonlocal traffic to the distribution layer, the wiring closet switch need not hold a complete routing table. A best practice design, where the distribution switch sends a default route to the wiring closet switch to reach interarea and external routes (OSPF stub or totally stub area configuration) should be used when OSPF for Routed Access is used in the wiring closet.

For more details, see the "High Availability Campus Network Design—Routed Access Layer using EIGRP or OSPF" document.

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.

Other OSPF Parameters

You can optionally configure other OSPF parameters in router configuration mode.

- Route summarization: When redistributing routes from other protocols. 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.

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.

Loopback Interfaces

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.

How to Configure OSPF

Default OSPF Configuration

Table 68: Default OSPF Configuration

Feature	Default Setting
Interface parameters	Cost: 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 Mb/s.
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.

Feature	Default Setting
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.
Router ID	No OSPF routing process defined.
Summary address	Disabled.
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.

Configuring Basic OSPF Parameters

To enable OSPF, create an OSPF routing process, specify the range of IP addresses to associate with the routing process, and assign area IDs to be associated with that range.

Procedure

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**router ospf** *process-id* **Example:** Switch(config)# **router ospf 15**	Enables 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. **Note** OSPF for Routed Access supports only one OSPFv2 and one OSPFv3 instance with a maximum number of 1000 dynamically learned routes.
Step 3	**network** *address wildcard-mask* **area** *area-id* **Example:** Switch(config-router)# **network 10.1.1.1 255.240.0.0 area 20**	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.
Step 4	**end** **Example:** Switch(config-router)#**end**	Returns to privileged EXEC mode.
Step 5	**show ip protocols** **Example:** Switch# **show ip protocols**	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

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. If you modify these parameters, be sure all routers in the network have compatible values.

Note   The **ip ospf** interface configuration commands are all optional.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** Switch(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 3	**ip ospf cost** *cost* **Example:** Switch(config-if)# ip ospf cost 8	(Optional) Explicitly specifies the cost of sending a packet on the interface.
Step 4	**ip ospf retransmit-interval** *seconds* **Example:** Switch(config-if)# ip ospf transmit-interval 10	(Optional) Specifies the number of seconds between link state advertisement transmissions. The range is 1 to 65535 seconds. The default is 5 seconds.
Step 5	**ip ospf transmit-delay** *seconds* **Example:** Switch(config-if)# ip ospf transmit-delay 2	(Optional) Sets 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.
Step 6	**ip ospf priority** *number* **Example:** Switch(config-if)# ip ospf priority 5	(Optional) Sets priority to help find the OSPF designated router for a network. The range is from 0 to 255. The default is 1.
Step 7	**ip ospf hello-interval** *seconds* **Example:** Switch(config-if)# ip ospf hello-interval 12	(Optional) Sets 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.
Step 8	**ip ospf dead-interval** *seconds* **Example:** Switch(config-if)# ip ospf dead-interval 8	(Optional) Sets 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.
Step 9	**ip ospf authentication-key** *key* **Example:**	(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

	Command or Action	**Purpose**
	`Switch(config-if)# ip ospf authentication-key password`	neighboring routers on the same network must have the same password to exchange OSPF information.
Step 10	**ip ospf message-digest-key** *keyid* **md5** *key* **Example:** `Switch(config-if)# ip ospf message digest-key 16 md5 your1pass`	(Optional) Enables MDS authentication. • *keyid*—An identifier from 1 to 255. • *key*—An alphanumeric password of up to 16 bytes.
Step 11	**ip ospf database-filter all out** **Example:** `Switch(config-if)# 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.
Step 12	**end** **Example:** `Switch(config)# ` **end**	Returns to privileged EXEC mode.
Step 13	**show ip ospf interface** [*interface-name*] **Example:** `Switch# show ip ospf interface`	Displays OSPF-related interface information.

Step 14	**show ip ospf neighbor detail** **Example:** `Switch# show ip ospf neighbor detail`	Displays 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.

Step 15	**copy running-config startup-config** **Example:** `Switch# ` **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring OSPF Area Parameters

Before you begin

✎

Note   The OSPF **area** router configuration commands are all optional.

Procedure

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**router ospf** *process-id* **Example:** Switch(config)# router ospf 109	Enables OSPF routing, and enter router configuration mode.
Step 3	**area** *area-id* **authentication** **Example:** Switch(config-router)# area 1 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.
Step 4	**area** *area-id* **authentication message-digest** **Example:** Switch(config-router)# area 1 authentication message-digest	(Optional) Enables MD5 authentication on the area.
Step 5	**area** *area-id* **stub** [**no-summary**] **Example:** Switch(config-router)# area 1 stub	(Optional) Define an area as a stub area. The **no-summary** keyword prevents an ABR from sending summary link advertisements into the stub area.
Step 6	**area** *area-id* **nssa** [**no-redistribution**] [**default-information-originate**] [**no-summary**] **Example:** Switch(config-router)# area 1 nssa default-information-originate	(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.

	Command or Action	Purpose
		• **no-redistribution**—Select to not send summary LSAs into the NSSA.
Step 7	**area** *area-id* **range** *address mask* **Example:** Switch(config-router)# area 1 range 255.240.0.0	(Optional) Specifies an address range for which a single route is advertised. Use this command only with area border routers.
Step 8	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 9	**show ip ospf** [*process-id*] **Example:** Switch# show ip ospf	Displays information about the OSPF routing process in general or for a specific process ID to verify configuration.
Step 10	**show ip ospf** [*process-id* [*area-id*]] **database** **Example:** Switch# show ip osfp database	Displays lists of information related to the OSPF database for a specific router.
Step 11	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Other OSPF Parameters

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**router ospf** *process-id* **Example:** Switch(config)# router ospf 10	Enables OSPF routing, and enter router configuration mode.

	Command or Action	**Purpose**
Step 3	**summary-address** *address mask* **Example:** Switch(config)# summary-address 10.1.1.1 255.255.255.0	(Optional) Specifies an address and IP subnet mask for redistributed routes so that only one summary route is advertised.
Step 4	**area** *area-id* **virtual-link** *router-id* [**hello-interval** *seconds*] [**retransmit-interval** *seconds*] [**trans**] [[**authentication-key** *key*] | **message-digest-key** *keyid* **md5** *key*]] **Example:** Switch(config)# area 2 virtual-link 192.168.255.1 hello-interval 5	(Optional) Establishes a virtual link and set its parameters.
Step 5	**default-information originate** [**always**] [**metric** *metric-value*] [**metric-type** *type-value*] [**route-map** *map-name*] **Example:** Switch(config)# default-information originate metric 100 metric-type 1	(Optional) Forces the ASBR to generate a default route into the OSPF routing domain. Parameters are all optional.
Step 6	**ip ospf name-lookup** **Example:** Switch(config)# ip ospf name-lookup	(Optional) Configures DNS name lookup. The default is disabled.
Step 7	**ip auto-cost reference-bandwidth** *ref-bw* **Example:** Switch(config)# ip auto-cost reference-bandwidth 5	(Optional) Specifies an address range for which a single route will be advertised. Use this command only with area border routers.
Step 8	**distance ospf** {[**inter-area** *dist1*] [**inter-area** *dist2*] [**external** *dist3*]} **Example:** Switch(config)# distance ospf inter-area 150	(Optional) Changes the OSPF distance values. The default distance for each type of route is 110. The range is 1 to 255.
Step 9	**passive-interface** *type number* **Example:** Switch(config)# passive-interface gigabitethernet 1/0/6	(Optional) Suppresses the sending of hello packets through the specified interface.
Step 10	**timers throttle spf** *spf-delay spf-holdtime spf-wait* **Example:**	(Optional) Configures route calculation timers.

	Command or Action	Purpose
	`Switch(config)# timers throttle spf 200 100 100`	• *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.
Step 11	**ospf log-adj-changes** **Example:** `Switch(config)# ospf log-adj-changes`	(Optional) Sends syslog message when a neighbor state changes.
Step 12	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 13	**show ip ospf** [*process-id* [*area-id*]] **database** **Example:** `Switch# show ip ospf database`	Displays lists of information related to the OSPF database for a specific router.
Step 14	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Changing LSA Group Pacing

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**router ospf** *process-id* **Example:**	Enables OSPF routing, and enter router configuration mode.

	Command or Action	**Purpose**
	Switch(config)# router ospf 25	
Step 3	**timers lsa-group-pacing** *seconds* **Example:** Switch(config-router)# timers lsa-group-pacing 15	Changes the group pacing of LSAs.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring a Loopback Interface

Procedure

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface loopback 0** **Example:** Switch(config)# interface loopback 0	Creates a loopback interface, and enter interface configuration mode.
Step 3	**ip address** address mask **Example:** Switch(config-if)# ip address 10.1.1.5 255.255.240.0	Assign an IP address to this interface.

	Command or Action	**Purpose**
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show ip interface** **Example:** Switch# show ip interface	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Monitoring OSPF

You can display specific statistics such as the contents of IP routing tables, caches, and databases.

Table 69: Show IP OSPF Statistics Commands

show ip ospf [*process-id*]	Displays 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**]	Displays lists of information related to the OSPF database.

show ip ospf border-routes	Displays the internal OSPF routing ABR and ASBR table entries.
show ip ospf interface [*interface-name*]	Displays OSPF-related interface information.
show ip ospf neighbor [*interface-name*] [*neighbor-id*] detail	Displays OSPF interface neighbor information.
show ip ospf virtual-links	Displays OSPF-related virtual links information.

Configuration Examples for OSPF

Example: Configuring Basic OSPF Parameters

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
```

Information About EIGRP

Enhanced IGRP (EIGRP) is a Cisco proprietary enhanced version of the 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. When a RIP route is used as the next hop to the destination, the transport control field is incremented as usual.

EIGRP Stub Routing

The EIGRP stub routing feature reduces resource utilization by moving routed traffic closer to the end user.

Note   The IP Lite feature set contains EIGRP stub routing capability, which 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.

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.

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 the figure given below, 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 69: EIGRP Stub Router Configuration
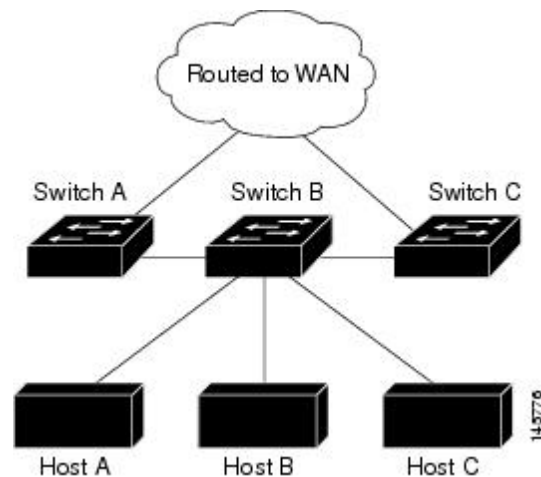
For more information about EIGRP stub routing, see "Configuring EIGRP Stub Routing" section of the *Cisco IOS IP Configuration Guide, Volume 2 of 3: Routing Protocols*.

Configuring Unicast Reverse Path Forwarding

The unicast reverse path forwarding (unicast RPF) feature helps to mitigate problems that are caused by the introduction of malformed or forged (spoofed) IP source addresses into a network by discarding IP packets that lack a verifiable IP source address. For example, a number of common types of denial-of-service (DoS) attacks, including Smurf and Tribal Flood Network (TFN), can take advantage of forged or rapidly changing source IP addresses to allow attackers to thwart efforts to locate or filter the attacks. For Internet service providers (ISPs) that provide public access, Unicast RPF deflects such attacks by forwarding only packets that have source addresses that are valid and consistent with the IP routing table. This action protects the network of the ISP, its customer, and the rest of the Internet.

Note
- Unicast RPF is supported in IP Lite.

For detailed IP unicast RPF configuration information, see the *Other Security Features* chapter in the *Cisco IOS Security Configuration Guide*.

Protocol-Independent Features

This section describes IP routing protocol-independent features that are available on switches running the IP Lite feature set . For a complete description of the IP routing protocol-independent commands in this chapter, see the "IP Routing Protocol-Independent Commands" chapter of the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols*.

Distributed Cisco Express Forwarding

Information About 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 a switch stack, the hardware uses distributed CEF (dCEF) in the stack. 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 and dCEF use the Forwarding Information Base (FIB) lookup table to perform destination-based switching of IP packets.

The two main components in CEF and dCEF 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 or switch stack uses Application Specific Integrated Circuits (ASICs) to achieve Gigabit-speed line rate IP traffic, CEF or dCEF forwarding applies only to the software-forwarding path, that is, traffic that is forwarded by the CPU.

How to Configure Cisco Express Forwarding

CEF or distributed CEF is enabled globally by default. If for some reason it is disabled, you can re-enable it by using the **ip cef** or **ip cef distributed** global configuration command.

The default configuration is CEF or dCEF 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 or dCEF on interfaces except for debugging purposes.

To enable CEF or dCEF globally and on an interface for software-forwarded traffic if it has been disabled:

SUMMARY STEPS

1. **configure terminal**
2. **ip cef**
3. **ip cef distributed**
4. **interface** *interface-id*
5. **ip route-cache cef**
6. **end**
7. **show ip cef**
8. **show cef linecard** [**detail**]
9. **show cef linecard** [*slot-number*] [**detail**]
10. **show cef interface** [*interface-id*]
11. **show adjacency**
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**ip cef** **Example:** Switch(config)# ip cef	Enables CEF operation on a non-stacking switch. Go to Step 4.
Step 3	**ip cef distributed** **Example:** Switch(config)# ip cef distributed	Enables CEF operation on a active switch.
Step 4	**interface** *interface-id* **Example:** Switch(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the Layer 3 interface to configure.

	Command or Action	Purpose
Step 5	**ip route-cache cef** **Example:** Switch(config-if)# ip route-cache cef	Enables CEF on the interface for software-forwarded traffic.
Step 6	**end** **Example:** Switch(config-if)# end	Returns to privileged EXEC mode.
Step 7	**show ip cef** **Example:** Switch# show ip cef	Displays the CEF status on all interfaces.
Step 8	**show cef linecard** [**detail**] **Example:** Switch# show cef linecard detail	(Optional) Displays CEF-related interface information on a non-stacking switch.
Step 9	**show cef linecard** [*slot-number*] [**detail**] **Example:** Switch# show cef linecard 5 detail	(Optional) Displays CEF-related interface information on a switch by stack member for all switches in the stack or for the specified switch. (Optional) For *slot-number*, enter the stack member switch number.
Step 10	**show cef interface** [*interface-id*] **Example:** Switch# show cef interface gigabitethernet 1/0/1	Displays detailed CEF information for all interfaces or the specified interface.
Step 11	**show adjacency** **Example:** Switch# show adjacency	Displays CEF adjacency table information.
Step 12	**copy running-config startup-config** **Example:** Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Number of Equal-Cost Routing Paths

Information About 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. Equal-cost routes are supported across switches in a stack.

Even though 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. Although the switch software allows a maximum of 32 equal-cost routes, the switch hardware will never use more than 16 paths per route.

How to Configure Equal-Cost Routing Paths

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**router** {**rip** | **ospf** | **eigrp**} **Example:** `Switch(config)# router eigrp 10`	Enters router configuration mode.
Step 3	**maximum-paths** *maximum* **Example:** `Switch(config-router)# maximum-paths 2`	Sets the maximum number of parallel paths for the protocol routing table. The range is from 1 to 16; the default is 4 for most IP routing protocols.
Step 4	**end** **Example:** `Switch(config-router)# end`	Returns to privileged EXEC mode.
Step 5	**show ip protocols** **Example:** `Switch# show ip protocols`	Verifies the setting in the *Maximum path* field.
Step 6	**copy running-config startup-config** **Example:**	(Optional) Saves your entries in the configuration file.

Command or Action	Purpose
Switch# copy running-config startup-config	

Static Unicast Routes

Information About 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 41-16. 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 70: Dynamic Routing Protocol Default Administrative Distances

Route Source	Default Distance
Connected interface	0
Static route	1
Enhanced IRGP summary route	5
Internal Enhanced IGRP	90
IGRP	100
OSPF	110
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.

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.

Follow these steps to configure a static route:

Procedure

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**ip route prefix mask** {*address* | *interface*} [*distance*] **Example:** Switch(config)# ip route prefix mask gigabitethernet 1/0/4	Establish a static route.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show ip route** **Example:** Switch# show ip route	Displays the current state of the routing table to verify the configuration.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

What to do next

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.

Default Routes and Networks

Information About 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.

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.

How to Configure Default Routes and Networks

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** Example: `Switch# configure terminal`	Enters global configuration mode.
Step 2	**ip default-network** *network number* Example: `Switch(config)# ip default-network 1`	Specifies a default network.
Step 3	**end** Example: `Switch(config)# end`	Returns to privileged EXEC mode.
Step 4	**show ip route** Example:	Displays the selected default route in the gateway of last resort display.

	Command or Action	Purpose
	`Switch# show ip route`	
Step 5	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Route Maps to Redistribute Routing Information

Information About Route Maps

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.

How to Configure a Route Map

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.

Note The keywords are the same as defined in the procedure to control the route distribution.

Procedure

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# configure terminal	Enters global configuration mode.
Step 2	**route-map** map-tag [**permit** | **deny**] [sequence number] **Example:** Switch(config)# route-map rip-to-ospf permit 4	Defines 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.
Step 3	**match metric** metric-value **Example:** Switch(config-route-map)# match metric 2000	Matches the specified route metric. The *metric-value* can be an EIGRP metric with a specified value from 0 to 4294967295.
Step 4	**match ip next-hop** {*access-list-number* | *access-list-name*} [...*access-list-number* | ...*access-list-name*] **Example:** Switch(config-route-map)# match ip next-hop 8 45	Matches a next-hop router address passed by one of the access lists specified (numbered from 1 to 199).
Step 5	**match tag** tag value [...tag-value] **Example:** Switch(config-route-map)# match tag 3500	Matches the specified tag value in a list of one or more route tag values. Each can be an integer from 0 to 4294967295.
Step 6	**match interface** type number [...type number] **Example:** Switch(config-route-map)# match interface gigabitethernet 1/0/1	Matches the specified next hop route out one of the specified interfaces.
Step 7	**match ip route-source** {*access-list-number* | *access-list-name*} [...*access-list-number* | ...*access-list-name*] **Example:**	Matches the address specified by the specified advertised access lists.

	Command or Action	Purpose
	`Switch(config-route-map)# match ip route-source 10 30`	
Step 8	**match route-type** {**local** | **internal** | **external** [**type-1** | **type-2**]} **Example:** `Switch(config-route-map)# match route-type local`	Matches the specified **route-type**: • **internal**—OSPF intra-area and interarea routes or EIGRP internal routes. • **external**—OSPF external routes (Type 1 or Type 2) or EIGRP external routes.
Step 9	**set metric** metric value **Example:** `Switch(config-route-map)# set metric 100`	Sets the metric value to give the redistributed routes (for EIGRP only). The *metric value* is an integer from -294967295 to 294967295.
Step 10	**set metric** bandwidth delay reliability loading mtu **Example:** `Switch(config-route-map)# set metric 10000 10 255 1 1500`	Sets 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.
Step 11	**set metric-type** {**type-1** | **type-2**} **Example:** `Switch(config-route-map)# set metric-type type-2`	Sets the OSPF external metric type for redistributed routes.
Step 12	**end** **Example:** `Switch(config-route-map)# end`	Returns to privileged EXEC mode.
Step 13	**show route-map** **Example:**	Displays all route maps configured or only the one specified to verify configuration.

	Command or Action	Purpose
	`Switch# show route-map`	
Step 14	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

How to Control Route Distribution

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.

✎

Note The keywords are the same as defined in the procedure to configure the route map for redistritbution.

The metrics of one routing protocol do not necessarily translate into the metrics of another. For example, the RIP metric is a hop count, and the IGRP metric is a combination of five qualities. 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.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**router** { **rip**	**ospf**	**eigrp**} **Example:** `Switch(config)# router eigrp 10`	Enters router configuration mode.
Step 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**] **Example:** `Switch(config-router)# redistribute eigrp 1`	Redistributes 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.

	Command or Action	**Purpose**
Step 4	**default-metric** *number* **Example:** Switch(config-router)# default-metric 1024	Cause the current routing protocol to use the same metric value for all redistributed routes (RIP and OSPF).
Step 5	**default-metric** bandwidth delay reliability loading mtu **Example:** Switch(config-router)# default-metric 1000 100 250 100 1500	Cause the EIGRP routing protocol to use the same metric value for all non-EIGRP redistributed routes.
Step 6	**end** **Example:** Switch(config-router)# end	Returns to privileged EXEC mode.
Step 7	**show route-map** **Example:** Switch# show route-map	Displays all route maps configured or only the one specified to verify configuration.
Step 8	**copy running-config startup-config** **Example:** Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Policy-Based Routing

Information About 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.

- Route map statement marked as permit is processed as follows:

  - A match command can match on length or multiple ACLs. A route map statement can contain multiple match commands. Logical or algorithm function is performed across all the match commands to reach a permit or deny decision.

    For example:

    match length A B

    match ip address acl1 acl2

    match ip address acl3

  A packet is permitted if it is permitted by match length A B or acl1 or acl2 or acl3

  - If the decision reached is permit, then the action specified by the set command is applied on the packet .
  - If the decision reached is deny, then the PBR action (specified in the set command) is not applied. Instead the processing logic moves forward to look at the next route-map statement in the sequence (the statement with the next higher sequence number). If no next statement exists, PBR processing terminates, and the packet is routed using the default IP routing table.

- For PBR, route-map statements marked as deny are not supported.

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 *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols*.

Policy-Based Routing Using Object Tracking

You can configure policy-based routing (PBR) to use object tracking to verify the most viable next-hop IP address to which to forward packets, using an Internet Control Message Protocol (ICMP) ping as the verification method.

PBR with Object Tracking is most suitable for devices that have multiple Ethernet connections as the next hop. Normally, Ethernet interfaces connect to DSL modems or cable modems, and do not detect a failure upstream, in the ISP broadband network. The Ethernet interface remains up, and any form of static routing points to that interface. PBR with object tracking allows you to back up two Ethernet interfaces, determine the interface that is available by sending ICMP pings to verify reachability, and then route traffic to that interface.

To verify the next-hop IP address for the device, PBR informs the object-tracking process that it is interested in tracking a certain object. The tracking process, in turn, informs PBR when the state of the object changes.

Note   VRF is not supported with PBR using Object Tracking.

How to Configure PBR

- To use PBR, you must have the IP Lite feature set enabled on the switch or active stack.

- 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 supports PBR based on match length.

- 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 128 IP policy route maps on the switch or switch stack.

- You can define a maximum of 512 access control entries (ACEs) for PBR on the switch or switch stack.

- 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 flappping.

- VRF and PBR are mutually exclusive on a switch interface. You cannot enable VRF when PBR is enabled on an interface. The reverse is also true, you cannot enable PBR when VRF is enabled on an interface.

- The number of hardware entries used by PBR depends on the route map itself, the ACLs used, and the order of the ACLs and route-map entries.

- PBR based on TOS, DSCP and IP Precedence are not supported.

- Set interface, set default next-hop and set default interface are not supported.

- **ip next-hop recursive** and **ip next-hop verify availability** features are not available and the next-hop should be directly connected.

- Policy-maps with no set actions are supported. Matching packets are routed normally.

- Policy-maps with no match clauses are supported. Set actions are applied to all packets.

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. 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.

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.

Procedure

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.

	Command or Action	**Purpose**
Step 2	**route-map** *map-tag* [**permit**] [*sequence number*] **Example:** Switch(config)# route-map pbr-map permit	Defines route maps that are used to control where packets are output, and enters 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-map statements with the same map tag define a single route map. • (Optional) **permit** − If **permit** is specified and the match criteria are met for this route map, the route is policy routed as defined by the set actions. • (Optional) *sequence number* − The sequence number shows the position of the route-map statement in the given route map.
Step 3	**match ip address** {*access-list-number* | *access-list-name*} [*access-list-number* |...*access-list-name*] **Example:** Switch(config-route-map)# match ip address 110 140	Matches the source and destination IP addresses that are permitted by one or more standard or extended access lists. ACLs can match on more than one source and destination IP address. If you do not specify a **match** command, the route map is applicable to all packets.
Step 4	**match length** *min max* **Example:** Switch(config-route-map)# match length 64 1500	Matches the length of the packet.
Step 5	**set ip next-hop** *ip-address* [...*ip-address*] **Example:** Switch(config-route-map)# set ip next-hop 10.1.6.2	Specifies the action to be taken on the packets that match the criteria. Sets next hop to which to route the packet (the next hop must be adjacent).
Step 6	**exit** **Example:** Switch(config-route-map)# exit	Returns to global configuration mode.
Step 7	**interface** *interface-id* **Example:** Switch(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the interface to be configured.
Step 8	**ip policy route-map** *map-tag* **Example:** Switch(config-if)# ip policy route-map pbr-map	Enables 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 the order of sequence number until the first match. If there is no match, packets are routed as usual.

	Command or Action	**Purpose**
Step 9	**ip route-cache policy** **Example:** Switch(config-if)# ip route-cache policy	(Optional) Enables fast-switching PBR. You must enable PBR before enabling fast-switching PBR.
Step 10	**exit** **Example:** Switch(config-if)# exit	Returns to global configuration mode.
Step 11	**ip local policy route-map** *map-tag* **Example:** Switch(config)# ip local policy route-map local-pbr	(Optional) Enables 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.
Step 12	**end** **Example:** Switch(config)# end	Returns to privileged EXEC mode.
Step 13	**show route-map** [*map-name*] **Example:** Switch# show route-map	(Optional) Displays all the route maps configured or only the one specified to verify configuration.
Step 14	**show ip policy** **Example:** Switch# show ip policy	(Optional) Displays policy route maps attached to the interface.
Step 15	**show ip local policy** **Example:** Switch# show ip local policy	(Optional) Displays whether or not local policy routing is enabled and, if so, the route map being used.

Verifying Next-Hop IP Using Object Tracking

To verify the next-hop IP address using PBR with object tracking, perform the following steps:

Procedure

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# configure terminal	Enters global configuration mode.
Step 2	**track** *object-number* **ip sla** *entry-number* **Example:** Switch(config)# track 100 ip sla 100	Tracks the state of an IP SLA object.

	Command or Action	Purpose
Step 3	**ip sla** *operation-number* **Example:** Switch(config)# ip sla 100	Starts a Cisco IOS IP Service Level Agreement (SLA) operation configuration, and enters IP SLA configuration mode.
Step 4	**icmp-echo** *ip-address* **source-ip** *ip-address* **Example:** Switch(config-ip-sla)# icmp-echo 172.19.255.253 source-ip 172.19.255.47	Configures an IP SLA Internet Control Message Protocol(ICMP) echo probe operation, and enters Echo configuration mode.
Step 5	**frequency** *seconds* **Example:** Switch(config-ip-sla-echo)# frequency 2	(Optional) Sets the rate, in seconds, at which a specified IP SLA operation is repeated.
Step 6	**threshold** *milliseconds* **Example:** Switch(config-ip-sla-echo)# threshold 1000	(Optional) Sets the length of time, in ms, required for a rising threshold event to be declared.
Step 7	**timeout** *milliseconds* **Example:** Switch(config-ip-sla-echo)# timeout 1500	(Optional) Sets the maximum time, in ms, required for the IP SLA operation to be completed.
Step 8	**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*] **Example:** Switch(config)# ip sla schedule 100 life forever start-time now	Configures the scheduling parameters for a single Cisco IOS IP SLA operation.
Step 9	**route-map** *map-tag* [**permit** | **deny**] *[sequence-number]* **Example:** Switch(config)# route-map alpha permit 10	Specifies a route map and enters route-map configuration mode.
Step 10	**match ip address** *[access-list-name]* **Example:** Switch(config-route-map)# match ip address exlist	Distributes routes that have a destination IPv4 network number address that is permitted by a standard access list.
Step 11	**set ip next-hop verify-availability** *[next-hop-address sequence* **track** *object]* **Example:** Switch(config-route-map)# set ip next-hop verify-availability 95.1.1.2.1 track 100	Configures the route map to verify the reachability of the tracked object.

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.

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.

Procedure

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# configure terminal	Enters global configuration mode.
Step 2	**router** { **rip** | **ospf** | **eigrp**} **Example:** Switch(config)# router ospf	Enters router configuration mode.
Step 3	**passive-interface** *interface-id* **Example:** Switch(config-router)# passive-interface gigabitethernet 1/0/1	Suppresses sending routing updates through the specified Layer 3 interface.
Step 4	**passive-interface default** **Example:** Switch(config-router)# passive-interface default	(Optional) Sets all interfaces as passive by default.
Step 5	**no passive-interface** *interface type* **Example:**	(Optional) Activates only those interfaces that need to have adjacencies sent.

	Command or Action	Purpose
	`Switch(config-router)# no passive-interface` `gigabitethernet1/0/3 gigabitethernet 1/0/5`	
Step 6	**network** *network-address* **Example:** `Switch(config-router)# network 10.1.1.1`	(Optional) Specifies the list of networks for the routing process. The *network-address* is an IP address.
Step 7	**end** **Example:** `Switch(config-router)# end`	Returns to privileged EXEC mode.
Step 8	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

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.)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**router** { **rip**	**eigrp**} **Example:** `Switch(config)# router eigrp 10`	Enters router configuration mode.

Step 3	**distribute-list** {*access-list-number*	*access-list-name*} **out** [*interface-name*	*routing process*	*autonomous-system-number*]
				Example:
				`Switch(config-router)# distribute-list 120 out`
				`gigabitethernet 1/0/7`
					Permits or denies routes from being advertised in routing updates, depending upon the action listed in the access list.

	Command or Action	**Purpose**
Step 4	**distribute-list** {*access-list-number* | *access-list-name*} **in** [*type-number*] **Example:** Switch(config-router)# distribute-list 125 in	Suppresses processing in routes listed in updates.
Step 5	**end** **Example:** Switch(config-router)# end	Returns to privileged EXEC mode.
Step 6	**copy running-config startup-config** **Example:** Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

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.

Procedure

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# configure terminal	Enters global configuration mode.
Step 2	**router** { **rip** | **ospf** | **eigrp**} **Example:** Switch(config)# router eigrp 10	Enters router configuration mode.
Step 3	**distance** *weight* {*ip-address* {*ip-address mask*}} [*ip access list*] **Example:** Switch(config-router)# distance 50 10.1.5.1	Defines 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.

	Command or Action	Purpose
		(Optional) *ip access list*—An IP standard or extended access list to be applied to incoming routing updates.
Step 4	**end** **Example:** Switch(config-router)# end	Returns to privileged EXEC mode.
Step 5	**show ip protocols** **Example:** Switch# show ip protocols	Displays the default administrative distance for a specified routing process.
Step 6	**copy running-config startup-config** **Example:** Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

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.

Prerequisites

Before you manage authentication keys, you must enable authentication. See the appropriate protocol section to see how to enable authentication for that protocol. 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.

How to Configure Authentication Keys

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.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# configure terminal	Enters global configuration mode.

	Command or Action	**Purpose**
Step 2	**key chain** *name-of-chain* **Example:** Switch(config)# key chain key10	Identifies a key chain, and enter key chain configuration mode.
Step 3	**key** *number* **Example:** Switch(config-keychain)# key 2000	Identifies the key number. The range is 0 to 2147483647.
Step 4	**key-string** *text* **Example:** Switch(config-keychain)# key-string Room 20, 10th floor	Identifies the key string. The string can contain from 1 to 80 uppercase and lowercase alphanumeric characters, but the first character cannot be a number.
Step 5	**accept-lifetime** *start-time* {**infinite** | *end-time* | **duration** *seconds*} **Example:** Switch(config-keychain)# accept-lifetime 12:30:00 Jan 25 1009 infinite	(Optional) Specifies 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**.
Step 6	**send-lifetime** *start-time* {**infinite** | *end-time* | **duration** *seconds*} **Example:** Switch(config-keychain)# accept-lifetime 23:30:00 Jan 25 1019 infinite	(Optional) Specifies 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**.
Step 7	**end** **Example:** Switch(config-keychain)# end	Returns to privileged EXEC mode.
Step 8	**show key chain** **Example:** Switch# show key chain	Displays authentication key information.
Step 9	**copy running-config startup-config** **Example:** Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Monitoring and Maintaining the IP Network

You can remove all contents of a particular cache, table, or database. You can also display specific statistics.

Table 71: Commands to Clear IP Routes or Display Route Status

clear ip route {*network* [*mask* | *****]}	Clears one or more routes from the IP routing table.
show ip protocols	Displays the parameters and state of the active routing protocol process.
show ip route [*address* [*mask*] [**longer-prefixes**]] | [*protocol* [*process-id*]]	Displays the current state of the routing table.
show ip route summary	Displays the current state of the routing table in summary form.
show ip route supernets-only	Displays supernets.
show ip cache	Displays the routing table used to switch IP traffic.
show route-map [*map-name*]	Displays all route maps configured or only the one specified.

PART VIII

Security

Security Features Overview

Security Features Overview

The switch supports a LAN base image or a LAN lite image with a reduced feature set, depending on switch hardware. The security features are as follows:

- IPv6 First Hop Security—A suite of security features to be applied at the first hop switch to protect against vulnerabilities inherent in IPv6 networks. These include, Binding Integrity Guard (Binding Table), Router Advertisement Guard (RA Guard), DHCP Guard, IPv6 Neighbor Discovery Inspection (ND Guard).

- Web Authentication—Allows a supplicant (client) that does not support IEEE 802.1x functionality to be authenticated using a web browser.

- Local Web Authentication Banner—A custom banner or an image file displayed at a web authentication login screen.

- IEEE 802.1x Authentication with ACLs and the RADIUS Filter-Id Attribute

- Password-protected access (read-only and read-write access) to management interfaces (device manager, Network Assistant, and the CLI) for protection against unauthorized configuration changes

- Multilevel security for a choice of security level, notification, and resulting actions

- Static MAC addressing for ensuring security

- Protected port option for restricting the forwarding of traffic to designated ports on the same switch

- Port security option for limiting and identifying MAC addresses of the stations allowed to access the port

- VLAN aware port security option to shut down the VLAN on the port when a violation occurs, instead of shutting down the entire port.

- Port security aging to set the aging time for secure addresses on a port.

- Protocol storm protection to control the rate of incoming protocol traffic to a switch by dropping packets that exceed a specified ingress rate.

- BPDU guard for shutting down a Port Fast-configured port when an invalid configuration occurs.

- Standard and extended IP access control lists (ACLs) for defining inbound security policies on Layer 2 interfaces (port ACLs).

- Extended MAC access control lists for defining security policies in the inbound direction on Layer 2 interfaces.

- Source and destination MAC-based ACLs for filtering non-IP traffic.

- DHCP snooping to filter untrusted DHCP messages between untrusted hosts and DHCP servers.

- IP source guard to restrict traffic on nonrouted interfaces by filtering traffic based on the DHCP snooping database and IP source bindings.

- Dynamic ARP inspection to prevent malicious attacks on the switch by not relaying invalid ARP requests and responses to other ports in the same VLAN.

- IEEE 802.1x port-based authentication to prevent unauthorized devices (clients) from gaining access to the network. These 802.1x features are supported:

  - Support for single-host, multi-host, multi-auth, and multi-domain-auth modes.

  - Multidomain authentication (MDA) to allow both a data device and a voice device, such as an IP phone (Cisco or non-Cisco), to independently authenticate on the same IEEE 802.1x-enabled switch port.

  - Dynamic voice virtual LAN (VLAN) for MDA to allow a dynamic voice VLAN on an MDA-enabled port.

  - VLAN assignment for restricting 802.1x-authenticated users to a specified VLAN.

  - Support for VLAN assignment on a port configured for multi-auth mode. The RADIUS server assigns a VLAN to the first host to authenticate on the port, and subsequent hosts use the same VLAN. Voice VLAN assignment is supported for one IP phone.

  - Port security for controlling access to 802.1x ports.

  - Voice VLAN to permit a Cisco IP Phone to access the voice VLAN regardless of the authorized or unauthorized state of the port.

  - IP phone detection enhancement to detect and recognize a Cisco IP phone.

  - Guest VLAN to provide limited services to non-802.1x-compliant users.

  - Restricted VLAN to provide limited services to users who are 802.1x compliant, but do not have the credentials to authenticate via the standard 802.1x processes.

  - 802.1x accounting to track network usage.

  - 802.1x with wake-on-LAN to allow dormant PCs to be powered on based on the receipt of a specific Ethernet frame.

  - 802.1x readiness check to determine the readiness of connected end hosts before configuring IEEE 802.1x on the switch.

  - Voice aware 802.1x security to apply traffic violation actions only on the VLAN on which a security violation occurs.

  - MAC authentication bypass (MAB) to authorize clients based on the client MAC address.

- Network Admission Control (NAC) Layer 2 802.1x validation of the antivirus condition or posture of endpoint systems or clients before granting the devices network access.

✎

Note   NAC is not supported on LanLite images.

- Network Edge Access Topology (NEAT) with 802.1X switch supplicant, host authorization with CISP, and auto enablement to authenticate a switch outside a wiring closet as a supplicant to another switch.

✎

Note   NEAT is not supported on LanLite images.

- IEEE 802.1x with open access to allow a host to access the network before being authenticated.

✎

Note   This feature is not supported on LanLite images.

- IEEE 802.1x authentication with downloadable ACLs and redirect URLs to allow per-user ACL downloads from a Cisco Secure ACS server to an authenticated switch.

- Support for dynamic creation or attachment of an auth-default ACL on a port that has no configured static ACLs.

✎

Note   This feature is not supported on LanLite images.

- Flexible-authentication sequencing to configure the order of the authentication methods that a port tries when authenticating a new host.

- Multiple-user authentication to allow more than one host to authenticate on an 802.1x-enabled port.

- TACACS+, a proprietary feature for managing network security through a TACACS server for both IPv4 and IPv6.

- RADIUS for verifying the identity of, granting access to, and tracking the actions of remote users through authentication, authorization, and accounting (AAA) services for both IPv4 and IPv6.

- Enhancements to RADIUS, TACACS+, and SSH to function over IPv6.

- Secure Socket Layer (SSL) Version 3.0 support for the HTTP 1.1 server authentication, encryption, and message integrity and HTTP client authentication to allow secure HTTP communications (requires the cryptographic version of the software).

- IEEE 802.1x Authentication with ACLs and the RADIUS Filter-Id Attribute.

- Support for IP source guard on static hosts.

- RADIUS Change of Authorization (CoA) to change the attributes of a certain session after it is authenticated. When there is a change in policy for a user or user group in AAA, administrators can send

the RADIUS CoA packets from the AAA server, such as Cisco Identity Services Engine, or Cisco Secure ACS to reinitialize authentication, and apply to the new policies.

- IEEE 802.1x User Distribution to allow deployments with multiple VLANs (for a group of users) to improve scalability of the network by load balancing users across different VLANs. Authorized users are assigned to the least populated VLAN in the group, assigned by RADIUS server.

  Note   This feature is not supported on LanLite images.

- Support for critical VLAN—multi-host/multi-auth enabled ports are placed in a critical VLAN in order to permit access to critical resources if AAA server becomes unreachable.

  Note   This feature is not supported on LanLite images.

- Support for Network Edge Access Topology (NEAT) to change the port host mode and to apply a standard port configuration on the authenticator switch port.

- VLAN-ID based MAC authentication to use the combined VLAN and MAC address information for user authentication to prevent network access from unauthorized VLANs.

- MAC move to allow hosts (including the hosts connected behind an IP phone) to move across ports within the same switch without any restrictions to enable mobility. With MAC move, the switch treats the reappearance of the same MAC address on another port in the same way as a completely new MAC address.

- Support for 3DES and AES with version 3 of the Simple Network Management Protocol (SNMPv3). This release adds support for the 168-bit Triple Data Encryption Standard (3DES) and the 128-bit, 192-bit, and 256-bit Advanced Encryption Standard (AES) encryption algorithms to SNMPv3.

- Support for Cisco TrustSec SXP protocol. This feature is not supported on LanLite images.

Preventing Unauthorized Access

- Preventing Unauthorized Access, on page 679

Preventing Unauthorized 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. For more information, see the Cisco IOS Login Enhancements documentation.

Controlling Switch Access with Passwords and Privilege Levels

Restrictions for Controlling Switch Access with Passwords and Privileges

The following are the restrictions for controlling switch access with passwords and privileges:

• Disabling password recovery will not work if you have set the switch to boot up manually by using the **boot manual** global configuration command. This command produces the boot loader prompt (*switch:*) after the switch is power cycled.

Information About Passwords and Privilege Levels

Default Password and Privilege Level Configuration

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.

This table shows the default password and privilege level configuration.

Table 72: 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.

Additional Password Security

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.

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.

Password Recovery

By default, 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 then by entering a new password.

The password-recovery disable feature protects access to the switch password by disabling part of this functionality. When this feature is enabled, the end user can interrupt the boot process only by agreeing to set the system back to the default configuration. With password recovery disabled, you can still interrupt the boot process and change the password, but the configuration file (config.text) and the VLAN database file (vlan.dat) are deleted.

If you disable password recovery, we recommend that you keep a backup copy of the configuration file on a secure server in case the end user interrupts the boot process and sets the system back to default values. Do not keep a backup copy of the configuration file on the switch. If the switch is operating in VTP transparent mode, we recommend that you also keep a backup copy of the VLAN database file on a secure server. When the switch is returned to the default system configuration, you can download the saved files to the switch by using the Xmodem protocol.

To re-enable password recovery, use the **service password-recovery** global configuration command.

Terminal Line Telnet Configuration

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 when you set a Telnet password for a terminal line.

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.

Privilege Levels

Cisco switches (and other devices) use privilege levels to provide password security for different levels of switch operation. By default, the Cisco IOS software operates in two modes (privilege levels) of password security: user EXEC (Level 1) and privileged EXEC (Level 15). 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.

Privilege Levels on Lines

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.

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.

Command Privilege Levels

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.

How to Control Switch Access with Passwords and Privilege Levels

Setting or Changing a Static Enable Password

The enable password controls access to the privileged EXEC mode. Follow these steps to set or change a static enable password:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **enable password** *password*
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**enable password** *password* **Example:** Switch(config)# **enable password secret321**	Defines a new password or changes an existing password for access to privileged EXEC mode. By default, no password is defined. For *password*, specify 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: **a.** Enter **abc**. **b.** Enter **Crtl-v**.

	Command or Action	**Purpose**
		c. Enter **?123**. When the system prompts you to enter the enable password, you need not precede the question mark with the Ctrl-v; you can simply enter abc?123 at the password prompt.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Protecting Enable and Enable Secret Passwords with Encryption

Follow these steps to establish an encrypted password that users must enter to access privileged EXEC mode (the default) or any privilege level you specify:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. Use one of the following:
    - enable password [level *level*]
      {*password encryption-type encrypted-password*}
    - enable secret [level *level*]
      {*password encryption-type encrypted-password*}
4. **service password-encryption**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.

Step 3	Use one of the following:
	
	• enable password [level *level*] {*password encryption-type encrypted-password*}
	• enable secret [level *level*] {*password encryption-type encrypted-password*}
	
	Example:
	
	Switch(config)# **enable password example102**
	
	or
	
	Switch(config)# **enable secret level 1 password secret123sample**
		• Defines a new password or changes an existing password for access to privileged EXEC mode.
		
		• Defines a secret password, which is saved using a nonreversible encryption method.
		
		  • (Optional) For *level*, the range is from 0 to 15. Level 1 is normal user EXEC mode privileges. The default level is 15 (privileged EXEC mode privileges).
		
		  • For *password*, specify 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) For *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 can not re-enter privileged EXEC mode. You cannot recover a lost encrypted password by any method.

Step 4	**service password-encryption** **Example:** Switch(config)# **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.

	Command or Action	Purpose
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Disabling Password Recovery

Follow these steps to disable password recovery to protect the security of your switch:

Before you begin

If you disable password recovery, we recommend that you keep a backup copy of the configuration file on a secure server in case the end user interrupts the boot process and sets the system back to default values. Do not keep a backup copy of the configuration file on the switch. If the switch is operating in VTP transparent mode, we recommend that you also keep a backup copy of the VLAN database file on a secure server. When the switch is returned to the default system configuration, you can download the saved files to the switch by using the Xmodem protocol.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **system disable password recovery switch** {*all* | *<1-9>*}
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	**Purpose**
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.

Step 3	**system disable password recovery switch** {*all* | *<1-9>*}
	
	Example:
	
	Switch(config)# **system disable password recovery switch all**
		Disables password recovery.
		
		    • *all* - Sets the configuration on switches in stack.
		    • *<1-9>* - Sets the configuration on the Switch Number selected.
		
		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.

Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

What to do next

To remove **disable password recovery**, use the **no system disable password recovery switch all** global configuration command.

Setting a Telnet Password for a Terminal Line

Beginning in user EXEC mode, follow these steps to set a Telnet password for the connected terminal line:

Before you begin

• Attach a PC or workstation with emulation software to the switch console port, or attach a PC to the Ethernet management 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.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **line vty 0 15**
4. **password** *password*
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	**Note** If a password is required for access to privileged EXEC mode, you will be prompted for it. Enters privileged EXEC mode.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**line vty 0 15** **Example:** Switch(config)# **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.
Step 4	**password** *password* **Example:** Switch(config-line)# **password abcxyz543**	Sets a Telnet password for the line or lines. For *password*, specify 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.
Step 5	**end** **Example:** Switch(config-line)# **end**	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Username and Password Pairs

Follow these steps to configure username and password pairs:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **username** *name* [**privilege** *level*] {**password** *encryption-type password*}
4. Use one of the following:

    - **line console 0**
    - **line vty 0 15**

5. **login local**
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. - Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**username** *name* [**privilege** *level*] {**password** *encryption-type password*} **Example:** Switch(config)# **username adamsample privilege 1 password secret456** Switch(config)# **username 111111111111 mac attribute**	Sets the username, privilege level, and password for each user. - For *name*, specify the user ID as one word or the MAC address. Spaces and quotation marks are not allowed. - You can configure a maximum of 12000 clients each, for both username and MAC filter. - (Optional) For *level*, specify 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. - For *encryption-type*, enter 0 to specify that an unencrypted password will follow. Enter 7 to specify that a hidden password will follow. - For *password*, specify 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.

	Command or Action	Purpose

Step 4	Use one of the following:
	  • **line console 0**
	  • **line vty 0 15**
	Example:
	Switch(config)# **line console 0**
	
	or
	
	Switch(config)# **line vty 15**
		Enters line configuration mode, and configures the console port (line 0) or the VTY lines (line 0 to 15).

Step 5	**login local** **Example:** Switch(config-line)# **login local**	Enables local password checking at login time. Authentication is based on the username specified in Step 3.
Step 6	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 7	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 8	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Setting the Privilege Level for a Command

Follow these steps to set the privilege level for a command:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **privilege** *mode* **level** *level command*
4. **enable password level** *level password*
5. **end**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**privilege** *mode* **level** *level command* **Example:** Switch(config)# **privilege exec level 14 configure**	Sets the privilege level for a command. • For *mode*, enter **configure** for global configuration mode, **exec** for EXEC mode, **interface** for interface configuration mode, or **line** for line configuration mode. • For *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. • For *command*, specify the command to which you want to restrict access.
Step 4	**enable password level** *level password* **Example:** Switch(config)# **enable password level 14 SecretPswd14**	Specifies the password to enable the privilege level. • For *level*, the range is from 0 to 15. Level 1 is for normal user EXEC mode privileges. • For *password*, specify 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.
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Changing the Default Privilege Level for Lines

Follow these steps to change the default privilege level for the specified line:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **line vty** *line*
4. **privilege level** *level*
5. **end**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**line vty** *line* **Example:** Switch(config)# **line vty 10**	Selects the virtual terminal line on which to restrict access.
Step 4	**privilege level** *level* **Example:** Switch(config)# **privilege level 15**	Changes the default privilege level for the line. For *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.
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**copy running-config startup-config** **Example:**	(Optional) Saves your entries in the configuration file.

Command or Action	Purpose
Switch# **copy running-config startup-config**	

What to do next

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.

Logging into and Exiting a Privilege Level

Beginning in user EXEC mode, follow these steps to log into a specified privilege level and exit a specified privilege level.

SUMMARY STEPS

1. **enable** *level*
2. **disable** *level*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** *level* **Example:** Switch> **enable 15**	Logs in to a specified privilege level. Following the example, Level 15 is privileged EXEC mode. For *level*, the range is 0 to 15.
Step 2	**disable** *level* **Example:** Switch# **disable 1**	Exits to a specified privilege level. Following the example, Level 1 is user EXEC mode. For *level*, the range is 0 to 15.

Monitoring Switch Access

Table 73: Commands for Displaying DHCP Information

show privilege	Displays the privilege level configuration.

Configuration Examples for Setting Passwords and Privilege Levels

Example: Setting or Changing a Static Enable Password

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
```

Example: Protecting Enable and Enable Secret Passwords with Encryption

This example shows how to configure the encrypted password *1FaD0$Xyti5Rkls3LoyxzS8* for privilege level 2:

```
Switch(config)# enable secret level 2 5 $1$FaD0$Xyti5Rkls3LoyxzS8
```

Example: Setting a Telnet Password for a Terminal Line

This example shows how to set the Telnet password to *let45me67in89*:

```
Switch(config)# line vty 10
Switch(config-line)# password let45me67in89
```

Example: Setting the Privilege Level for a Command

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
```

Additional References

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Configuring TACACS+

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Controlling Switch Access with Terminal Access Controller Access Control System Plus (TACACS+)

The following are the prerequisites for set up and configuration of switch access with Terminal Access Controller Access Control System Plus (TACACS+) (must be performed in the order presented):

1. Configure the switches with the TACACS+ server addresses.

2. Set an authentication key.

3. Configure the key from Step 2 on the TACACS+ servers.

4. Enable AAA.

5. Create a login authentication method list.

6. Apply the list to the terminal lines.

7. Create an authorization and accounting method list.

The following are the prerequisites for controlling switch access with TACACS+:

- You must have access to a configured TACACS+ server to configure TACACS+ features on your switch. Also, you must have access to TACACS+ services maintained in a database on a TACACS+ daemon typically running on a LINUX or Windows workstation.

- We recommend a redundant connection between a switch stack and the TACACS+ server. This is to help ensure that the TACACS+ server remains accessible in case one of the connected stack members is removed from the switch stack.

- You need a system running the TACACS+ daemon software to use TACACS+ on your switch.

- To use TACACS+, it must be enabled.

- Authorization must be enabled on the switch to be used.

- Users must first successfully complete TACACS+ authentication before proceeding to TACACS+ authorization.

- To use any of the AAA commands listed in this section or elsewhere, you must first enable AAA with the **aaa new-model** command.

- 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.

- 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.

- 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+.

Information About TACACS+

TACACS+ and Switch Access

This section describes TACACS+. TACACS+ provides detailed accounting information and flexible administrative control over the authentication and authorization processes. It is facilitated through authentication, authorization, accounting (AAA) and can be enabled only through AAA commands.

Related Topics

TACACS+ Overview

TACACS+ is a security application that provides centralized validation of users attempting to gain access to 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.

Figure 70: Typical TACACS+ Network Configuration
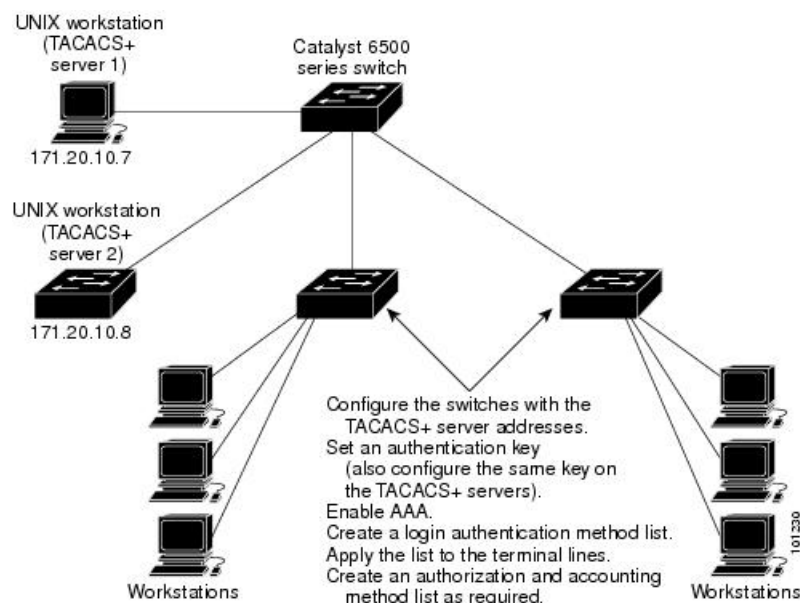
TACACS+, administered through the AAA security services, can provide these services:

- 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.

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

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, 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.

TACACS+ Configuration Options

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

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.

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.

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.

How to Configure TACACS+

This section describes how to configure your switch to support TACACS+.

Identifying the TACACS+ Server Host and Setting the Authentication Key

Follow these steps to identify the TACACS+ server host and set the authentication key:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **tacacs-server host** *hostname*
4. **aaa new-model**
5. **aaa group server tacacs+** *group-name*
6. **server** *ip-address*
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**tacacs-server host** *hostname* **Example:** Switch(config)# **tacacs-server host yourserver**	Identifies the IP host or hosts maintaining a TACACS+ server. Enter this command multiple times to create a list of preferred hosts. The software searches for hosts in the order in which you specify them. For *hostname*, specify the name or IP address of the host.
Step 4	**aaa new-model** **Example:** Switch(config)# **aaa new-model**	Enables AAA.

	Command or Action	Purpose
Step 5	**aaa group server tacacs+** *group-name* **Example:** Switch(config)# **aaa group server tacacs+** **your_server_group**	(Optional) Defines the AAA server-group with a group name. This command puts the Switch in a server group subconfiguration mode.
Step 6	**server** *ip-address* **Example:** Switch(config)# **server 10.1.2.3**	(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 3.
Step 7	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 8	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 9	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring TACACS+ Login Authentication

Follow these steps to configure TACACS+ login authentication:

Before you begin

To configure AAA authentication, you define a named list of authentication methods and then apply that list to various ports.

Note 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.

For more information about the **ip http authentication** command, see the *Cisco IOS Security Command Reference, Release 12.4*.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa authentication login** {**default** | *list-name*} *method1* [*method2...*]
5. **line** [**console** | **tty** | **vty**] *line-number* [*ending-line-number*]
6. **login authentication** {**default** | *list-name*}
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

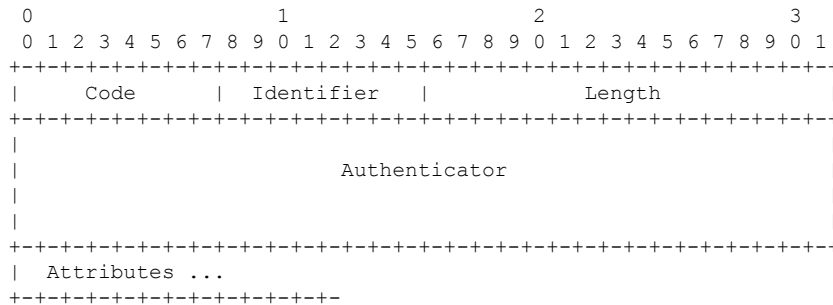|	**Command or Action**	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**aaa new-model** **Example:** Switch(config)# **aaa new-model**	Enables AAA.
Step 4	**aaa authentication login** {**default**	*list-name*} *method1* [*method2...*] **Example:** Switch(config)# **aaa authentication login default tacacs+ local**	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. • For *list-name*, specify a character string to name the list you are creating. • For *method1...*, specify 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:

	Command or Action	**Purpose**
		• *enable*—Use 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.
		• *line* —Use 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*—Use the local username database for authentication. You must enter username information in the database. Use the **username** *password* global configuration command.
		• *local-case*—Use 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*—Do not use any authentication for login.
Step 5	**line** [**console** | **tty** | **vty**] *line-number* [*ending-line-number*] **Example:** Switch(config)# **line 2 4**	Enters line configuration mode, and configures the lines to which you want to apply the authentication list.
Step 6	**login authentication** {**default** | *list-name*} **Example:** Switch(config-line)# **login authentication default**	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. • For *list-name*, specify the list created with the **aaa authentication login** command.
Step 7	**end** **Example:** Switch(config-line)# **end**	Returns to privileged EXEC mode.
Step 8	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.

	Command or Action	**Purpose**
Step 9	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring TACACS+ Authorization for Privileged EXEC Access and Network Services

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.

> **Note** Authorization is bypassed for authenticated users who log in through the CLI even if authorization has been configured.

Follow these steps to specify TACACS+ authorization for privileged EXEC access and network services:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa authorization network tacacs+**
4. **aaa authorization exec tacacs+**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** `Switch> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 3	**aaa authorization network tacacs+** **Example:**	Configures the switch for user TACACS+ authorization for all network-related service requests.

	Command or Action	Purpose
	Switch(config)# **aaa authorization network tacacs+**	
Step 4	**aaa authorization exec tacacs+** **Example:** Switch(config)# **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).
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Starting TACACS+ Accounting

Follow these steps to start TACACS+ Accounting:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa accounting network start-stop tacacs+**
4. **aaa accounting exec start-stop tacacs+**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
	`Switch> enable`	
Step 2	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 3	**aaa accounting network start-stop tacacs+** **Example:** `Switch(config)# aaa accounting network start-stop tacacs+`	Enables TACACS+ accounting for all network-related service requests.
Step 4	**aaa accounting exec start-stop tacacs+** **Example:** `Switch(config)# 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.
Step 5	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

What to do next

To establish a session with a router if the AAA server is unreachable, use the **aaa accounting system guarantee-first** command. It 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.

Establishing a Session with a Router if the AAA Server is Unreachable

To establishing a session with a router if the AAA server is unreachable, use the **aaa accounting system guarantee-first** command. It 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.

Monitoring TACACS+

Table 74: Commands for Displaying TACACS+ Information

Command	Purpose
show tacacs	Displays TACACS+ server statistics.

Additional References for Configuring Secure Shell

Related Documents

Related Topic	Document Title
Configuring Identity Control policies and Identity Service templates for Session Aware networking.	Session Aware Networking Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)
Configuring RADIUS, TACACS+, Secure Shell, 802.1X and AAA.	Securing User Services Configuration Guide Library, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for TACACS+

Release	Feature Information
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Configuring RADIUS

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Controlling Switch Access with RADIUS

This section lists the prerequisites for controlling Switch access with RADIUS.

General:

• RADIUS and AAA must be enabled to use any of the configuration commands in this chapter.

• RADIUS is facilitated through AAA and can be enabled only through AAA commands.

• 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.

- You should have access to and should configure a RADIUS server before configuring RADIUS features on your Switch.

- 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.

- To use the Change-of-Authorization (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.

For RADIUS operation:

- Users must first successfully complete RADIUS authentication before proceeding to RADIUS authorization, if it is enabled.

Related Topics

Restrictions for Controlling Switch Access with RADIUS

This topic covers restrictions for controlling Switch access with RADIUS.

General:

- To prevent a lapse in security, you cannot configure RADIUS through a network management application.

RADIUS is not suitable in the following 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.

Related Topics

Information about RADIUS

RADIUS and Switch Access

This section describes how to enable and configure RADIUS. RADIUS provides detailed accounting information and flexible administrative control over the authentication and authorization processes.

Related Topics

RADIUS Overview

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.

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. See Figure 2: Transitioning from RADIUS to TACACS+ Services below.

- 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 Chapter 11, "Configuring IEEE 802.1x Port-Based Authentication."

- 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.

Figure 71: Transitioning from RADIUS to TACACS+ Services
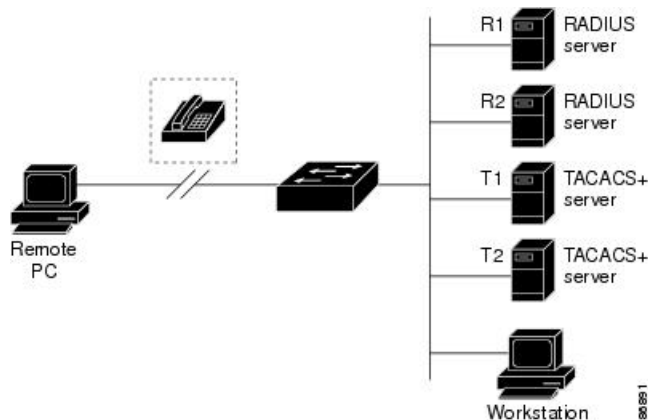
Related Topics

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 the following responses from the RADIUS server:

   - ACCEPT—The user is authenticated.

   - REJECT—The user is either not authenticated and is prompted to re-enter the username and password, or access is denied.

   - CHALLENGE—A challenge requires additional data from the user.

   - 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. 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

Related Topics

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).

- Change-of-Authorization Requests

- CoA Request Response Code

- CoA Request Commands

- Session Reauthentication

- Stacking Guidelines for Session Termination

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

This feature is integrated with Cisco Secure Access Control Server (ACS) 5.1.

The RADIUS interface is enabled by default on Catalyst switches. However, some basic configuration is required for the following attributes:

- Security and Password—refer to the "Preventing Unauthorized Access to Your Switch" section in this guide.

- Accounting—refer to the "Starting RADIUS Accounting" section in the Configuring Switch-Based Authentication chapter in this guide.

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 acknowledgment (ACK) [CoA-ACK]

- CoA non-acknowledgment (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.

This table shows the IETF attributes are supported for this feature.

Table 75: Supported IETF Attributes

Attribute Number	Attribute Name
24	State
31	Calling-Station-ID
44	Acct-Session-ID
80	Message-Authenticator
101	Error-Cause

This table shows the possible values for the Error-Cause attribute.

Table 76: Error-Cause Values

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

Preconditions

To use the 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.

CoA Request Response Code

The CoA Request response code can be used to convey a command to the switch.

Related Topics

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.

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 acknowledgment (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 vendor-specific attributes (VSAs).

Related Topics

CoA ACK Response Code

If the authorization state is changed successfully, a positive acknowledgment (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 acknowledgment (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 77: CoA Commands Supported on the switch

Command **7**	Cisco VSA
Reauthenticate host	Cisco:Avpair="subscriber:command=reauthenticate"

Command [7]	Cisco VSA
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"

[7] All CoA commands must include the session identifier between the switch and the CoA client.

Related Topics

CoA Request Response Code, on page 716

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 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 EAPoL (Extensible Authentication Protocol over Lan) -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.

Session Reauthentication in a Switch Stack

When a switch stack receives a session reauthentication message:

- It checkpoints the need for a re-authentication before returning an acknowledgment (ACK).

- It initiates reauthentication for the appropriate session.

- If authentication completes with either success or failure, the signal that triggered the reauthentication is removed from the stack's member switch.

- If the stack's active switch fails before authentication completes, reauthentication is initiated after active switch changeover based on the original command (which is subsequently removed).

- If the active switch fails before sending an ACK, the new active switch treats the re-transmitted command as a new command.

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 re-initialization 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, re-enable 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 re-enable the port).

CoA Disconnect-Request

This command is a standard Disconnect-Request. 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 re-sent from the client. If the session is not found following re-sending, a Disconnect-ACK is sent with the "Session Context Not Found" error-code attribute.

Related Topics

Session Identification, on page 716

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"

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 re-sent 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 re-sending 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.

Related Topics

Session Identification, on page 716

CoA Request: Bounce-Port

This command is carried in a standard CoA-Request message that contains the following 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. 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, re-enables 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 re-sent 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 re-started on the new active switch.

Related Topics

Stacking Guidelines for Session Termination

No special handling is required for CoA Disconnect-Request messages in a switch stack.

Stacking Guidelines for CoA-Request Bounce-Port

Because the **bounce-port** command is targeted at a session, not a port, if the session is not found, the command cannot be executed.

When the Auth Manager command handler on the active switch receives a valid **bounce-port** command, it checkpoints the following information before returning a CoA-ACK message:

- the need for a port-bounce

- the port-id (found in the local session context)

The switch initiates a port-bounce (disables the port for 10 seconds, then re-enables it).

If the port-bounce is successful, the signal that triggered the port-bounce is removed from the standby switch.

If the active switch fails before the port-bounce completes, a port-bounce is initiated after an active switch changeover based on the original command (which is subsequently removed).

If the active switch fails before sending a CoA-ACK message, the new active switch treats the re-sent command as a new command.

Stacking Guidelines for CoA-Request Disable-Port

Because the **disable-port** command is targeted at a session, not a port, if the session is not found, the command cannot be executed.

When the Auth Manager command handler on the active switch receives a valid **disable-port** command, it verifies this information before returning a CoA-ACK message:

- the need for a port-disable

- the port-id (found in the local session context)

The switch attempts to disable the port.

If the port-disable operation is successful, the signal that triggered the port-disable is removed from the standby switch.

If the active switch fails before the port-disable operation completes, the port is disabled after an active switch changeover based on the original command (which is subsequently removed).

If the active switch fails before sending a CoA-ACK message, the new active switch treats the re-sent command as a new command.

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 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.

Related Topics

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. The default method list is automatically applied to all ports except those that have a named method list explicitly defined.

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.

Related Topics

Configuring RADIUS Login Authentication, on page 737

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 fail-over backup to the first one.

Related Topics

Defining AAA Server Groups, on page 739

AAA Authorization

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.

Related Topics

Configuring RADIUS Authorization for User Privileged Access and Network Services, on page 742

RADIUS Accounting

The AAA accounting feature tracks the services that users are using and the amount of network resources that they are consuming. When you enable AAA accounting, 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. You can then analyze the data for network management, client billing, or auditing.

Related Topics

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 attributevalue (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, the following AV pair causes Cisco's "multiple named IP address pools" feature to be activated during IP authorization (during PPP's Internet Protocol Control Protocol (IPCP) address assignment):

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

If you insert an "*", the AV pair "ip:addr-pool=first" becomes optional. Note that any AV pair can be made optional:

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

The following example shows how to cause a user logging in from a network access server to have immediate access to EXEC commands:

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

Other vendors have their own unique vendor-IDs, options, and associated VSAs. For more information about vendor-IDs and VSAs, see RFC 2138, "Remote Authentication Dial-In User Service (RADIUS)."

Attribute 26 contains the following three elements:

   • Type

   • Length

   • String (also known as data)

        • Vendor-Id
        • Vendor-Type
        • Vendor-Length
        • Vendor-Data

The figure below shows the packet format for a VSA encapsulated "behind" attribute 26.

Figure 72: VSA Encapsulated Behind Attribute 26
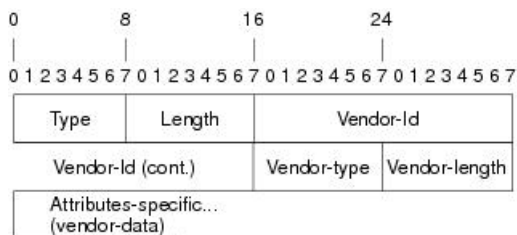
Note	It is up to the vendor to specify the format of their VSA. The Attribute-Specific field (also known as Vendor-Data) is dependent on the vendor's definition of that attribute.

The table below describes significant fields listed in the Vendor-Specific RADIUS IETF Attributes table (second table below), which lists supported vendor-specific RADIUS attributes (IETF attribute 26).

Table 78: Vendor-Specific Attributes Table Field Descriptions

Field	Description
Number	All attributes listed in the following table are extensions of IETF attribute 26.
Vendor-Specific Command Codes	A defined code used to identify a particular vendor. Code 9 defines Cisco VSAs, 311 defines Microsoft VSAs, and 529 defines Ascend VSAs.
Sub-Type Number	The attribute ID number. This number is much like the ID numbers of IETF attributes, except it is a "second layer" ID number encapsulated behind attribute 26.
Attribute	The ASCII string name of the attribute.
Description	Description of the attribute.

Table 79: Vendor-Specific RADIUS IETF Attributes

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
MS-CHAP Attributes				
26	311	1	MSCHAP-Response	Contains the response value provided by a PPP MS-CHAP user in response to the challenge. It is only used in Access-Request packets. This attribute is identical to the PPP CHAP Identifier. (RFC 2548

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	311	11	MSCHAP-Challenge	Contains the challenge sent by a network access server to an MS-CHAP user. It can be used in both Access-Request and Access-Challenge packets. (RFC 2548)
VPDN Attributes				
26	9	1	l2tp-cm-local-window-size	Specifies the maximum receive window size for L2TP control messages. This value is advertised to the peer during tunnel establishment.
26	9	1	l2tp-drop-out-of-order	Respects sequence numbers on data packets by dropping those that are received out of order. This does not ensure that sequence numbers will be sent on data packets, just how to handle them if they are received.
26	9	1	l2tp-hello-interval	Specifies the number of seconds for the hello keepalive interval. Hello packets are sent when no data has been sent on a tunnel for the number of seconds configured here.
26	9	1	l2tp-hidden-avp	When enabled, sensitive AVPs in L2TP control messages are scrambled or hidden.
26	9	1	l2tp-nosession-timeout	Specifies the number of seconds that a tunnel will stay active with no sessions before timing out and shutting down.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	1	tunnel-tos-reflect	Copies the IP ToS field from the IP header of each payload packet to the IP header of the tunnel packet for packets entering the tunnel at the LNS.
26	9	1	l2tp-tunnel-authen	If this attribute is set, it performs L2TP tunnel authentication.
26	9	1	l2tp-tunnel-password	Shared secret used for L2TP tunnel authentication and AVP hiding.
26	9	1	l2tp-udp-checksum	This is an authorization attribute and defines whether L2TP should perform UDP checksums for data packets. Valid values are "yes" and "no." The default is no.
Store and Forward Fax Attributes				
26	9	3	Fax-Account-Id-Origin	Indicates the account ID origin as defined by system administrator for the **mmoip aaa receive-id** or the **mmoip aaa send-id** commands.
26	9	4	Fax-Msg-Id=	Indicates a unique fax message identification number assigned by Store and Forward Fax.
26	9	5	Fax-Pages	Indicates the number of pages transmitted or received during this fax session. This page count includes cover pages.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	6	Fax-Coverpage-Flag	Indicates whether or not a cover page was generated by the off-ramp gateway for this fax session. True indicates that a cover page was generated; false means that a cover page was not generated.
26	9	7	Fax-Modem-Time	Indicates the amount of time in seconds the modem sent fax data (x) and the amount of time in seconds of the total fax session (y), which includes both fax-mail and PSTN time, in the form x/y. For example, 10/15 means that the transfer time took 10 seconds, and the total fax session took 15 seconds.
26	9	8	Fax-Connect-Speed	Indicates the modem speed at which this fax-mail was initially transmitted or received. Possible values are 1200, 4800, 9600, and 14400.
26	9	9	Fax-Recipient-Count	Indicates the number of recipients for this fax transmission. Until e-mail servers support Session mode, the number should be 1.
26	9	10	Fax-Process-Abort-Flag	Indicates that the fax session was cancelled or successful. True means that the session was cancelled; false means that the session was successful.
26	9	11	Fax-Dsn-Address	Indicates the address to which DSNs will be sent.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	12	Fax-Dsn-Flag	Indicates whether or not DSN has been enabled. True indicates that DSN has been enabled; false means that DSN has not been enabled.
26	9	13	Fax-Mdn-Address	Indicates the address to which MDNs will be sent.
26	9	14	Fax-Mdn-Flag	Indicates whether or not message delivery notification (MDN) has been enabled. True indicates that MDN had been enabled; false means that MDN had not been enabled.
26	9	15	Fax-Auth-Status	Indicates whether or not authentication for this fax session was successful. Possible values for this field are success, failed, bypassed, or unknown.
26	9	16	Email-Server-Address	Indicates the IP address of the e-mail server handling the on-ramp fax-mail message.
26	9	17	Email-Server-Ack-Flag	Indicates that the on-ramp gateway has received a positive acknowledgment from the e-mail server accepting the fax-mail message.
26	9	18	Gateway-Id	Indicates the name of the gateway that processed the fax session. The name appears in the following format: hostname.domain-name.
26	9	19	Call-Type	Describes the type of fax activity: fax receive or fax send.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	20	Port-Used	Indicates the slot/port number of the Cisco AS5300 used to either transmit or receive this fax-mail.
26	9	21	Abort-Cause	If the fax session cancels, indicates the system component that signaled the cancel operation. Examples of system components that could trigger a cancel operation are FAP (Fax Application Process), TIFF (the TIFF reader or the TIFF writer), fax-mail client, fax-mail server, ESMTP client, or ESMTP server.
H323 Attributes				
26	9	23	Remote-Gateway-ID (h323-remote-address)	Indicates the IP address of the remote gateway.
26	9	24	Connection-ID (h323-conf-id)	Identifies the conference ID.
26	9	25	Setup-Time (h323-setup-time)	Indicates the setup time for this connection in Coordinated Universal Time (UTC) formerly known as Greenwich Mean Time (GMT) and Zulu time.
26	9	26	Call-Origin (h323-call-origin)	Indicates the origin of the call relative to the gateway. Possible values are originating and terminating (answer).
26	9	27	Call-Type (h323-call-type)	Indicates call leg type. Possible values are **telephony** and **VoIP**.
26	9	28	Connect-Time (h323-connect-time)	Indicates the connection time for this call leg in UTC.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	29	Disconnect-Time (h323-disconnect-time)	Indicates the time this call leg was disconnected in UTC.
26	9	30	Disconnect-Cause (h323-disconnect-cause)	Specifies the reason a connection was taken offline per Q.931 specification.
26	9	31	Voice-Quality (h323-voice-quality)	Specifies the impairment factor (ICPIF) affecting voice quality for a call.
26	9	33	Gateway-ID (h323-gw-id)	Indicates the name of the underlying gateway.
Large Scale Dialout Attributes				
26	9	1	callback-dialstring	Defines a dialing string to be used for callback.
26	9	1	data-service	No description available.
26	9	1	dial-number	Defines the number to dial.
26	9	1	force-56	Determines whether the network access server uses only the 56 K portion of a channel, even when all 64 K appear to be available.
26	9	1	map-class	Allows the user profile to reference information configured in a map class of the same name on the network access server that dials out.
26	9	1	send-auth	Defines the protocol to use (PAP or CHAP) for username-password authentication following CLID authentication.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	1	send-name	PPP name authentication. To apply for PAP, do not configure the **ppp pap sent-name password** command on the interface. For PAP, "preauth:send-name" and "preauth:send-secret" will be used as the PAP username and PAP password for outbound authentication. For CHAP, "preauth:send-name" will be used not only for outbound authentication, but also for inbound authentication. For a CHAP inbound case, the NAS will use the name defined in "preauth:send-name" in the challenge packet to the caller box. **Note** The send-name attribute has changed over time: Initially, it performed the functions now provided by both the send-name and remote-name attributes. Because the remote-name attribute has been added, the send-name attribute is restricted to its current behavior.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	1	send-secret	PPP password authentication. The vendor-specific attributes (VSAs) "preauth:send-name" and "preauth:send-secret" will be used as the PAP username and PAP password for outbound authentication. For a CHAP outbound case, both "preauth:send-name" and "preauth:send-secret" will be used in the response packet.
26	9	1	remote-name	Provides the name of the remote host for use in large-scale dial-out. Dialer checks that the large-scale dial-out remote name matches the authenticated name, to protect against accidental user RADIUS misconfiguration. (For example, dialing a valid phone number but connecting to the wrong device.)
Miscellaneous Attributes				

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	2	Cisco-NAS-Port	Specifies additional vendor specific attribute (VSA) information for NAS-Port accounting. To specify additional NAS-Port information in the form an Attribute-Value Pair (AVPair) string, use the **radius-server vsa send** global configuration command. **Note** This VSA is typically used in Accounting, but may also be used in Authentication (Access-Request) packets.
26	9	1	min-links	Sets the minimum number of links for MLP.
26	9	1	proxyacl#<n>	Allows users to configure the downloadable user profiles (dynamic ACLs) by using the authentication proxy feature so that users can have the configured authorization to permit traffic going through the configured interfaces.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	1	spi	Carries the authentication information needed by the home agent to authenticate a mobile node during registration. The information is in the same syntax as the **ip mobile secure host <addr>** configuration command. Basically it contains the rest of the configuration command that follows that string, verbatim. It provides the Security Parameter Index (SPI), key, authentication algorithm, authentication mode, and replay protection timestamp range.

Related Topics

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.

Related Topics

How to Configure RADIUS

Identifying the RADIUS Server Host

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.

You can configure the Switch to use AAA server groups to group existing server hosts for authentication. For more information, see Related Topics below.

You also need to configure some settings on the RADIUS server. These settings include the IP address of the Switch and the key string to be shared by both the server and the Switch. For more information, see the RADIUS server documentation.

Follow these steps to configure per-server RADIUS server communication.

Before you begin

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 Related Topics below.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **radius-server host** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**radius-server host** {*hostname*	*ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*] **Example:** Switch(config)# **radius-server host 172.29.36.49 auth-port 1612 key rad1**	Specifies the IP address or hostname of the remote RADIUS server host. • (Optional) For **auth-port** *port-number*, specify the UDP destination port for authentication requests. • (Optional) For **acct-port** *port-number*, specify the UDP destination port for accounting requests. • (Optional) For **timeout** *seconds*, specify the time interval that the Switch waits for the RADIUS server to reply before resending. The range is 1 to 1000. This

	Command or Action	**Purpose**
		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) For **retransmit** *retries*, specify 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) For **key** *string*, specify 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.
Step 4	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:**	(Optional) Saves your entries in the configuration file.

Command or Action	Purpose
Switch# **copy running-config startup-config**	

Related Topics

RADIUS Server Host, on page 721

Defining AAA Server Groups, on page 739

Configuring Settings for All RADIUS Servers, on page 745

Configuring RADIUS Login Authentication

Follow these steps to configure RADIUS 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.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa authentication login** {**default** | *list-name*} *method1* [*method2...*]
5. **line** [**console** | **tty** | **vty**] *line-number* [*ending-line-number*]
6. **login authentication** {**default** | *list-name*}
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.

	Command or Action	**Purpose**
Step 3	**aaa new-model** **Example:** Switch(config)# **aaa new-model**	Enables AAA.

Step 4	**aaa authentication login** {**default** | *list-name*} *method1* [*method2...*]
	
	Example:
	
	Switch(config)# **aaa authentication login default local**
		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.
		
		• For *list-name*, specify a character string to name the list you are creating.
		
		• For *method1...*, specify 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*—Use 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*—Use RADIUS authentication. Before you can use this authentication method, you must configure the RADIUS server.
		
		  • *line*—Use 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*—Use the local username database for authentication. You must enter username information in the database. Use the **username** *name* **password** global configuration command.
		
		  • *local-case*—Use 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*—Do not use any authentication for login.

	Command or Action	Purpose
Step 5	**line** [**console** | **tty** | **vty**] *line-number* [*ending-line-number*] **Example:** Switch(config)# **line 1 4**	Enters line configuration mode, and configure the lines to which you want to apply the authentication list.
Step 6	**login authentication** {**default** | *list-name*} **Example:** Switch(config)# **login authentication default**	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. • For *list-name*, specify the list created with the **aaa authentication login** command.
Step 7	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 8	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 9	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Related Topics

Defining AAA Server Groups

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.

Follow these steps to define AAA server groups:

SUMMARY STEPS

1. **enable**
2. **configure terminal**

3. **radius-server host** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]
4. **aaa new-model**
5. **aaa group server radius** *group-name*
6. **server** *ip-address*
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**radius-server host** {*hostname*	*ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*] **Example:** Switch(config)# **radius-server host 172.29.36.49 auth-port 1612 key rad1**	Specifies the IP address or hostname of the remote RADIUS server host. • (Optional) For **auth-port** *port-number*, specify the UDP destination port for authentication requests. • (Optional) For **acct-port** *port-number*, specify the UDP destination port for accounting requests. • (Optional) For **timeout** *seconds*, specify 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) For **retransmit** *retries*, specify 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) For **key** *string*, specify the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server.

	Command or Action	**Purpose**
		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.
Step 4	**aaa new-model** **Example:** Switch(config)# **aaa new-model**	Enables AAA.
Step 5	**aaa group server radius** *group-name* **Example:** Switch(config)# **aaa group server radius group1**	Defines the AAA server-group with a group name. This command puts the switch in a server group configuration mode.
Step 6	**server** *ip-address* **Example:** Switch(config-sg-radius)# **server 172.20.0.1 auth-port 1000 acct-port 1001**	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.
Step 7	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 8	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.

	Command or Action	Purpose
Step 9	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Related Topics

Configuring RADIUS Authorization for User Privileged Access and Network Services

Note Authorization is bypassed for authenticated users who log in through the CLI even if authorization has been configured.

Follow these steps to configure RADIUS authorization for user priviledged access and network services:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa authorization network radius**
4. **aaa authorization exec radius**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**aaa authorization network radius** **Example:** Switch(config)# **aaa authorization network radius**	Configures the switch for user RADIUS authorization for all network-related service requests.
Step 4	**aaa authorization exec radius** **Example:** Switch(config)# **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).
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

What to do next

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.

Related Topics

AAA Authorization, on page 722

Starting RADIUS Accounting

Follow these steps to start RADIUS accounting:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa accounting network start-stop radius**
4. **aaa accounting exec start-stop radius**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**aaa accounting network start-stop radius** **Example:** Switch(config)# **aaa accounting network start-stop radius**	Enables RADIUS accounting for all network-related service requests.
Step 4	**aaa accounting exec start-stop radius** **Example:** Switch(config)# **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.
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.

	Command or Action	Purpose
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

What to do next

To establishing a session with a router if the AAA server is unreachable, use the **aaa accounting system guarantee-first** command. This 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.

Related Topics

RADIUS Accounting, on page 722

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.

Configuring Settings for All RADIUS Servers

Beginning in privileged EXEC mode, follow these steps to configure settings for all RADIUS servers:

SUMMARY STEPS

1. **configure terminal**
2. **radius-server key** *string*
3. **radius-server retransmit** *retries*
4. **radius-server timeout** *seconds*
5. **radius-server deadtime** *minutes*
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 2	**radius-server key** *string* Example: Switch(config)# **radius-server key your_server_key**	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.
Step 3	**radius-server retransmit** *retries* Example: Switch(config)# **radius-server retransmit 5**	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.
Step 4	**radius-server timeout** *seconds* Example: Switch(config)# **radius-server timeout 3**	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.
Step 5	**radius-server deadtime** *minutes* Example: Switch(config)# **radius-server deadtime 0**	When a RADIUS server is not responding to authentication requests, this command specifies a time to stop the request on that server. This avoids the wait for the request to timeout before trying the next configured server. The default is 0; the range is 1 to 1440 minutes.
Step 6	**end** Example: Switch(config)# **end**	Returns to privileged EXEC mode.
Step 7	**show running-config** Example: Switch# **show running-config**	Verifies your entries.

	Command or Action	Purpose
Step 8	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.
	Example:	
	`Switch# copy running-config startup-config`	

Related Topics

Configuring the Switch to Use Vendor-Specific RADIUS Attributes

Follow these steps to configure the switch to use vendor-specific RADIUS attributes:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **radius-server vsa send** [**accounting** | **authentication**]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode.
	Example:	• Enter your password if prompted.
	`Switch> enable`	
Step 2	**configure terminal**	Enters global configuration mode.
	Example:	
	`Switch# configure terminal`	
Step 3	**radius-server vsa send** [**accounting**	**authentication**]	Enables the switch to recognize and use VSAs as defined by RADIUS IETF attribute 26.
	Example:	• (Optional) Use the **accounting** keyword to limit the set of recognized vendor-specific attributes to only accounting attributes.
	`Switch(config)# radius-server vsa send`	• (Optional) Use the **authentication** keyword to limit the set of recognized vendor-specific attributes to only authentication attributes.

	Command or Action	Purpose
		If you enter this command without keywords, both accounting and authentication vendor-specific attributes are used.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Related Topics

Vendor-Specific RADIUS Attributes, on page 723

Configuring the Switch for Vendor-Proprietary RADIUS Server Communication

Follow these steps to configure the switch to use vendor-proprietary RADIUS server communication:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **radius-server host** {*hostname* | *ip-address*} **non-standard**
4. **radius-server key** *string*
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
	`Switch> enable`	
Step 2	**configure terminal**	Enters global configuration mode.
	Example:	
	`Switch# configure terminal`	
Step 3	**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.
	Example:	
	`Switch(config)# radius-server host 172.20.30.15 nonstandard`	
Step 4	**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.
	Example:	
	`Switch(config)# radius-server key rad124`	**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.
Step 5	**end**	Returns to privileged EXEC mode.
	Example:	
	`Switch(config)# end`	
Step 6	**show running-config**	Verifies your entries.
	Example:	
	`Switch# show running-config`	
Step 7	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.
	Example:	
	`Switch# copy running-config startup-config`	

What to do next

This feature allows access and authentication requests to be evenly across all RADIUS servers in a server group. For more information, see the "RADIUS Server Load Balancing" chapter of the *Cisco IOS Security Configuration Guide, Release 12.4*.

Related Topics

Configuring CoA on the Switch

Follow these steps to configure CoA on a switch. This procedure is required.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa server radius dynamic-author**
5. **client** {*ip-address* | *name*} [**vrf** *vrfname*] [**server-key** *string*]
6. **server-key** [**0** | **7**] *string*
7. **port** *port-number*
8. **auth-type** {**any** | **all** | **session-key**}
9. **ignore session-key**
10. **ignore server-key**
11. **authentication command bounce-port ignore**
12. **authentication command disable-port ignore**
13. **end**
14. **show running-config**
15. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**aaa new-model** **Example:**	Enables AAA.

	Command or Action	Purpose
	Switch(config)# **aaa new-model**	
Step 4	**aaa server radius dynamic-author** **Example:** Switch(config)# **aaa server radius dynamic-author**	Configures the switch as an authentication, authorization, and accounting (AAA) server to facilitate interaction with an external policy server.
Step 5	**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.
Step 6	**server-key** [**0** | **7**] *string* **Example:** Switch(config-sg-radius)# **server-key your_server_key**	Configures the RADIUS key to be shared between a device and RADIUS clients.
Step 7	**port** *port-number* **Example:** Switch(config-sg-radius)# **port 25**	Specifies the port on which a device listens for RADIUS requests from configured RADIUS clients.
Step 8	**auth-type** {**any** | **all** | **session-key**} **Example:** Switch(config-sg-radius)# **auth-type any**	Specifies the type of authorization the switch uses for RADIUS clients. The client must match all the configured attributes for authorization.
Step 9	**ignore session-key**	(Optional) Configures the switch to ignore the session-key. For more information about the **ignore** command, see the *Cisco IOS Intelligent Services Gateway Command Reference* on Cisco.com.
Step 10	**ignore server-key** **Example:** Switch(config-sg-radius)# **ignore server-key**	(Optional) Configures the switch to ignore the server-key. For more information about the **ignore** command, see the *Cisco IOS Intelligent Services Gateway Command Reference* on Cisco.com.
Step 11	**authentication command bounce-port ignore** **Example:** Switch(config-sg-radius)# **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.

	Command or Action	Purpose
Step 12	**authentication command disable-port ignore** **Example:** Switch(config-sg-radius)# **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. Use standard CLI or SNMP commands to re-enable the port.
Step 13	**end** **Example:** Switch(config-sg-radius)# **end**	Returns to privileged EXEC mode.
Step 14	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 15	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring RADIUS Server Load Balancing

This feature allows access and authentication requests to be evenly across all RADIUS servers in a server group. For more information, see the "RADIUS Server Load Balancing" chapter of the *Cisco IOS Security Configuration Guide, Release 12.4*.

Monitoring CoA Functionality

Table 80: Privileged EXEC show Commands

Command	Purpose
show aaa attributes protocol radius	Displays AAA attributes of RADIUS commands.

Table 81: Global Troubleshooting Commands

Command	Purpose
debug radius	Displays information for troubleshooting RADIUS.
debug aaa coa	Displays information for troubleshooting CoA processing.

Command	Purpose
debug aaa pod	Displays information for troubleshooting POD packets.
debug aaa subsys	Displays information for troubleshooting POD packets.
debug cmdhd [**detail**	**error**	**events**]	Displays information for troubleshooting command headers.

For detailed information about the fields in these displays, see the command reference for this release.

Configuration Examples for Controlling Switch Access with RADIUS

Examples: Identifying the RADIUS Server Host

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
```

Example: Using Two Different RADIUS Group Servers

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
```

Examples: Configuring the Switch to Use Vendor-Specific RADIUS Attributes

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"
```

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"
```

Example: Configuring the Switch for Vendor-Proprietary RADIUS Server Communication

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
```

Additional References for Configuring Secure Shell

Related Documents

Related Topic	Document Title
Configuring Identity Control policies and Identity Service templates for Session Aware networking.	Session Aware Networking Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)
Configuring RADIUS, TACACS+, Secure Shell, 802.1X and AAA.	Securing User Services Configuration Guide Library, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for RADIUS

Release	Feature Information
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

CHAPTER **35**

Configuring Kerberos

Prerequisites for Controlling Switch Access with Kerberos

The following are the prerequisites for controlling switch access with Kerberos.

- 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.

- 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.

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.

Restrictions for Controlling Switch Access with Kerberos

The following lists any restrictions for controlling switch access with Kerberos.

Information about Kerberos

This section provides Kerberos information.

Kerberos and Switch Access

This section describes how to enable and configure the Kerberos security system, which authenticates requests for network resources by using a trusted third party.

✎

Note   In the Kerberos configuration examples and in the *Cisco IOS Security Command Reference, Release 12.4*, the trusted third party can be a switch that supports Kerberos, that is configured as a network security server, and that can authenticate users by using the Kerberos protocol.

Kerberos Overview

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 life span, are stored in user credential caches. The Kerberos server uses the tickets instead of user names 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).

Kerberos supports these network services:

- Telnet

- rlogin

- rsh

This table lists the common Kerberos-related terms and definitions.

Table 82: Kerberos 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[8] 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 re-entering a username and password. Credentials have a default life span 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[9]	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[10]	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[11].
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.

Term	Definition
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.

[8] ticket granting ticket
[9] key distribution center
[10] key table
[11] 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, on page 760

2. Obtaining a TGT from a KDC, on page 761

3. Authenticating to Network Services, on page 761

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 re-entering 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.

For instructions about how to authenticate to a KDC, see the "Obtaining a TGT from a KDC" section in the "Security Server Protocols" chapter of the *Cisco IOS Security Configuration Guide, Release 12.4*.

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.

For instructions about how to authenticate to a network service, see the "Authenticating to Network Services" section in the "Security Server Protocols" chapter of the *Cisco IOS Security Configuration Guide, Release 12.4.*

How to Configure Kerberos

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.

For instructions, see the "Kerberos Configuration Task List" section in the "Security Server Protocols" chapter of the *Cisco IOS Security Configuration Guide, Release 12.4.*

Monitoring the Kerberos Configuration

To display the Kerberos configuration, use the **show running-config** privileged EXEC command.

Additional References

Related Documents

Related Topic	Document Title
Configuring Identity Control policies and Identity Service templates for Session Aware networking.	Session Aware Networking Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches) http://www.cisco.com/en/US/docs/ios-xml/ios/san/configuration/xe-3se/3850/san-xe-3se-3850-book.html
Configuring RADIUS, TACACS+, Secure Shell, 802.1X and AAA	Securing User Services Configuration Guide Library, Cisco IOS XE Release 3SE (Catalyst 3850 Switches) http://www.cisco.com/en/US/docs/ios-xml/ios/security/config_library/xe-3se/3850/secuser-xe-3se-3850-lib
Kerberos Commands	*Cisco IOS Security Command Reference, Release 12.4*
Kerberos Configuration Examples	*Cisco IOS Security Configuration Guide, Release 12.4.*

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for Kerberos

Release	Feature Information
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Configuring Local Authentication and Authorization

How to Configure Local Authentication and Authorization

Configuring the Switch for 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.

Note 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.

Follow these steps to configure AAA to operate without a server by setting the switch to implement AAA in local mode:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa authentication login default local**
5. **aaa authorization exec local**
6. **aaa authorization network local**
7. **username** *name* [**privilege** *level*] {**password** *encryption-type password*}
8. **end**
9. **show running-config**

10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**aaa new-model** **Example:** Switch(config)# **aaa new-model**	Enables AAA.
Step 4	**aaa authentication login default local** **Example:** Switch(config)# **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.
Step 5	**aaa authorization exec local** **Example:** Switch(config)# **aaa authorization exec local**	Configures user AAA authorization, check the local database, and allow the user to run an EXEC shell.
Step 6	**aaa authorization network local** **Example:** Switch(config)# **aaa authorization network local**	Configures user AAA authorization for all network-related service requests.
Step 7	**username** *name* [**privilege** *level*] {**password** *encryption-type password*} **Example:** Switch(config)# **username your_user_name privilege 1 password 7 secret567**	Enters the local database, and establishes a username-based authentication system. Repeat this command for each user. • For *name*, specify the user ID as one word. Spaces and quotation marks are not allowed. • (Optional) For *level*, specify the privilege level the user has after gaining access. The range is 0 to 15.

	Command or Action	Purpose
		Level 15 gives privileged EXEC mode access. Level 0 gives user EXEC mode access. • For *encryption-type*, enter 0 to specify that an unencrypted password follows. Enter 7 to specify that a hidden password follows. • For *password*, specify 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.
Step 8	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 9	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 10	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Related Topics

Monitoring Local Authentication and Authorization

To display Local Authentication and Authorization configuration, use the **show running-config** privileged EXEC command.

Additional References

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for Local Authentication and Authorization

Release	Feature Information
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Configuring Secure Shell (SSH)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Configuring the Switch for Secure Shell (SSH) and Secure Copy Protocol (SCP)

The following are the prerequisites for configuring the switch for secure shell (SSH):

- For SSH to work, the switch needs an RSA public/private key pair. This is the same with Secure Copy Protocol (SCP), which relies on SSH for its secure transport.

- Before enabling SCP, you must correctly configure SSH, authentication, and authorization on the switch.

- Because SCP relies on SSH for its secure transport, the router must have an Rivest, Shamir, and Adelman (RSA) key pair.

- SCP relies on SSH for security.

• SCP 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 must have appropriate authorization to use SCP.

• 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.

Related Topics

Restrictions for Configuring the Switch for SSH

The following are restrictions for configuring the Switch for secure shell.

• 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.

• This software release does not support IP Security (IPSec).

• When using SCP, you cannot enter the password into the **copy** command. You must enter the password when prompted.

Related Topics

Information about SSH

Secure Shell (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 and Switch Access

Secure Shell (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 functions the same in IPv6 as in IPv4. For IPv6, SSH supports IPv6 addresses and enables secure, encrypted connections with remote IPv6 nodes over an IPv6 transport.

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+

- RADIUS

- Local authentication and authorization

Related Topics

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 the SSH server is running on an active switch and the active switch fails, the new active switch uses the RSA key pair generated by the previous active switch.

- 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.

- 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.

Related Topics

Secure Copy Protocol Overview

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 router 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.

Secure Copy Protocol

The Secure Copy Protocol (SCP) feature provides a secure and authenticated method for copying switch configurations or switch image files. 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 switch can determine whether the user has the correct privilege level. To configure the Secure Copy feature, you should understand the SCP concepts.

Related Topics

How to Configure SSH

Setting Up the Switch to Run SSH

Follow these steps to set up your Switch to run SSH:

Before you begin

Configure user authentication for local or remote access. This step is required. For more information, see Related Topics below.

SUMMARY STEPS

1. **enable**

2. configure terminal

3. hostname *hostname*

4. ip domain-name *domain_name*

5. crypto key generate rsa

6. end

7. show running-config

8. copy running-config startup-config

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** Example: Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 3	**hostname** *hostname* Example: Switch(config)# **hostname your_hostname**	Configures a hostname and IP domain name for your Switch. **Note** Follow this procedure only if you are configuring the Switch as an SSH server.
Step 4	**ip domain-name** *domain_name* Example: Switch(config)# **ip domain-name your_domain**	Configures a host domain for your Switch.
Step 5	**crypto key generate rsa** Example: Switch(config)# **crypto key generate rsa**	Enables the SSH server for local and remote authentication on the Switch and generates an RSA key pair. Generating an RSA key pair for the Switch automatically enables SSH. 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. **Note** Follow this procedure only if you are configuring the Switch as an SSH server.
Step 6	**end** Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Switch(config)# **end**	
Step 7	show running-config **Example:** Switch# **show running-config**	Verifies your entries.
Step 8	copy running-config startup-config **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Related Topics

Configuring the SSH Server

Follow these steps to configure the SSH server:

> ✎
>
> **Note** This procedure is only required if you are configuring the Switch as an SSH server.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip ssh version** [**1** | **2**]
4. **ip ssh** {**time-out** *seconds* | **authentication-retries** *number*}
5. Use one or both of the following:
   - line vty*line_number*[*ending_line_number*]
   - **transport input ssh**
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example:	• Enter your password if prompted.
	Switch> **enable**	
Step 2	**configure terminal**	Enters global configuration mode.
	Example:	
	Switch# **configure terminal**	
Step 3	**ip ssh version** [**1** | **2**]	(Optional) Configures the Switch to run SSH Version 1 or SSH Version 2.
	Example:	• **1**—Configure the Switch to run SSH Version 1.
	Switch(config)# **ip ssh version 1**	• **2**—Configure 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.
Step 4	**ip ssh** {**time-out** *seconds* | **authentication-retries** *number*}	Configures the SSH control parameters:
	Example:	• **time-out** *seconds*: Specify 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.
	Switch(config)# **ip ssh time-out 90** OR Switch(config)# **ip ssh authentication-retries 2**	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.
		• **authentication-retries** *number*: Specify the number of times that a client can re-authenticate to the server. The default is 3; the range is 0 to 5.
		Repeat this step when configuring both parameters.
Step 5	Use one or both of the following:	(Optional) Configures the virtual terminal line settings.
	• line vty*line_number*[*ending_line_number*] • **transport input ssh**	• Enters line configuration mode to configure the virtual terminal line settings. For *line_number* and *ending_line_number*, specify a pair of lines. The range is 0 to 15.
	Example: Switch(config)# **line vty 1 10** or	• Specifies that the Switch prevent non-SSH Telnet connections. This limits the router to only SSH connections.

	Command or Action	Purpose
	Switch(config-line)# **transport input ssh**	
Step 6	**end** **Example:** Switch(config-line)# **end**	Returns to privileged EXEC mode.
Step 7	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 8	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Monitoring the SSH Configuration and Status

This table displays the SSH server configuration and status.

Table 83: Commands for Displaying the SSH Server Configuration and Status

Command	Purpose
show ip ssh	Shows the version and configuration information for the SSH server.
show ssh	Shows the status of the SSH server.

Additional References for Configuring Secure Shell

Related Documents

Related Topic	Document Title
Configuring Identity Control policies and Identity Service templates for Session Aware networking.	Session Aware Networking Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)
Configuring RADIUS, TACACS+, Secure Shell, 802.1X and AAA.	Securing User Services Configuration Guide Library, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for SSH

Release	Feature Information
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

CHAPTER 38

Configuring Secure Socket Layer HTTP

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information about Secure Sockets Layer (SSL) HTTP

Secure HTTP Servers and Clients Overview

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://.

> **Note** SSL evolved into Transport Layer Security (TLS) in 1999, but is still used in this particular context.

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, in turn, 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.

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 re-enable 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.

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
  02161743 45322D33 3535302D 31332E73 756D6D30 342D3335 3530301E 170D3933
  30333031 30303030 35395A17 0D323030 31303130 30303030 305A3059 312F302D
```

```
<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 re-enable a secure HTTP server, a new self-signed certificate is generated.

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.

For additional information on Certificate Authorities, see the "Configuring Certification Authority Interoperability" chapter in the *Cisco IOS Security Configuration Guide, Release 12.4*.

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_NULL_SHA key exchange with NULL for message encryption and SHA for message digest (only for SSL 3.0).

3. SSL_RSA_WITH_NULL_MD5 key exchange with NULL for message encryption and MD5 for message digest (only for SSL 3.0).

4. SSL_RSA_WITH_RC4_128_MD5—RSA key exchange with RC4 128-bit encryption and MD5 for message digest

5. SSL_RSA_WITH_RC4_128_SHA—RSA key exchange with RC4 128-bit encryption and SHA for message digest

6. SSL_RSA_WITH_3DES_EDE_CBC_SHA—RSA key exchange with 3DES and DES-EDE3-CBC for message encryption and SHA for message digest

7. SSL_RSA_WITH_AES_128_CBC_SHA—RSA key exchange with AES 128-bit encryption and SHA for message digest (only for SSL 3.0).

8. SSL_RSA_WITH_AES_256_CBC_SHA—RSA key exchange with AES 256-bit encryption and SHA for message digest (only for SSL 3.0).

9. SSL_RSA_WITH_DHE_AES_128_CBC_SHA—RSA key exchange with AES 128-bit encryption and SHA for message digest (only for SSL 3.0).

10. SSL_RSA_WITH_DHE_AES_256_CBC_SHA—RSA key exchange with AES 256-bit encryption and SHA for message digest (only for SSL 3.0).

Note The latest versions of Chrome do not support the four original cipher suites, thus disallowing access to both web GUI and guest portals.

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.

Default SSL Configuration

The standard HTTP server is enabled.

SSL is enabled.

No CA trustpoints are configured.

No self-signed certificates are generated.

SSL Configuration Guidelines

When SSL is used in a switch cluster, the SSL session terminates at the cluster commander. Cluster member switches must run standard HTTP.

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.

In a switch stack, the SSL session terminates at the active switch.

How to Configure Secure HTTP Servers and Clients

Configuring a CA Trustpoint

For secure HTTP connections, we recommend that you configure an official CA trustpoint. A CA trustpoint is more secure than a self-signed certificate.

Beginning in privileged EXEC mode, follow these steps to configure a CA Trustpoint:

SUMMARY STEPS

1. **configure terminal**
2. **hostname** *hostname*
3. **ip domain-name** *domain-name*
4. **crypto key generate rsa**
5. **crypto ca trustpoint** *name*
6. **enrollment url** *url*
7. **enrollment http-proxy** *host-name port-number*
8. **crl query** *url*
9. **primary** *name*
10. **exit**
11. **crypto ca authentication** *name*
12. **crypto ca enroll** *name*
13. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 2	**hostname** *hostname* Example: Switch(config)# **hostname your_hostname**	Specifies the hostname of the switch (required only if you have not previously configured a hostname). The hostname is required for security keys and certificates.
Step 3	**ip domain-name** *domain-name* Example: Switch(config)# **ip domain-name your_domain**	Specifies the IP domain name of the switch (required only if you have not previously configured an IP domain name). The domain name is required for security keys and certificates.
Step 4	**crypto key generate rsa** Example: Switch(config)# **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.
Step 5	**crypto ca trustpoint** *name* Example: Switch(config)# **crypto ca trustpoint your_trustpoint**	Specifies a local configuration name for the CA trustpoint and enter CA trustpoint configuration mode.

	Command or Action	**Purpose**
Step 6	**enrollment url** *url* **Example:** Switch(ca-trustpoint)# **enrollment url http://your_server:80**	Specifies the URL to which the switch should send certificate requests.
Step 7	**enrollment http-proxy** *host-name port-number* **Example:** Switch(ca-trustpoint)# **enrollment http-proxy your_host 49**	(Optional) Configures the switch to obtain certificates from the CA through an HTTP proxy server. • For *host-name* , specify the proxy server used to get the CA. • For *port-number*, specify the port number used to access the CA.
Step 8	**crl query** *url* **Example:** Switch(ca-trustpoint)# **crl query ldap://your_host:49**	Configures the switch to request a certificate revocation list (CRL) to ensure that the certificate of the peer has not been revoked.
Step 9	**primary** *name* **Example:** Switch(ca-trustpoint)# **primary your_trustpoint**	(Optional) Specifies that the trustpoint should be used as the primary (default) trustpoint for CA requests. • For *name*, specify the trustpoint that you just configured.
Step 10	**exit** **Example:** Switch(ca-trustpoint)# **exit**	Exits CA trustpoint configuration mode and return to global configuration mode.
Step 11	**crypto ca authentication** *name* **Example:** Switch(config)# **crypto ca authentication your_trustpoint**	Authenticates the CA by getting the public key of the CA. Use the same name used in Step 5.
Step 12	**crypto ca enroll** *name* **Example:** Switch(config)# **crypto ca enroll your_trustpoint**	Obtains the certificate from the specified CA trustpoint. This command requests a signed certificate for each RSA key pair.
Step 13	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Configuring the Secure HTTP Server

Beginning in privileged EXEC mode, follow these steps to configure a 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.

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
```

SUMMARY STEPS

1. **show ip http server status**
2. **configure terminal**
3. **ip http secure-server**
4. **ip http secure-port** *port-number*
5. **ip http secure-ciphersuite** {[**3des-ede-cbc-sha**] [**rc4-128-md5**] [**rc4-128-sha**] [**des-cbc-sha**]}
6. **ip http secure-client-auth**
7. **ip http secure-trustpoint** *name*
8. **ip http path** *path-name*
9. **ip http access-class** *access-list-number*
10. **ip http max-connections** *value*
11. **ip http timeout-policy idle** *seconds* **life** *seconds* **requests** *value*
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show ip http server status** **Example:** Switch# **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

	Command or Action	Purpose
		HTTP secure server capability: Not present
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**ip http secure-server** **Example:** Switch(config)# **ip http secure-server**	Enables the HTTPS server if it has been disabled. The HTTPS server is enabled by default.
Step 4	**ip http secure-port** *port-number* **Example:** Switch(config)# **ip http secure-port 443**	(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.
Step 5	**ip http secure-ciphersuite** {[**3des-ede-cbc-sha**] [**rc4-128-md5**] [**rc4-128-sha**] [**des-cbc-sha**]} **Example:** Switch(config)# **ip http secure-ciphersuite rc4-128-md5**	(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.
Step 6	**ip http secure-client-auth** **Example:** Switch(config)# **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.
Step 7	**ip http secure-trustpoint** *name* **Example:** Switch(config)# **ip http secure-trustpoint your_trustpoint**	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.
Step 8	**ip http path** *path-name* **Example:** Switch(config)# **ip http path /your_server:80**	(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).

	Command or Action	Purpose
Step 9	**ip http access-class** *access-list-number* **Example:** Switch(config)# **ip http access-class 2**	(Optional) Specifies an access list to use to allow access to the HTTP server.
Step 10	**ip http max-connections** *value* **Example:** Switch(config)# **ip http max-connections 4**	(Optional) Sets the maximum number of concurrent connections that are allowed to the HTTP server. We recommend that the value be at least 10 and not less. This is required for the UI to function as expected.
Step 11	**ip http timeout-policy idle** *seconds* **life** *seconds* **requests** *value* **Example:** Switch(config)# **ip http timeout-policy idle 120 life 240 requests 1**	(Optional) Specifies how long a connection to the HTTP server can remain open under the defined circumstances: • **idle**—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**—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**—the maximum number of requests processed on a persistent connection. The maximum value is 86400. The default is 1.
Step 12	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Configuring the Secure HTTP Client

Beginning in privileged EXEC mode, follow these steps to configure a 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.

SUMMARY STEPS

1. **configure terminal**
2. **ip http client secure-trustpoint** *name*

> **3.** **ip http client secure-ciphersuite** {[**3des-ede-cbc-sha**] [**rc4-128-md5**] [**rc4-128-sha**] [**des-cbc-sha**]}
> **4.** **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**ip http client secure-trustpoint** *name* **Example:** Switch(config)# **ip http client secure-trustpoint your_trustpoint**	(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.
Step 3	**ip http client secure-ciphersuite** {[**3des-ede-cbc-sha**] [**rc4-128-md5**] [**rc4-128-sha**] [**des-cbc-sha**]} **Example:** Switch(config)# **ip http client secure-ciphersuite rc4-128-md5**	(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.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Monitoring Secure HTTP Server and Client Status

To monitor the SSL secure server and client status, use the privileged EXEC commands in the following table.

Table 84: Commands for Displaying the SSL Secure Server and Client Status

Command	**Purpose**
show ip http client secure status	Shows the HTTP secure client configuration.
show ip http server secure status	Shows the HTTP secure server configuration.
show running-config	Shows the generated self-signed certificate for secure HTTP connections.

Additional References for Configuring Secure Shell

Related Documents

Related Topic	Document Title
Configuring Identity Control policies and Identity Service templates for Session Aware networking.	Session Aware Networking Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)
Configuring RADIUS, TACACS+, Secure Shell, 802.1X and AAA.	Securing User Services Configuration Guide Library, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for SSL HTTP

Release	Feature Information
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Configuring IPv4 ACLs

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Restrictions for Configuring Network Security with ACLs

General Network Security

The following are restrictions for configuring network security with 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. VLAN maps also accept a name.

- A standard ACL and an extended ACL cannot have the same name.

- 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.

IPv4 ACL Network Interfaces

The following restrictions apply to IPv4 ACLs to network interfaces:

- When controlling access to an interface, you can use a named or numbered ACL.

- 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 or a VLAN map applied to the VLAN.

- If you apply an ACL to a Layer 3 interface and routing is not enabled on the switch, 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.

Note By default, the router sends Internet Control Message Protocol (ICMP) unreachable messages when a packet is denied by an access group on a Layer 3 interface. 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. They do not generate ICMP unreachable messages. ICMP unreachable messages can be disabled on router ACLs with the **no ip unreachables** interface command.

MAC ACLs on 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:

- 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.

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.

Information about Network Security with ACLs

This chapter describes how to configure network security on the switch by using access control lists (ACLs), which in commands and tables are also referred to as access lists.

Cisco TrustSec and ACLs

Catalyst 3850 switches running the IP base or IP services feature set also support Cisco TrustSec Security Group Tag (SCT) Exchange Protocol (SXP). This feature supports security group access control lists (SGACLs), which define ACL policies for a group of devices instead of an IP address. The SXP control protocol allows tagging packets with SCTs without a hardware upgrade, and runs between access layer devices at the Cisco

TrustSec domain edge and distribution layer devices within the Cisco TrustSec domain. Catalyst 3850 switches operate as access layer switches in the Cisco TrustSec network.

The sections on SXP define the capabilities supported on the Catalyst 3850 switches.

ACL Overview

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.

Access Control Entries

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.

ACL Supported Types

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.

Supported ACLs

The switch supports three types of ACLs to filter traffic:

- Port ACLs access-control traffic entering a Layer 2 interface. You can apply only one IP access list and one MAC access list to a Layer 2 interface.

- Router ACLs access-control routed traffic between VLANs and are applied to Layer 3 interfaces in a specific direction (inbound or outbound).

- VLAN ACLs or VLAN maps access-control all packets (bridged and routed). You can use VLAN maps to filter traffic between devices in the same VLAN. VLAN maps are configured to provide access control based on Layer 3 addresses for IPv4. Unsupported protocols are access-controlled through MAC addresses

using Ethernet ACEs. After a VLAN map is applied to a VLAN, all packets (routed or bridged) entering the VLAN are checked against the VLAN map. Packets can either enter the VLAN through a switch port or through a routed port after being routed.

ACL Precedence

When VLAN maps, Port ACLs, and router ACLs are configured on the same switch, the filtering precedence, from greatest to least, is port ACL, router ACL, then VLAN map. The following examples describe simple use cases:

- When both an input port ACL and a VLAN map are applied, incoming packets received on ports with a port ACL applied are filtered by the port ACL. Other packets are filtered by the VLAN map

- When an input router ACL and input port ACL exist in a switch virtual interface (SVI), incoming packets received on ports to which a port ACL is applied are filtered by the port ACL. Incoming 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, incoming packets received on the ports to which a port ACL is applied are filtered by the port ACL. Outgoing routed IP packets are filtered by the router ACL. Other packets are not filtered.

- When a VLAN map, input router ACL, and input port ACL exist in an SVI, incoming packets received on the ports to which a port ACL is applied are only filtered by the port ACL. Incoming routed IP packets received on other ports are filtered by both the VLAN map and the router ACL. Other packets are filtered only by the VLAN map.

- When a VLAN map, output router ACL, and input port ACL exist in an SVI, incoming packets received on the ports to which a port ACL is applied are only filtered by the port ACL. Outgoing routed IP packets are filtered by both the VLAN map and the router ACL. Other packets are filtered only by the VLAN map.

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. Port ACLs can be applied to the interface in outbound and inbound direction. The following 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 on an 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 73: Using ACLs to Control Traffic in a Network

This 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
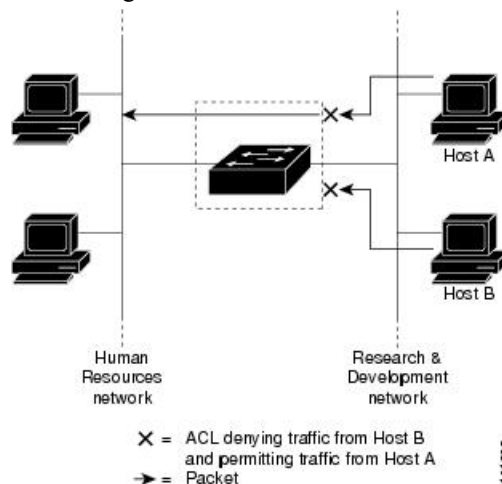
inbound direction.

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.

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.

The switch supports these 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 type information for matching operations.

As with port ACLs, the switch examines ACLs associated with features configured on a given interface. 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.

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 or switch stack.

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 74: Using VLAN Maps to Control Traffic

This 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
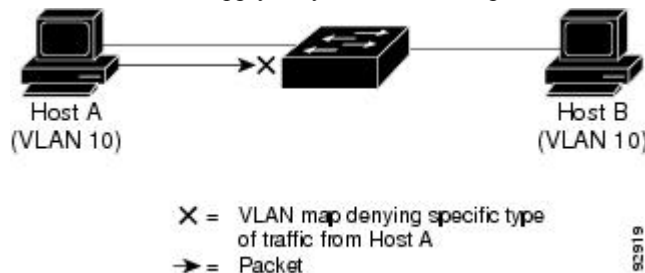
VLAN.

ACEs and 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 access control entries (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.

Example: ACEs and Fragmented and Unfragmented Traffic

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.

ACLs and Switch Stacks

ACL support is the same for a switch stack as for a standalone switch. ACL configuration information is propagated to all switches in the stack. All switches in the stack, including the active switch, process the information and program their hardware.

Active Switch and ACL Functions

The active switch performs these ACL functions:

- It processes the ACL configuration and propagates the information to all stack members.

- It distributes the ACL information to any switch that joins the stack.

- If packets must be forwarded by software for any reason (for example, not enough hardware resources), the active switch forwards the packets only after applying ACLs on the packets.

- It programs its hardware with the ACL information it processes.

Stack Member and ACL Functions

Stack members perform these ACL functions:

- They receive the ACL information from the active switch and program their hardware.

- A stack member configured as a standby switch, performs the functions of the active switch in the event the active switch fails.

Active Switch Failure and ACLs

Both the active and standby switches have the ACL information. When the active switch fails, the standby takes over. The new active switch distributes the ACL information to all stack members.

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.

IPv4 ACL Switch Unsupported Features

Configuring IPv4 ACLs on the switch is the same as configuring IPv4 ACLs on other Cisco switches and routers.

The switch does not support these Cisco IOS router ACL-related features:

- Non-IP protocol ACLs

- IP accounting

- Reflexive ACLs and dynamic ACLs are not supported.

Access List Numbers

The number you use to denote your ACL shows the type of access list that you are creating.

This 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 85: Access List Numbers

Access List Number	Type	Supported
1–99	IP standard access list	Yes
100–199	IP extended access list	Yes

Access List Number	Type	Supported
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

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.

Numbered Standard IPv4 ACLs

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.

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 VLANs, to terminal lines, or to interfaces.

Numbered Extended IPv4 ACLs

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.

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.

Some protocols also have specific parameters and keywords that apply to that protocol.

You can define an extended TCP, UDP, ICMP, IGMP, or other IP ACL. The switch also supports these IP protocols:

> **Note**   ICMP echo-reply cannot be filtered. All other ICMP codes or types can be filtered.

These IP protocols are supported:

- Authentication Header Protocol (**ahp**)

- Encapsulation Security Payload (**esp**)

- Enhanced Interior Gateway Routing Protocol (**eigrp**)

- 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**)

Named IPv4 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 and . The advantage of using named ACLs instead of numbered lists is that you can delete individual entries from a named list.

Consider these guidelines before configuring named ACLs:

- Numbered ACLs are also available.

- A standard ACL and an extended ACL cannot have the same name.

- You can use standard or extended ACLs (named or numbered) in VLAN maps.

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.

Smart Logging

When smart logging is enabled on the switch and an ACL configured with smart logging is attached to a Layer 2 interface (port ACL), the contents of packets denied or permitted because of the ACL are also sent to a specified NetFlow collector.

Hardware and Software Treatment of IP ACLs

ACL processing is performed in hardware. If the hardware reaches its capacity to store ACL configurations, all packets on that interface are dropped.

Note If an ACL configuration cannot be implemented in hardware due to an out-of-resource condition on a switch or stack member, then only the traffic in that VLAN arriving on that switch is 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 privileged EXEC command to obtain some basic hardware ACL statistics for switched and routed packets.

VLAN Map Configuration Guidelines

VLAN maps are 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.

The following are the VLAN map configuration guidelines:

- 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.

- 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 a VLAN map configuration cannot be applied in hardware, all packets in that VLAN are dropped.

VLAN Maps with Router ACLs

To access control both bridged and routed traffic, you can use VLAN maps only or a combination of router ACLs and VLAN maps. You can define router ACLs on both input and output routed VLAN interfaces, and you can define a VLAN map to access control the bridged traffic.

If a packet flow matches a VLAN-map deny clause in the ACL, regardless of the router ACL configuration, the packet flow is denied.

Note When you use router ACLs with VLAN maps, packets that require logging on the router ACLs are not logged if they are denied by a VLAN map.

If the VLAN map has a match clause for the type of packet (IP or MAC) and the packet does not match the type, the default is to drop the packet. If there is no match clause in the VLAN map, and no action specified, the packet is forwarded if it does not match any VLAN map entry.

VLAN Maps and Router ACL Configuration Guidelines

These guidelines are for configurations where you need to have an router ACL and a VLAN map on the same VLAN. These guidelines do not apply to configurations where you are mapping router ACLs and VLAN maps on different VLANs.

If you must configure a router ACL and a VLAN map on the same VLAN, use these guidelines for both router ACL and VLAN map configuration:

- You can configure only one VLAN map and one router ACL in each direction (input/output) on a VLAN interface.

- Whenever possible, try to write the ACL with all entries having a single action except for the final, default action of the other type. That is, write the ACL using one of these two forms:

  permit... permit... permit... deny ip any any

  or

  deny... deny... deny... permit ip any any

- To define multiple actions in an ACL (permit, deny), group each action type together to reduce the number of entries.

- Avoid including Layer 4 information in an ACL; adding this information complicates the merging process. The best merge results are obtained if the ACLs are filtered based on IP addresses (source and destination) and not on the full flow (source IP address, destination IP address, protocol, and protocol ports). It is also helpful to use *don't care* bits in the IP address, whenever possible.

  If you need to specify the full-flow mode and the ACL contains both IP ACEs and TCP/UDP/ICMP ACEs with Layer 4 information, put the Layer 4 ACEs at the end of the list. This gives priority to the filtering of traffic based on IP addresses.

Time Ranges for 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. The **time-range** keyword and argument are referenced in the named and numbered extended ACL task tables.

These are some 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 hardware memory. 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.

IPv4 ACL Interface Considerations

When you apply the **ip access-group** interface configuration command to a Layer 3 interface (an SVI, a Layer 3 EtherChannel, or a routed port), the interface must have been configured with an IP address. Layer 3 access

groups filter packets that are routed or are received by Layer 3 processes on the CPU. They do not affect packets bridged within a VLAN.

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 routing 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.

How to Configure ACLs

Configuring IPv4 ACLs

These are the steps to use IP ACLs on the switch:

SUMMARY STEPS

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. You can also apply standard and extended IP ACLs to VLAN maps.

DETAILED STEPS

	Command or Action	Purpose
Step 1	Create an ACL by specifying an access list number or name and the access conditions.	
Step 2	Apply the ACL to interfaces or terminal lines. You can also apply standard and extended IP ACLs to VLAN maps.	

Creating a Numbered Standard ACL

Follow these steps to create a numbered standard ACL:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **access-list** *access-list-number* {**deny** | **permit**} *source source-wildcard*]

> 4. **end**
> 5. **show running-config**
> 6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**access-list** *access-list-number* {**deny**	**permit**} *source source-wildcard*] **Example:** Switch(config)# **access-list 2 deny your_host**	Defines 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. **Note** Logging is supported only on ACLs attached to Layer 3 interfaces.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Creating a Numbered Extended ACL

Follow these steps to create a numbered extended ACL:

SUMMARY STEPS

1. **configure terminal**
2. **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*]
3. **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*]
4. **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*]
5. **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*]
6. **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*]
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**access-list** *access-list-number* {**deny**	**permit**} *protocol source source-wildcard destination destination-wildcard*	Defines an extended IPv4 access list and the access conditions.

Command or Action	Purpose
[**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log** [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*]	The *access-list-number* is a decimal number from 100 to 199 or 2000 to 2699.
Example: Switch(config)# **access-list 101 permit ip host 10.1.1.2 any precedence 0 tos 0 log**	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 P 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 the following steps.
	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.
	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

	Command or Action	**Purpose**
		the entry or **log-input** to include the input interface in the log entry.
		• **time-range**—Specify the time-range name.
		• **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.
		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**.
Step 3	**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*] **Example:** `Switch(config)# access-list 101 permit tcp any any eq 500`	Defines an extended TCP access list and the access conditions. The parameters are the same as those described for an extended IPv4 ACL, 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. 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 4	**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*] **Example:** `Switch(config)# access-list 101 permit udp any any eq 100`	(Optional) Defines an extended UDP access list and the access conditions. 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** keywords are not valid for UDP.

	Command or Action	**Purpose**
Step 5	**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*]	Defines an extended ICMP access list and the access conditions.
		The ICMP parameters are the same as those described for most IP protocols in an extended IPv4 ACL, with the addition of the ICMP message type and code parameters. These optional keywords have these meanings:
	Example:	*icmp-type*—Enter to filter by ICMP message type, a number from 0 to 255.
	`Switch(config)# access-list 101 permit icmp any any 200`	*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.
Step 6	**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) Defines an extended IGMP access list and the access conditions.
		The IGMP parameters are the same as those described for most IP protocols in an extended IPv4 ACL, with this optional parameter.
	Example:	*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**.
	`Switch(config)# access-list 101 permit igmp any any 14`	
Step 7	**end**	Returns to privileged EXEC mode.
	Example:	
	`Switch(config)# end`	

Creating Named Standard ACLs

Follow these steps to create a standard ACL using names:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip access-list standard** *name*
4. Use one of the following:

   - **deny** {*source* [*source-wildcard*] \| **host** *source* \| **any**} [**log**]
   - **permit** {*source* [*source-wildcard*] \| **host** *source* \| **any**} [**log**]

5. **end**

6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**ip access-list standard** *name* **Example:** Switch(config)# **ip access-list standard 20**	Defines a standard IPv4 access list using a name, and enter access-list configuration mode. The name can be a number from 1 to 99.
Step 4	Use one of the following: • **deny** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**] • **permit** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**] **Example:** Switch(config-std-nacl)# **deny 192.168.0.0 0.0.255.255 255.255.0.0 0.0.255.255** or Switch(config-std-nacl)# **permit 10.108.0.0 0.0.0.0 255.255.255.0 0.0.0.0**	In access-list configuration mode, specify one or more conditions denied or permitted to decide if the packet is forwarded or dropped. • **host** *source*—A source and source wildcard of *source* 0.0.0.0. • **any**—A source and source wildcard of 0.0.0.0 255.255.255.255.
Step 5	**end** **Example:** Switch(config-std-nacl)# **end**	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:**	Verifies your entries.

	Command or Action	Purpose
	Switch# **show running-config**	
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Creating Extended Named ACLs

Follow these steps to create an extended ACL using names:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip access-list extended** *name*
4. {**deny** | **permit**} *protocol* {*source* [*source-wildcard*] | **host** *source* | **any**} {*destination* [*destination-wildcard*] | host *destination* | **any**} [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log**] [**time-range** *time-range-name*]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**ip access-list extended** *name* **Example:** Switch(config)# **ip access-list extended 150**	Defines an extended IPv4 access list using a name, and enter access-list configuration mode. The name can be a number from 100 to 199.

	Command or Action	Purpose
Step 4	{**deny** | **permit**} *protocol* {*source* [*source-wildcard*] | **host** *source* | **any**} {*destination* [*destination-wildcard*] | host *destination* | **any**} [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log**] [**time-range** *time-range-name*] **Example:** Switch(config-ext-nacl)# **permit 0 any any**	In access-list configuration mode, specify the conditions allowed or denied. Use the **log** keyword to get access list logging messages, including violations. • **host** *source*—A source and source wildcard of *source* 0.0.0.0. • **host** *destintation*—A destination and destination wildcard of *destination* 0.0.0.0. • **any**—A source and source wildcard or destination and destination wildcard of 0.0.0.0 255.255.255.255.
Step 5	**end** **Example:** Switch(config-ext-nacl)# **end**	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

When you are creating 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.

Being able to selectively remove lines from a named ACL is one reason you might use named ACLs instead of numbered ACLs.

What to do next

After creating a named ACL, you can apply it to interfaces or to VLANs .

Configuring Time Ranges for ACLs

Follow these steps to configure a time-range parameter for an ACL:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **time-range** *time-range-name*
4. Use one of the following:

    - **absolute** [**start** *time date*] [**end** *time date*]
    - **periodic** *day-of-the-week hh:mm to* [*day-of-the-week*] *hh:mm*
    - **periodic** {**weekdays** | **weekend** | **daily**} *hh:mm to hh:mm*

5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch(config)# **enable**	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**time-range** *time-range-name* **Example:** Switch(config)# **time-range workhours**	Assigns a meaningful name (for example, *workhours*) to the time range to be created, and enter time-range configuration mode. The name cannot contain a space or quotation mark and must begin with a letter.
Step 4	Use one of the following: - **absolute** [**start** *time date*] [**end** *time date*] - **periodic** *day-of-the-week hh:mm to* [*day-of-the-week*] *hh:mm* - **periodic** {**weekdays**	**weekend**	**daily**} *hh:mm to hh:mm* **Example:** Switch(config-time-range)# **absolute start 00:00 1 Jan 2006 end 23:59 1 Jan 2006** or	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.

	Command or Action	**Purpose**
	Switch(config-time-range)# **periodic weekdays 8:00 to 12:00**	
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

What to do next

Repeat the steps if you have multiple items that you want in effect at different times.

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.

Follow these steps to restrict incoming and outgoing connections between a virtual terminal line and the addresses in an ACL:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **line** [**console** | **vty**] *line-number*
4. **access-class** *access-list-number* {**in** | **out**}
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch(config)# **enable**	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**line** [**console** | **vty**] *line-number* **Example:** Switch(config)# **line console 0**	Identifies a specific line to configure, and enter 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.
Step 4	**access-class** *access-list-number* {**in** | **out**} **Example:** Switch(config-line)# **access-class 10 in**	Restricts incoming and outgoing connections between a particular virtual terminal line (into a device) and the addresses in an access list.
Step 5	**end** **Example:** Switch(config-line)# **end**	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Applying an IPv4 ACL to an Interface

This section describes how to apply IPv4 ACLs to network interfaces.

Beginning in privileged EXEC mode, follow these steps to control access to an interface:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip access-group** {*access-list-number* | *name*} {**in** | **out**}
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	Identifies a specific interface for configuration, and enter interface configuration mode. The interface can be a Layer 2 interface (port ACL), or a Layer 3 interface (router ACL).
Step 3	**ip access-group** {*access-list-number*	*name*} {**in**	**out**} **Example:** Switch(config-if)# **ip access-group 2 in**	Controls access to the specified interface. The **out** keyword is not supported for Layer 2 interfaces (port ACLs).
Step 4	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** Switch# **show running-config**	Displays the access list configuration.
Step 6	**copy running-config startup-config** **Example:**	(Optional) Saves your entries in the configuration file.

Command or Action	Purpose
Switch# **copy running-config startup-config**	

Creating 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.

Follow these steps to create a named MAC extended ACL:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mac access-list extended** *name*
4. {**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*]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** Example: Switch> **enable**	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 3	**mac access-list extended** *name* Example: Switch(config)# **mac access-list extended mac1**	Defines an extended MAC access list using a name.
Step 4	{**deny**	**permit**} {**any**	**host** *source MAC address*	*source MAC address mask*} {**any**	**host** *destination MAC address*	*destination MAC address mask*} [*type mask*	**lsap** *lsap*	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

	Command or Action	Purpose
	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*] **Example:** Switch(config-ext-macl)# **deny any any decnet-iv** or Switch(config-ext-macl)# **permit any any**	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*—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*—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**—A non-IP protocol. • **cos** *cos*—An IEEE 802.1Q cost of service number from 0 to 7 used to set priority.
Step 5	**end** **Example:** Switch(config-ext-macl)# **end**	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Applying a MAC ACL to a Layer 2 Interface

Follow these steps to apply a MAC access list to control access to a Layer 2 interface:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*

4. **mac access-group** {*name*} {**in** | **out** }
5. **end**
6. **show mac access-group** [**interface** *interface-id*]
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/2**	Identifies a specific interface, and enter interface configuration mode. The interface must be a physical Layer 2 interface (port ACL).
Step 4	**mac access-group** {*name*} {**in**	**out** } **Example:** Switch(config-if)# **mac access-group mac1 in**	Controls access to the specified interface by using the MAC access list. Port ACLs are supported in the outbound and inbound directions .
Step 5	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 6	**show mac access-group** [**interface** *interface-id*] **Example:** Switch# **show mac access-group interface gigabitethernet1/0/2**	Displays the MAC access list applied to the interface or all Layer 2 interfaces.
Step 7	**show running-config** **Example:**	Verifies your entries.

	Command or Action	Purpose
	Switch# **show running-config**	
Step 8	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

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

To create a VLAN map and apply it to one or more VLANs, perform these steps:

Before you begin

Create the standard or extended IPv4 ACLs or named MAC extended ACLs that you want to apply to the VLAN.

SUMMARY STEPS

1. **vlan access-map** *name* [**number**]
2. **match** {**ip** | **mac**} **address** {*name* | *number*} [*name* | *number*]
3. Enter one of the following commands 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):

   - **action** { **forward**}

     Switch(config-access-map)# **action forward**

   - **action** { **drop**}

     Switch(config-access-map)# **action drop**

4. **vlan filter** *mapname* **vlan-list** *list*

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**vlan access-map** *name* [**number**] **Example:** Switch(config)# **vlan access-map map_1 20**	Creates 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. 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. Entering this command changes to access-map configuration mode.
Step 2	**match** {**ip** | **mac**} **address** {*name* | *number*} [*name* | *number*] **Example:** Switch(config-access-map)# **match ip address ip2**	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. **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.

Step 3	Enter one of the following commands 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):
	
	• **action** { **forward**}
	
	  Switch(config-access-map)# **action forward**
	
	• **action** { **drop**}
	
	  Switch(config-access-map)# **action drop**
		Sets the action for the map entry.

Step 4	**vlan filter** *mapname* **vlan-list** *list* **Example:**	Applies the VLAN map to one or more VLAN IDs.

Command or Action	Purpose
Switch(config)# **vlan filter map 1 vlan-list 20-22**	The list can be a single VLAN ID (22), a consecutive list (10-22), or a string of VLAN IDs (12, 22, 30). Spaces around the comma and hyphen are optional.

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:

SUMMARY STEPS

1. **configure terminal**
2. **vlan access-map** *name* [**number**]
3. **match** {**ip** | **mac**} **address** {*name* | *number*} [*name* | *number*]
4. **action** {**drop** | **forward**}
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**vlan access-map** *name* [**number**] **Example:** Switch(config)# **vlan access-map map_1 20**	Creates 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. 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. Entering this command changes to access-map configuration mode.
Step 3	**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

	Command or Action	Purpose
	Example: Switch(config-access-map)# **match ip address ip2**	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 4	**action** {**drop** | **forward**} **Example:** Switch(config-access-map)# **action forward**	(Optional) Sets the action for the map entry. The default is to forward.
Step 5	**end** **Example:** Switch(config-access-map)# **end**	Returns to global configuration mode.
Step 6	**show running-config** **Example:** Switch# **show running-config**	Displays the access list configuration.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

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:

SUMMARY STEPS

1. **configure terminal**
2. **vlan filter** *mapname* **vlan-list** *list*
3. **end**
4. **show running-config**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:**	Enters global configuration mode.

	Command or Action	**Purpose**
	Switch# **configure terminal**	
Step 2	**vlan filter** *mapname* **vlan-list** *list*	Applies the VLAN map to one or more VLAN IDs.
	Example:	The list can be a single VLAN ID (22), a consecutive list (10-22), or a string of VLAN IDs (12, 22, 30). Spaces
	Switch(config)# **vlan filter map 1 vlan-list 20-22**	around the comma and hyphen are optional.
Step 3	**end**	Returns to privileged EXEC mode.
	Example:	
	Switch(config)# **end**	
Step 4	**show running-config**	Displays the access list configuration.
	Example:	
	Switch# **show running-config**	
Step 5	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.
	Example:	
	Switch# **copy running-config startup-config**	

Configuring VACL Logging

Beginning in privileged EXEC mode:

SUMMARY STEPS

1. **configure terminal**
2. **vlan access-map name** [*number*]
3. **action drop log**
4. **exit**
5. **vlan access-log** {**maxflow** *max_number* | **threshold** *pkt_count*}
6. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal**	Enters global configuration mode.
	Example:	

	Command or Action	Purpose
	Switch# **configure terminal**	
Step 2	**vlan access-map name** [*number*] **Example:** Switch(config)# **vlan access-map gandymede 10**	Creates a VLAN map. Give it a name and optionally a number. The number is the sequence number of the entry within the map. The sequence number range is from 0 to 65535. 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. Specifying the map name and optionally a number enters the access-map configuration mode.
Step 3	**action drop log** **Example:** Switch(config-access-map)# **action drop log**	Sets the VLAN access map to drop and log IP packets.
Step 4	**exit** **Example:** Switch(config-access-map)# **exit**	Exits the VLAN access map configuration mode and return to the global configuration mode.
Step 5	**vlan access-log** {**maxflow** *max_number* | **threshold** *pkt_count*} **Example:** Switch(config)# **vlan access-log threshold 4000**	Configures the VACL logging parameters. • **maxflow** *max_number*—Sets the log table size. The content of the log table can be deleted by setting the **maxflow** to 0. When the log table is full, the software drops logged packets from new flows. The range is from 0 to 2048. The default is 500. • **threshold** *pkt_count*—Sets the logging threshold. A logging message is generated if the threshold for a flow is reached before the 5-minute interval. The threshold range is from 0 to 2147483647. The default threshold is 0, which means that a syslog message is generated every 5 minutes.
Step 6	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Monitoring IPv4 ACLs

You can monitor IPv4 ACLs by displaying the ACLs that are configured on the switch, and displaying the ACLs that have been applied to interfaces and VLANs.

When you use the **ip access-group** interface configuration command to apply ACLs to a Layer 2 or 3 interface, you can display the access groups on the interface. You can also display the MAC ACLs applied to a Layer 2 interface. You can use the privileged EXEC commands as described in this table to display this information.

Table 86: Commands for Displaying Access Lists and Access Groups

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-id*]	Displays MAC access lists applied to all Layer 2 interfaces or the specified Layer 2 interface.

You can also monitor VLAN maps by displaying information about VLAN access maps or VLAN filters. Use the privileged EXEC commands in this table to display VLAN map information.

Table 87: Commands for Displaying VLAN Map Information

Command	Purpose
show vlan access-map [*mapname*]	Displays information about all VLAN access maps or the specified access map.
show vlan filter [**access-map** *name*	**vlan** *vlan-id*]	Displays information about all VLAN filters or about a specified VLAN or VLAN access map.

Configuration Examples for ACLs

Examples: Using Time Ranges with ACLs

This example shows how to verify after you configure time ranges for *workhours* and to configure January 1, 2006, as a company holiday.

```
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)
```

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)
```

Examples: Including 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.

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
```

Examples: Troubleshooting ACLs

If this ACL manager message appears and [chars] is the access-list name,

```
ACLMGR-2-NOVMR: Cannot generate hardware representation of access list [chars]
```

The switch has insufficient resources to create a hardware representation of the ACL. 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.

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 hardware memory.

IPv4 ACL Configuration Examples

This section provides examples of configuring and applying IPv4 ACLs. For detailed information about compiling ACLs, see the *Cisco IOS Security Configuration Guide, Release 12.4* and to the Configuring IP Services" section in the "IP Addressing and Services" chapter of the *Cisco IOS IP Configuration Guide, Release 12.4*.

ACLs in a Small Networked Office

Figure 75: Using Router ACLs to Control Traffic

This 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
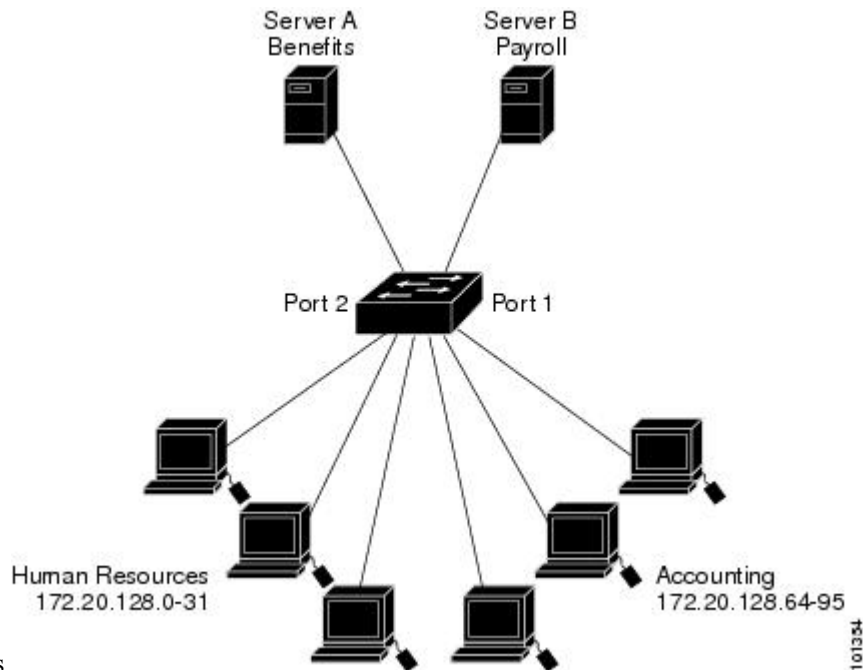
access.

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.

Examples: ACLs in a Small Networked Office

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# how access-lists
Standard IP access list 6
    10 permit 172.20.128.64, wildcard bits 0.0.0.31
Switch(config)# interface gigabitethernet1/0/1
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
```

```
   10 permit ip any 172.20.128.64 0.0.0.31
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# ip access-group 106 in
```

Example: Numbered ACLs

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 gigabitethernet2/0/1
Switch(config-if)# ip access-group 2 in
```

Examples: Extended ACLs

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 gigabitethernet2/0/1
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.

```
Switch(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 eq 23
Switch(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 eq 25
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# ip access-group 102 in
```

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 on stack member 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 gigabitethernet1/0/1
Switch(config-if)# ip access-group 102 in
```

Examples: Named ACLs

Creating named standard and extended ACLs

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 gigabitethernet3/0/2
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
```

Deleting individual ACEs from named ACLs

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
```

Examples: Time Range Applied to an IP ACL

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 gigabitethernet2/0/1
Switch(config-if)# ip access-group strict in
```

Examples: Commented IP ACL Entries

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
```

Examples: ACL Logging

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 gigabitethernet1/0/1
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 gigabitethernet1/0/2
Switch(config-if)# ip access-group ext1 in
```

This is a an example of a log for an extended ACL:

```
01:24:23:%SEC-6-IPACCESSLOGDP:list ext1 permitted icmp 10.1.1.15 -> 10.1.1.61 (0/0), 1
packet
01:25:14:%SEC-6-IPACCESSLOGDP:list ext1 permitted icmp 10.1.1.15 -> 10.1.1.61 (0/0), 7
packets
01:26:12:%SEC-6-IPACCESSLOGP:list ext1 denied udp 0.0.0.0(0) -> 255.255.255.255(0), 1 packet
01:31:33:%SEC-6-IPACCESSLOGP:list ext1 denied udp 0.0.0.0(0) -> 255.255.255.255(0), 8 packets
```

Note that all logging entries for IP ACLs start with %SEC-6-IPACCESSLOG with minor variations in format depending on the kind of ACL and the access entry that has been matched.

This is an example of an output message when the **log-input** keyword is entered:

```
00:04:21:%SEC-6-IPACCESSLOGDP:list inputlog permitted icmp 10.1.1.10 (Vlan1 0001.42ef.a400)
 ->
10.1.1.61 (0/0), 1 packet
```

A log message for the same sort of packet using the **log** keyword does not include the input interface information:

```
00:05:47:%SEC-6-IPACCESSLOGDP:list inputlog permitted icmp 10.1.1.10 -> 10.1.1.61 (0/0), 1
 packet
```

Configuration Examples for ACLs and VLAN Maps

Example: Creating an ACL and a VLAN Map to Deny a Packet

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 *ip1* ACL 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
```

Example: Creating an ACL and a VLAN Map to Permit a Packet

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: Default Action of Dropping IP Packets and Forwarding MAC Packets

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-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: Default Action of Dropping MAC Packets and Forwarding IP Packets

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

Example: Default Action of Dropping All Packets

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
```

Configuration Examples for Using VLAN Maps in Your Network

Example: Wiring Closet Configuration

Figure 76: 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. 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,
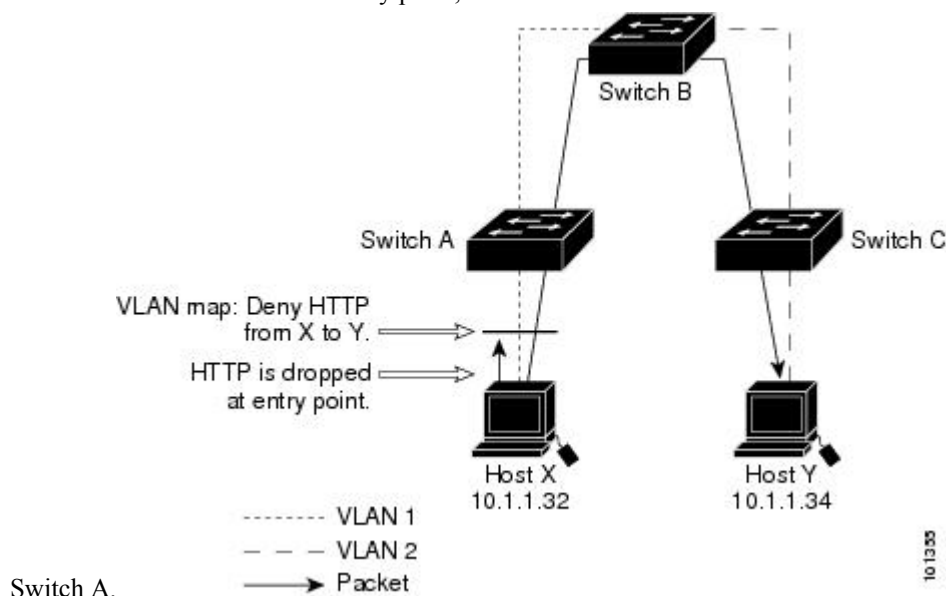
Switch A.

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
```

Example: Restricting Access to a Server on Another VLAN

Figure 77: Restricting Access to a Server on Another VLAN

You can restrict access to a server on another VLAN. For example, server 10.1.1.100 in VLAN 10 needs to have access denied to these hosts:

- Hosts in subnet 10.1.2.0/8 in VLAN 20 should not have access.

- Hosts 10.1.1.4 and 10.1.1.8 in VLAN 10 should not have access.
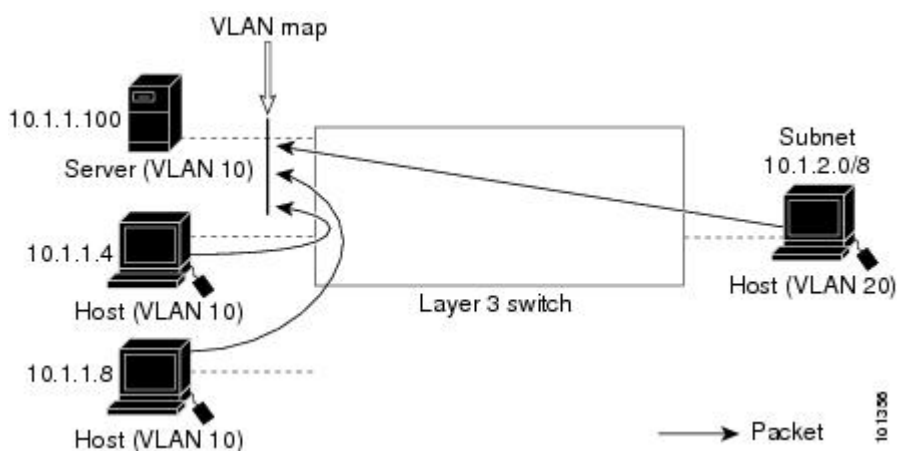
Example: Denying Access to a Server on Another VLAN

This example shows how to deny access to a server on another VLAN by creating the VLAN map SERVER 1 that denies access to hosts in subnet 10.1.2.0.8, host 10.1.1.4, and host 10.1.1.8 and permits other IP traffic. The final step is to apply the map SERVER1 to VLAN 10.

Define the IP ACL that will match the correct packets.

```
Switch(config)# ip access-list extended SERVER1_ACL
Switch(config-ext-nacl))# permit ip 10.1.2.0 0.0.0.255 host 10.1.1.100
Switch(config-ext-nacl))# permit ip host 10.1.1.4 host 10.1.1.100
Switch(config-ext-nacl))# permit ip host 10.1.1.8 host 10.1.1.100
Switch(config-ext-nacl))# exit
```

Define a VLAN map using this ACL that will drop IP packets that match SERVER1_ACL and forward IP packets that do not match the ACL.

```
Switch(config)# vlan access-map SERVER1_MAP
Switch(config-access-map)# match ip address SERVER1_ACL
Switch(config-access-map)# action drop
Switch(config)# vlan access-map SERVER1_MAP 20
```

```
Switch(config-access-map)# action forward
Switch(config-access-map)# exit
```

Apply the VLAN map to VLAN 10.

```
Switch(config)# vlan filter SERVER1_MAP vlan-list 10
```

Configuration Examples of Router ACLs and VLAN Maps Applied to VLANs

This section gives examples of applying router ACLs and VLAN maps to a VLAN for switched, bridged, routed, and multicast packets. Although the following illustrations show packets being forwarded to their destination, each time the packet's path crosses a line indicating a VLAN map or an ACL, it is also possible that the packet might be dropped, rather than forwarded.

Example: ACLs and Switched Packets

Figure 78: Applying ACLs on Switched Packets

This example shows how an ACL is applied on packets that are switched within a VLAN. Packets switched within the VLAN without being routed or forwarded by fallback bridging are only subject to the VLAN map of the input VLAN.
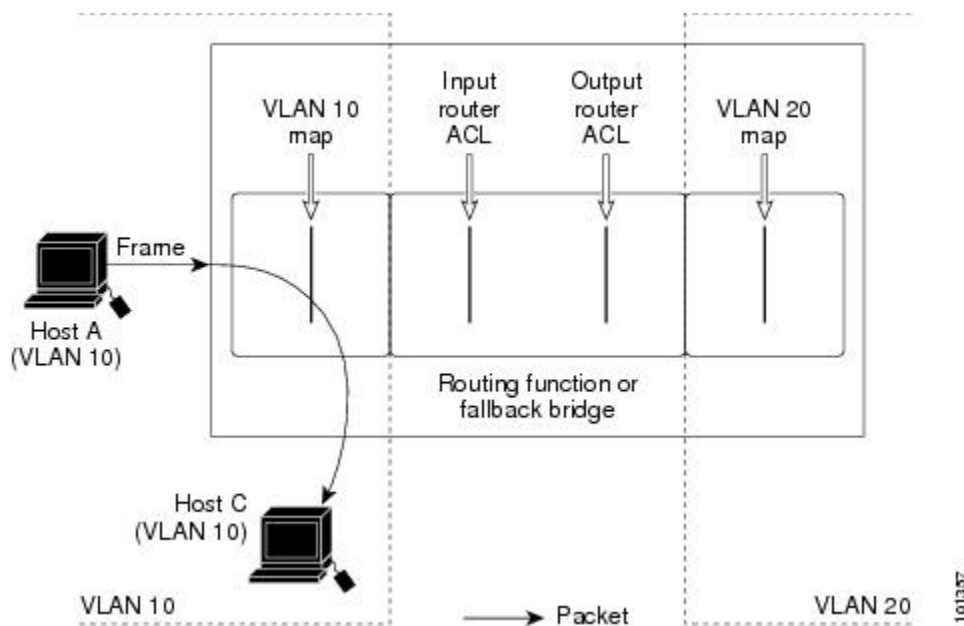
Example: ACLs and Bridged Packets

Figure 79: Applying ACLs on Bridged Packets

This example shows how an ACL is applied on fallback-bridged packets. For bridged packets, only Layer 2 ACLs are applied to the input VLAN. Only non-IP, non-ARP packets can be fallback-bridged.
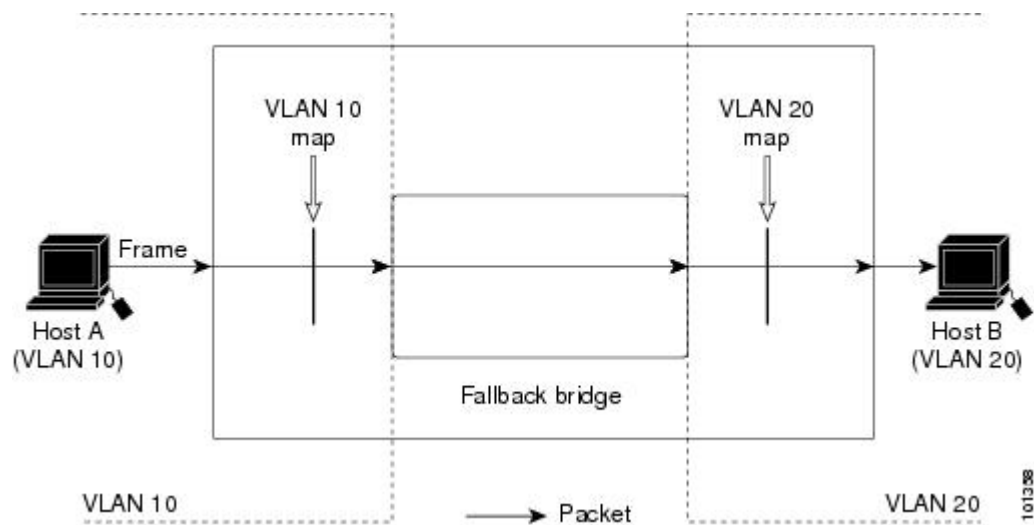
Example: ACLs and Routed Packets

Figure 80: Applying ACLs on Routed Packets

This example shows how ACLs are applied on routed packets. The ACLs are applied in this order:

1. VLAN map for input VLAN

2. Input router ACL

3. Output router ACL

4. VLAN map for output VLAN
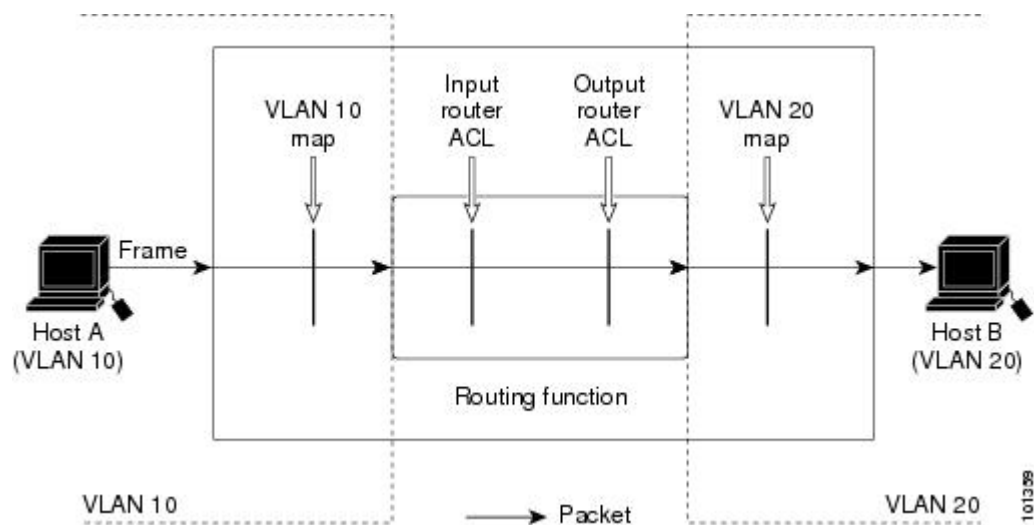
Example: ACLs and Multicast Packets

Figure 81: Applying ACLs on Multicast Packets

This example shows how ACLs are applied on packets that are replicated for IP multicasting. A multicast packet being routed has two different kinds of filters applied: one for destinations that are other ports in the input VLAN and another for each of the destinations that are in other VLANs to which the packet has been routed. The packet might be routed to more than one output VLAN, in which case a different router output ACL and VLAN map would apply for each destination VLAN. The final result is that the packet might be permitted in some of the output VLANs and not in others. A copy of the packet is forwarded to those destinations where it is permitted. However, if the input VLAN map drops the packet, no destination receives a copy of the packet.

Additional References

Related Documents

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for ACLs

Release	Feature Information
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Configuring IPv6 ACLs

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

IPv6 ACLs Overview

You can filter IP Version 6 (IPv6) traffic by creating IPv6 access control lists (ACLs) and applying them to interfaces similarly to the way that you create and apply IP Version 4 (IPv4) named ACLs. You can also create and apply input router ACLs to filter Layer 3 management traffic when the switch is running the IP base and LAN base feature sets.

A switch supports two types of IPv6 ACLs:

- IPv6 router ACLs are supported on outbound or inbound traffic on Layer 3 interfaces, which can be routed ports, switch virtual interfaces (SVIs), or Layer 3 EtherChannels. IPv6 router ACLs apply only to IPv6 packets that are routed.

- IPv6 port ACLs are supported on inbound and outbound Layer 2 interfaces. IPv6 port ACLs are applied to all IPv6 packets entering the interface.

The switch does not support VLAN ACLs (VLAN maps) for IPv6 traffic.

You can apply both IPv4 and IPv6 ACLs to an interface. As with IPv4 ACLs, IPv6 port ACLs take precedence over router ACLs.

Switch Stacks and IPv6 ACLs

The active switch supports IPv6 ACLs in hardware and distributes the IPv6 ACLs to the stack members.

If a standby switch takes over as the active switch, it distributes the ACL configuration to all stack members. The member switches sync up the configuration distributed by the new active switch and flush out entries that are not required.

When an ACL is modified, attached to, or detached from an interface, the active switch distributes the change to all stack members.

Interactions with Other Features and Switches

- If an IPv6 router ACL is configured to deny a packet, the packet is not routed. A copy of the packet is sent 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 or switch stack, and you can apply both IPv4 and IPv6 ACLs to the same interface. Each ACL must have a unique name; an error message appears if you try to use a name that is already configured.

  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, packets are dropped on the interface and an unload error message is logged.

Restrictions for IPv6 ACLs

With IPv4, you can configure standard and extended numbered IP ACLs, named IP ACLs, and MAC ACLs. IPv6 supports only named ACLs.

The switch supports most Cisco IOS-supported IPv6 ACLs with some exceptions:

- 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).

- This release supports only port ACLs and router ACLs for IPv6; it does not support VLAN ACLs (VLAN maps).

- The switch does not apply MAC-based ACLs on IPv6 frames.

> • You cannot apply IPv6 port ACLs to Layer 2 EtherChannels.
>
> • When configuring an ACL, there is no restriction on keywords entered in the ACL, regardless of whether or not they are supported on the platform. When you apply the ACL to an interface that requires hardware forwarding (physical ports or SVIs), the switch checks to determine whether or not the ACL can be supported on the interface. If not, attaching the ACL is rejected.
>
> • If an ACL is applied 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 ACL that is currently attached to the interface.

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, the packets associated with the ACL are dropped on the interface.
>
> • Logging is supported for router ACLs, but not for port ACLs.
>
> • The switch supports IPv6 address-matching for a full range of prefix-lengths.

Default Configuration for IPv6 ACLs

The default IPv6 ACL configuration is as follows:

```
Switch# show access-lists preauth_ipv6_acl
IPv6 access list preauth_ipv6_acl (per-user)
permit udp any any eq domain sequence 10
permit tcp any any eq domain sequence 20
permit icmp any any nd-ns sequence 30
permit icmp any any nd-na sequence 40
permit icmp any any router-solicitation sequence 50
permit icmp any any router-advertisement sequence 60
permit icmp any any redirect sequence 70
permit udp any eq 547 any eq 546 sequence 80
permit udp any eq 546 any eq 547 sequence 90
deny ipv6 any any sequence 100
```

Configuring IPv6 ACLs

To filter IPv6 traffic, you perform these steps:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. {**ipv6 access-list** *list-name*
4. {**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*]

5. {**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**]

6. {**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*]]

7. {**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*]

8. **end**

9. **show ipv6 access-list**

10. **show running-config**

11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	{**ipv6 access-list** *list-name* **Example:** Switch(config)# **ipv6 access-list example_acl_list**	Defines an IPv6 ACL name, and enters IPv6 access list configuration mode.
Step 4	{**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*]	Enter deny or permit to specify whether to deny or permit the packet if conditions are matched. These are the conditions: • For protocol, enter the 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. • The *source-ipv6-prefix/prefix-length* or *destination-ipv6-prefix/ prefix-length* is the source or destination IPv6 network or class of networks for which to set deny or permit conditions, specified in

Command or Action	Purpose
	hexadecimal and using 16-bit values between colons (see RFC 2373).
	• Enter any as an abbreviation for the IPv6 prefix ::/0.
	• For **host** *source-ipv6-address* or *destination-ipv6-address*, enter the source or destination IPv6 host address for which to set deny or permit conditions, specified in hexadecimal using 16-bit values between colons.
	• (Optional) For operator, specify an operand that compares the source or destination ports of the specified protocol. Operands are **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.
	• (Optional) The **port-number** is a decimal number from 0 to 65535 or the name of a TCP or UDP port. You can use TCP port names only when filtering TCP. You can use UDP port names only when filtering UDP.
	• (Optional) Enter **dscp** value to 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) Enter **fragments** to check noninitial fragments. This keyword is visible only if the protocol is ipv6.
	• (Optional) Enter **log** to cause an logging message to be sent to the console about the packet that matches the entry. Enter **log-input** to include the input interface in the log entry. Logging is supported only for router ACLs.
	• (Optional) Enter **routing** to specify that IPv6 packets be routed.
	• (Optional) Enter **sequence** *value* to specify the sequence number for the access list statement. The acceptable range is from 1 to 4,294,967,295.
	• (Optional) Enter **time-range** name to specify the time range that applies to the deny or permit statement.

	Command or Action	Purpose
Step 5	{**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}—Matches only packets that are not on a given port number. • **psh**—Push function bit set. • **range** {*port* | protocol}—Matches only packets in the port number range. • **rst**—Reset bit set. • **syn**—Synchronize bit set. • **urg**—Urgent pointer bit set.
Step 6	{**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 7	{**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 1, 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

	Command or Action	Purpose
		and code name. To see a list of ICMP message type names and code names, use the ? key or see command reference for this release.
Step 8	**end**	Return to privileged EXEC mode.
Step 9	**show ipv6 access-list**	Verify the access list configuration.
Step 10	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 11	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

What to do next

Attach the IPv6 ACL to an Interface

Attaching an IPv6 ACL to an Interface

You can apply an ACL to outbound or inbound traffic on Layer 3 interfaces, or to inbound traffic on Layer 2 interfaces. You can also apply ACLs only to inbound management traffic on Layer 3 interfaces.

Follow these steps to control access to an interface:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **no switchport**
5. **ipv6 address** *ipv6-address*
6. **ipv6 traffic-filter** *access-list-name* {**in** | **out**}
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Switch> **enable**	• Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**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.
Step 4	**no switchport**	If applying a router ACL, this changes the interface from Layer 2 mode (the default) to Layer 3 mode.
Step 5	**ipv6 address** *ipv6-address*	Configure an IPv6 address on a Layer 3 interface (for router ACLs).
Step 6	**ipv6 traffic-filter** *access-list-name* {**in** | **out**}	Apply the access list to incoming or outgoing traffic on the interface.
Step 7	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 8	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 9	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Monitoring 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 privileged EXEC commands shown in the table below:

Command	Purpose
show access-lists	Displays all access lists configured on the switch.
show ipv6 access-list [*access-list-name*]	Displays all configured IPv6 access lists or the access list specified by name.

This is an example of the output from the show access-lists privileged EXEC command. The output shows all access lists that are configured on the switch or switch stack.

```
Switch # show access-lists
Extended IP access list hello
    10 permit ip any any
IPv6 access list ipv6
    permit ipv6 any any sequence 10
```

This is an example of the output from the **show ipv6 access-list** privileged EXEC command. The output shows only IPv6 access lists configured on the switch or switch stack

```
Switch# show ipv6 access-list
IPv6 access list inbound
    permit tcp any any eq bgp (8 matches) sequence 10
    permit tcp any any eq telnet (15 matches) sequence 20
    permit udp any any sequence 30

IPv6 access list outbound
    deny udp any any sequence 10
    deny tcp any any eq telnet sequence 20
```

Additional References

Related Documents

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Configuring DHCP

Restrictions for DHCP

The following scenario is not supported:

A non-DHCP snooping VLAN, and the SVI of the non-DHCP snooping VLAN is configured on a device. The SVI of the non-DHCP snooping VLAN is configured with the status of *no shutdown*. In this scenario, the DHCP packets in the non-DHCP snooping VLAN are not forwarded to the trusted ports.

If the SVI of the non-DHCP snooping VLAN is not configured or is configured with the *shutdown* status, DHCP packets are forwarded to the trusted ports, and DHCP clients can obtain IP address from the DHCP server.

Information About DHCP

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. The switch can act as a DHCP server.

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 through an untrusted interface. By default, the switch considers all interfaces untrusted. So, the switch must be configured to trust some interfaces to use DHCP Snooping. 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, an example of an interface you might configure as trusted is one connected to a port on a device in the same network. An example of an untrusted interface is one that 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.

Normally, it is not desirable to broadcast packets to wireless clients. So, DHCP snooping replaces destination broadcast MAC address (ffff.ffff.ffff) with unicast MAC address for DHCP packets that are going from server to wireless clients. The unicast MAC address is retrieved from CHADDR field in the DHCP payload. This processing is applied for server to client packets such as DHCP OFFER, DHCP ACK, and DHCP NACK messages. The **ip dhcp snooping wireless bootp-broadcast enable** can be used to revert this behavior. When the wireless BOOTP broadcast is enabled, the broadcast DHCP packets from server are forwarded to wireless clients without changing the destination MAC address.

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 on the VLANs to which subscriber devices using option-82 are assigned.

The following illustration shows 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 82: DHCP Relay Agent in a Metropolitan Ethernet Network
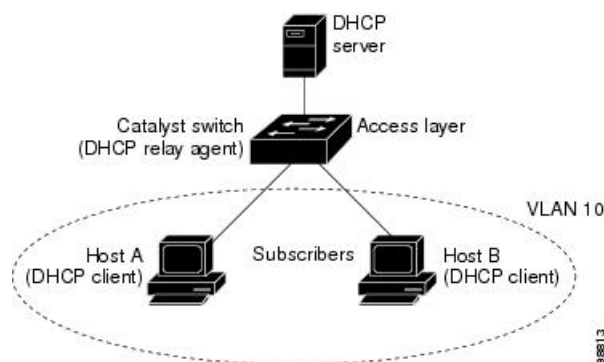
When you enable the DHCP snooping information option 82 on the switch, the following sequence of events occurs:

• 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. You can configure the remote ID and circuit ID.

- 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. Then the DHCP server echoes 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 do not change (see the illustration, *Suboption Packet Formats*):

- 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. For example, on a switch with 24 10/100/1000 ports and four small form-factor pluggable (SFP) module slots, port 3 is the Gigabit Ethernet 1/0/1 port, port 4 is the Gigabit Ethernet 1/0/2 port, and so forth. Port 27 is the SFP module slot Gigabit Ethernet1/0/25, and so forth.

The illustration, *Suboption Packet Formats*. shows the packet formats for the remote-ID suboption and the circuit-ID suboption when the default suboption configuration is used. For the circuit-ID suboption, the module number corresponds to the switch number in the stack. The switch uses the packet formats when you globally enable DHCP snooping and enter the ip dhcp snooping information option global configuration command.

Figure 83: Suboption Packet Formats

Circuit ID Suboption Frame Format

Suboption type | Circuit ID type
Length | Length

| 1 | 6 | 0 | 4 | VLAN | Module | Port |

1 byte | 1 byte | 1 byte | 1 byte | 2 bytes | 1 byte | 1 byte

Remote ID Suboption Frame Format

Suboption type | Remote ID type
Length | Length

| 2 | 8 | 0 | 6 | MAC address |

1 byte | 1 byte | 1 byte | 1 byte | 6 bytes
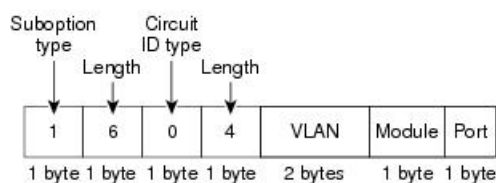
The illustration, *User-Configured Suboption Packet Formats,* shows the packet formats for user-configured remote-ID and circuit-ID suboptions The switch uses these packet formats when DHCP snooping is globally enabled and when the **ip dhcp snooping information option format remote-id** global configuration 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 84: User-Configured Suboption Packet Formats

Circuit ID Suboption Frame Format (for user-configured string):
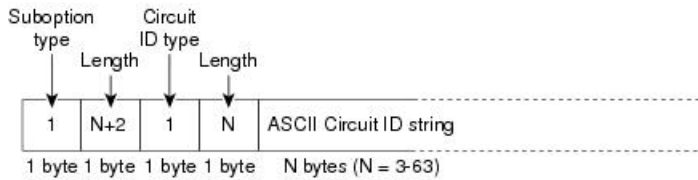
Remote ID Suboption Frame Format (for user-configured string):

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. For more information about manual and automatic address bindings, see the "Configuring DHCP" chapter of the *Cisco IOS IP Configuration Guide, Release 12.4*.

For procedures to enable and configure the Cisco IOS DHCP server database, see the "DHCP Configuration Task List" section in the "Configuring DHCP" chapter of the *Cisco IOS IP Configuration Guide, Release 12.4*.

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 64,000 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 cancel-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 Gi1/0/4 21ae5fbb
192.1.168.3 3 0003.44d6.c52f 2BB648EB Gi1/0/4 1bdb223f
192.1.168.2 3 0003.47d9.c8f1 2BB648AB Gi1/0/4 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.

DHCP Snooping and Switch Stacks

DHCP snooping is managed on the active switch. When a new switch joins the stack, the switch receives the DHCP snooping configuration from the active switch. When a member switch leaves the stack, all DHCP snooping address bindings associated with the switch age out.

All snooping statistics are generated on the active switch. If a new active switch is elected, the statistics counters reset.

When a stack merge occurs, all DHCP snooping bindings in the active switch are lost if it is no longer the active switch. With a stack partition, the existing active switch is unchanged, and the bindings belonging to

the partitioned switches age out. The new active switch of the partitioned stack begins processing the new incoming DHCP packets.

How to Configure DHCP Features

Default DHCP Snooping Configuration

Table 88: Default DHCP Configuration

Feature	Default Setting
DHCP server	Enabled in Cisco IOS software, requires configuration[12]
DHCP relay agent	Enabled[13]
DHCP packet forwarding address	None configured
Checking the relay agent information	Enabled (invalid messages are dropped)
DHCP relay agent forwarding policy	Replace the existing relay agent information
DHCP snooping enabled globally	Disabled
DHCP snooping information option	Enabled
DHCP snooping option to accept packets on untrusted input interfaces[14]	Disabled
DHCP snooping limit rate	None configured
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.

[12] The switch responds to DHCP requests only if it is configured as a DHCP server.
[13] The switch relays DHCP packets only if the IP address of the DHCP server is configured on the SVI of the DHCP client.

[14] 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

- 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.

- 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.

Configuring the DHCP Server

The switch can act as a DHCP server.

For procedures to configure the switch as a DHCP server, see the "Configuring DHCP" section of the "IP addressing and Services" section of the *Cisco IOS IP Configuration Guide, Release 12.4*.

DHCP Server and Switch Stacks

The DHCP binding database is managed on the stack's active switch. When a new active switch is assigned, the new active switch downloads the saved binding database from the TFTP server. When a switchover happens, the new active switch stack will use its database file that has been synced from the old active switch stack using the SSO function. The IP addresses associated with the lost bindings are released. You should configure an automatic backup by using the **ip dhcp database** *url* [**timeout** *seconds* | **write-delay** *seconds*] global configuration command.

Configuring the DHCP Relay Agent

Follow these steps to enable the DHCP relay agent on the switch:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **service dhcp**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Switch> **enable**	• Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**service dhcp** **Example:** Switch(config)# **service dhcp**	Enables the DHCP server and relay agent on your switch. By default, this feature is enabled.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

What to do next

See the "*Configuring DHCP*" section of the "IP Addressing and Services" section of the *Cisco IOS IP Configuration Guide, Release 12.4* for these procedures:

- Checking (validating) the relay agent information

- Configuring the relay agent forwarding policy

Specifying the 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.

Beginning in privileged EXEC mode, follow these steps to specify the packet forwarding address:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface vlan** *vlan-id*
4. **ip address** *ip-address subnet-mask*
5. **ip helper-address** *address*
6. **end**
7. Use one of the following:

    • **interface range** *port-range*
    • **interface** *interface-id*

8. **switchport mode access**
9. **switchport access vlan** *vlan-id*
10. **end**
11. **show running-config**
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface vlan** *vlan-id* **Example:** Switch(config)# interface vlan 1	Creates a switch virtual interface by entering a VLAN ID, and enter interface configuration mode.
Step 4	**ip address** *ip-address subnet-mask* **Example:** Switch(config-if)# ip address 192.108.1.27 255.255.255.0	Configures the interface with an IP address and an IP subnet.

	Command or Action	**Purpose**
Step 5	**ip helper-address** *address* **Example:** Switch(config-if)# ip helper-address 172.16.1.2	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.
Step 6	**end** **Example:** Switch(config-if)# end	Returns to global configuration mode.

Step 7	Use one of the following:
	
	    • **interface range** *port-range*
	    • **interface** *interface-id*
	
	 Example:
	
	 Switch(config)# interface gigabitethernet1/0/2
		Configures multiple physical ports that are connected to the DHCP clients, and enter interface range configuration mode.
		
		 or
		
		 Configures a single physical port that is connected to the DHCP client, and enter interface configuration mode.

Step 8	**switchport mode access** **Example:** Switch(config-if)# switchport mode access	Defines the VLAN membership mode for the port.
Step 9	**switchport access vlan** *vlan-id* **Example:** Switch(config-if)# switchport access vlan 1	Assigns the ports to the same VLAN as configured in Step 2.
Step 10	**end** **Example:** Switch(config-if)# end	Returns to privileged EXEC mode.
Step 11	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 12	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Prerequisites for Configuring DHCP Snooping and Option 82

The prerequisites for DHCP Snooping and Option 82 are as follows:

- You must globally enable DHCP snooping on the switch.

- 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.

- If you want the switch to respond to DHCP requests, it must be configured as a DHCP server.

- Before configuring the DHCP snooping information option on your switch, be sure to configure the device that is acting as the DHCP server. You must specify the IP addresses that the DHCP server can assign or exclude, or you must configure DHCP options for these devices.

- For DHCP snooping to function properly, all DHCP servers must be connected to the switch through trusted interfaces. In a service-provider network, a trusted interface is connected to a port on a device in the same network.

- You must configure the switch to use the Cisco IOS DHCP server binding database to use it for DHCP snooping.

- To use the DHCP snooping option of accepting packets on untrusted inputs, the switch must be an aggregation switch that receives packets with option-82 information from an edge switch.

- The following prerequisites apply to DHCP snooping binding database configuration:

    - You must configure a destination on the DHCP snooping binding database to use the switch for DHCP snooping.

    - 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 Network Time Protocol (NTP).

    - If NTP is configured, the switch writes binding changes to the binding file only when the switch system clock is synchronized with NTP.

- Before configuring the DHCP relay agent on your switch, make sure to configure the device that is acting as the DHCP server. 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 you want the switch to relay DHCP packets, the IP address of the DHCP server must be configured on the switch virtual interface (SVI) of the DHCP client.

- 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.

Enabling DHCP Snooping and Option 82

Follow these steps to enable DHCP snooping on the switch:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip dhcp snooping**
4. **ip dhcp snooping vlan** *vlan-range*
5. **ip dhcp snooping information option**
6. **ip dhcp snooping information option format remote-id** [**string** *ASCII-string* | **hostname**]
7. **ip dhcp snooping information option allow-untrusted**
8. **interface** *interface-id*
9. **ip dhcp snooping vlan** *vlan* **information option format-type circuit-id** [**override**] **string** *ASCII-string*
10. **ip dhcp snooping trust**
11. **ip dhcp snooping limit rate** *rate*
12. **exit**
13. **ip dhcp snooping verify mac-address**
14. **end**
15. **show running-config**
16. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**ip dhcp snooping** **Example:** Switch(config)# **ip dhcp snooping**	Enables DHCP snooping globally.
Step 4	**ip dhcp snooping vlan** *vlan-range* **Example:** Switch(config)# **ip dhcp snooping vlan 10**	Enables DHCP snooping on a VLAN or range of VLANs. The range is 1 to 4094. 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

	Command or Action	**Purpose**
		hyphens, or a range of VLAN IDs separated by entering the starting and ending VLAN IDs separated by a space.
		• 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.
Step 5	**ip dhcp snooping information option** **Example:** Switch(config)# **ip dhcp snooping information option**	Enables the switch to insert and remove DHCP relay information (option-82 field) in forwarded DHCP request messages to the DHCP server. This is the default setting.
Step 6	**ip dhcp snooping information option format remote-id** [**string** *ASCII-string* | **hostname**] **Example:** Switch(config)# **ip dhcp snooping information option format remote-id string acsiistring2**	(Optional) Configures the remote-ID suboption. You can configure the remote ID as: • String of up to 63 ASCII characters (no spaces) • Configured 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.
Step 7	**ip dhcp snooping information option allow-untrusted** **Example:** Switch(config)# **ip dhcp snooping information option allow-untrusted**	(Optional) If the switch is an aggregation switch connected to an edge switch, this command enables 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.
Step 8	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet2/0/1**	Specifies the interface to be configured, and enter interface configuration mode.
Step 9	**ip dhcp snooping vlan** *vlan* **information option format-type circuit-id** [**override**] **string** *ASCII-string* **Example:** Switch(config-if)# **ip dhcp snooping vlan 1 information option format-type curcuit-id override string ovrride2**	(Optional) Configures the circuit-ID suboption for the specified interface. Specify the VLAN and port identifier, using a VLAN ID in the range of 1 to 4094. 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).

	Command or Action	**Purpose**
		(Optional) Use the **override** keyword when you do not want the circuit-ID suboption inserted in TLV format to define subscriber information.
Step 10	**ip dhcp snooping trust** Example: Switch(config-if)# **ip dhcp snooping trust**	(Optional) Configures the interface as trusted or untrusted. Use the **no** keyword to configure an interface to receive messages from an untrusted client. The default setting is untrusted.
Step 11	**ip dhcp snooping limit rate** *rate* Example: Switch(config-if)# **ip dhcp snooping limit rate 100**	(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.
Step 12	**exit** Example: Switch(config-if)# **exit**	Returns to global configuration mode.
Step 13	**ip dhcp snooping verify mac-address** Example: Switch(config)# **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.
Step 14	**end** Example: Switch(config)# **end**	Returns to privileged EXEC mode.
Step 15	**show running-config** Example: Switch# **show running-config**	Verifies your entries.
Step 16	**copy running-config startup-config** Example: Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Enabling the Cisco IOS DHCP Server Database

For procedures to enable and configure the Cisco IOS DHCP server database, see the "DHCP Configuration Task List" section in the "Configuring DHCP" chapter of the Cisco IOS IP Configuration Guide, Release 12.4

Monitoring DHCP Snooping Information

Table 89: Commands for Displaying DHCP Information

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 snooping statistics	Displays the DHCP snooping statistics in summary or detail form.
show ip source binding	Display the dynamically and statically configured bindings.

Note   If DHCP snooping is enabled and an interface changes to the down state, the switch does not delete the statically configured bindings.

Configuring DHCP Server Port-Based Address Allocation

Information About Configuring 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.

Default Port-Based Address Allocation Configuration

By default, DHCP server port-based address allocation is disabled.

Port-Based Address Allocation Configuration Guidelines

- By default, DHCP server port-based address allocation is disabled.

- 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.

Enabling the DHCP Snooping Binding Database Agent

Beginning in privileged EXEC mode, follow these steps to enable and configure the DHCP snooping binding database agent on the switch:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip dhcp snooping database** {**flash**[*number*]**:**/*filename* | **ftp://***user***:***password***@***host*/*filename* | **http://**[[*username***:***password*]**@**]{*hostname* | *host-ip*}[/*directory*] /*image-name***.tar** | **rcp://***user***@***host*/*filename*}| **tftp://***host*/*filename*
4. **ip dhcp snooping database timeout** *seconds*
5. **ip dhcp snooping database write-delay** *seconds*
6. **end**
7. **ip dhcp snooping binding** *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id* **expiry** *seconds*
8. **show ip dhcp snooping database** [**detail**]
9. **show running-config**
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable**	Enables privileged EXEC mode.
	Example:	• Enter your password if prompted.

	Command or Action	Purpose
	Switch> **enable**	
Step 2	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 3	**ip dhcp snooping database** {**flash**[*number*]**:**/*filename* | **ftp:**//*user***:***password*@*host*/*filename* | **http:**//[[*username***:***password*]@]{*hostname* / *host-ip*}[/*directory*] /*image-name***.tar** | **rcp:**//*user*@*host*/*filename*}| **tftp:**//*host*/*filename* Example: Switch(config)# **ip dhcp snooping database** **tftp://10.90.90.90/snooping-rp2**	Specifies the URL for the database agent or the binding file by using one of these forms: • **flash**[*number*]**:**/*filename* (Optional) Use the *number* parameter to specify the stack member number of the active switch. The range for *number* is 1 to 9. • **ftp:**//*user***:***password*@*host*/*filename* • **http:**//[[*username***:***password*]@]{*hostname* / *host-ip*}[/*directory*] /*image-name***.tar** • **rcp:**//*user*@*host*/*filename* • **tftp:**//*host*/*filename*
Step 4	**ip dhcp snooping database timeout** *seconds* Example: Switch(config)# **ip dhcp snooping database timeout** **300**	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.
Step 5	**ip dhcp snooping database write-delay** *seconds* Example: Switch(config)# **ip dhcp snooping database** **write-delay 15**	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).
Step 6	**end** Example: Switch(config)# **end**	Returns to privileged EXEC mode.
Step 7	**ip dhcp snooping binding** *mac-address* **vlan** *vlan-id* *ip-address* **interface** *interface-id* **expiry** *seconds* Example: Switch# **ip dhcp snooping binding 0001.1234.1234** **vlan 1 172.20.50.5 interface gi1/1 expiry 1000**	(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.

	Command or Action	Purpose
		Use this command when you are testing or debugging the switch.
Step 8	**show ip dhcp snooping database** [**detail**] **Example:** `Switch# show ip dhcp snooping database detail`	Displays the status and statistics of the DHCP snooping binding database agent.
Step 9	**show running-config** **Example:** `Switch# show running-config`	Verifies your entries.
Step 10	**copy running-config startup-config** **Example:** `Switch# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Enabling DHCP Server Port-Based Address Allocation

Follow these steps to globally enable port-based address allocation and to automatically generate a subscriber identifier on an interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip dhcp use subscriber-id client-id**
4. **ip dhcp subscriber-id interface-name**
5. **interface** *interface-id*
6. **ip dhcp server use subscriber-id client-id**
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Switch> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**ip dhcp use subscriber-id client-id** **Example:** Switch(config)# **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.
Step 4	**ip dhcp subscriber-id interface-name** **Example:** Switch(config)# **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.
Step 5	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	Specifies the interface to be configured, and enter interface configuration mode.
Step 6	**ip dhcp server use subscriber-id client-id** **Example:** Switch(config-if)# **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.
Step 7	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 8	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 9	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

What to do next

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.

Monitoring DHCP Server Port-Based Address Allocation

Table 90: Commands for Displaying DHCP Port-Based Address Allocation Information

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.

Additional References

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for DHCP Snooping and Option 82

Release	Feature Information
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Release	Feature Information
	Introduced support for the following commands: • **show ip dhcp snooping statistics** user EXEC command for displaying DHCP snooping statistics. • **clear ip dhcp snooping statistics** privileged EXEC command for clearing the snooping statistics counters.

Configuring 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.

This chapter contains the following topics:

Information About IP Source Guard

IP Source Guard

You can use IP source guard to prevent traffic attacks if a host tries to use the IP address of its neighbor and you can enable IP source guard 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.

The switch uses a source IP lookup table in hardware to bind IP addresses to ports. For IP and MAC filtering, a combination of source IP and source MAC lookups are used. IP traffic with a source IP address is the binding table is allowed, all other traffic is denied.

The IP source binding table has bindings that are learned by DHCP snooping or are manually configured (static IP source bindings). An entry in this table has an IP address, its associated MAC address, and its associated VLAN number. The switch uses the IP source binding table only when IP source guard is enabled.

IPSG is supported only on Layer 2 ports, including access and trunk ports. You can configure IPSG with source IP address filtering or with source IP and MAC address filtering.

IP Source Guard for Static Hosts

✎

Note Do not use IPSG (IP source guard) for static hosts on uplink ports or trunk ports.

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. In a stacked environment, when the active switch failover occurs, the IP source guard entries for static hosts attached to member ports are retained. 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 vender 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

- 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.

| Note | 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. |

• You can enable this feature when 802.1x port-based authentication is enabled.

How to Configure IP Source Guard

Enabling IP Source Guard

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip verify source** [**mac-check**]
5. **exit**
6. **ip source binding** *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id*
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet 1/0/1**	Specifies the interface to be configured, and enters interface configuration mode.
Step 4	**ip verify source** [**mac-check**]	Enables IP source guard with source IP address filtering.

	Command or Action	Purpose
	Example: Switch(config-if)# **ip verify source**	(Optional) **mac-check**—Enables IP Source Guard with source IP address and MAC address filtering.
Step 5	**exit** **Example:** Switch(config-if)# **exit**	Returns to global configuration mode.
Step 6	**ip source binding** *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id* **Example:** Switch(config)# **ip source binding 0100.0230.0002 vlan 11 10.0.0.4 interface gigabitethernet1/0/1**	Adds a static IP source binding. Enter this command for each static binding.
Step 7	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 8	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 9	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring IP Source Guard for Static Hosts on a Layer 2 Access Port

You must 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 by setting an IP device tracking maximum on that interface, IPSG with static hosts rejects all the IP traffic from that interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip device tracking**

4. **interface** *interface-id*
5. **switchport mode access**
6. **switchport access vlan** *vlan-id*
7. **ip verify source**[**tracking**] [**mac-check**]
8. **ip device tracking maximum** *number*
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**ip device tracking** **Example:** Switch(config)# **ip device tracking**	Turns on the IP host table, and globally enables IP device tracking.
Step 4	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet 1/0/1**	Enters interface configuration mode.
Step 5	**switchport mode access** **Example:** Switch(config-if)# **switchport mode access**	Configures a port as access.
Step 6	**switchport access vlan** *vlan-id* **Example:** Switch(config-if)# **switchport access vlan 10**	Configures the VLAN for this port.
Step 7	**ip verify source**[**tracking**] [**mac-check**] **Example:**	Enables IP source guard with source IP address filtering. (Optional) **tracking**—Enables IP source guard for static hosts.

	Command or Action	Purpose
	`Switch(config-if)# ip verify source tracking mac-check`	(Optional) **mac-check**—Enables MAC address filtering. The command **ip verify source tracking mac-check**enables IP source guard for static hosts with MAC address filtering.
Step 8	**ip device tracking maximum** *number* **Example:** `Switch(config-if)# ip device tracking maximum 8`	Establishes 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.
Step 9	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.

Monitoring IP Source Guard

Table 91: Privileged EXEC show Commands

Command	Purpose
show ip verify source [**interface** *interface-id*]	Displays the IP source guard configuration on the switch or on a specific interface.
show ip device tracking { **all** | **interface** *interface-id* | **ip** *ip-address* | **mac** *imac-address*}	Displays information about the entries in the IP device tracking table.

Table 92: Interface Configuration Commands

Command	Purpose
ip verify source tracking	Verifies the data source.

For detailed information about the fields in these displays, see the command reference for this release.

Additional References

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Configuring Dynamic ARP Inspection

Restrictions for Dynamic ARP Inspection

This section lists the restrictions and guidelines for configuring Dynamic ARP Inspection on the switch.

- Dynamic ARP inspection is an ingress security feature; it does not perform any egress checking.

- Dynamic ARP inspection is not effective for hosts connected to switches that do not support dynamic ARP inspection 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 dynamic ARP inspection checks from the one with no checking. This action secures the ARP caches of hosts in the domain enabled for dynamic ARP inspection.

- Dynamic 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.

  When DHCP snooping is disabled or in non-DHCP environments, use ARP ACLs to permit or to deny packets.

- Dynamic ARP inspection is supported on access ports, trunk ports, and EtherChannel ports.

| **Note** | Do not enable Dynamic ARP inspection on RSPAN VLANs. If Dynamic ARP inspection is enabled on RSPAN VLANs, Dynamic ARP inspection 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 rate limit is calculated separately on each switch in a switch stack. For a cross-stack EtherChannel, this means that the actual rate limit might be higher than the configured value. For example, if you set the rate limit to 30 pps on an EtherChannel that has one port on switch 1 and one port on switch 2, each port can receive packets at 29 pps without causing the EtherChannel to become error-disabled.

- 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 dynamic ARP inspection-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 dynamic ARP inspection 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.

- In the presence of vlan-bridging & IP device tracking, the cross-stack ARP packet forwarding will not work.

Understanding Dynamic ARP Inspection

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 26-1 shows an example of ARP cache poisoning.

Figure 85: ARP Cache Poisoning
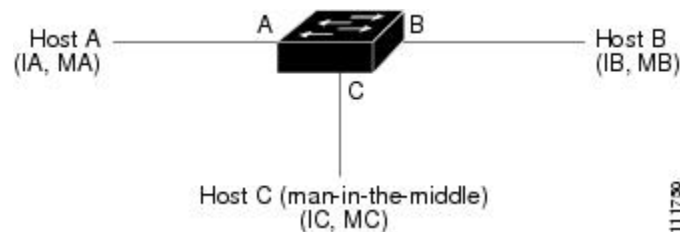
Hosts A, B, and C are connected to the switch on interfaces A, B and C, all of which are on the same subnet. Their IP and MAC addresses are shown in parentheses; for example, Host A uses IP address IA and MAC address MA. When Host A needs to communicate to Host B at the IP layer, it broadcasts an ARP 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.

Dynamic ARP inspection 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.

Dynamic ARP inspection 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

Dynamic ARP inspection 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.

You enable dynamic ARP inspection on a per-VLAN basis by using the **ip arp inspection vlan** *vlan-range* global configuration command.

In non-DHCP environments, dynamic ARP inspection can validate ARP packets against user-configured ARP access control lists (ACLs) for hosts with statically configured IP addresses. You define an ARP ACL by using the **arp access-list** *acl-name* global configuration command.

You can configure dynamic ARP inspection to drop ARP packets when the IP addresses in the packets are invalid or when the MAC addresses in the body of the ARP packets do not match the addresses specified in the Ethernet header. Use the **ip arp inspection validate** {[**src-mac**] [**dst-mac**] [**ip**]} global configuration command.

Interface Trust States and Network Security

Dynamic ARP inspection associates a trust state with each interface on the switch. Packets arriving on trusted interfaces bypass all dynamic ARP inspection validation checks, and those arriving on untrusted interfaces undergo the dynamic ARP inspection 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 theip arp inspection trust interface configuration command.

⚠️

Caution Use the trust state configuration carefully. Configuring interfaces as untrusted when they should betrusted can result in a loss of connectivity.

In the following figure, assume that both Switch A and Switch B are running dynamic ARP inspection 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 86: ARP Packet Validation on a VLAN Enabled for Dynamic ARP Inspection
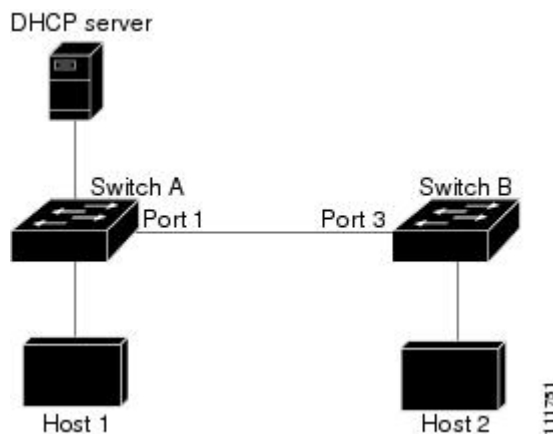
Configuring interfaces to be trusted when they are actually untrusted leaves a security hole in the network. If Switch A is not running dynamic ARP inspection, Host 1 can easily poison the ARP cache of Switch B (and Host 2, if the link between the switches is configured as trusted). This condition can occur even though Switch B is running dynamic ARP inspection.

Dynamic ARP inspection ensures that hosts (on untrusted interfaces) connected to a switch running dynamic ARP inspection do not poison the ARP caches of other hosts in the network. However, dynamic ARP inspection does not prevent hosts in other portions of the network from poisoning the caches of the hosts that are connected to a switch running dynamic ARP inspection.

In cases in which some switches in a VLAN run dynamic ARP inspection and other switches do not, configure the interfaces connecting such switches as untrusted. However, to validate the bindings of packets from nondynamic ARP inspection switches, configure the switch running dynamic ARP inspection with ARP ACLs. When you cannot determine such bindings, at Layer 3, isolate switches running dynamic ARP inspection from switches not running dynamic ARP inspection 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 dynamic ARP inspection 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 The rate limit for an EtherChannel is applied separately to each switch in a stack. For example, if a limit of 20 pps is configured on the EtherChannel, each switch with ports in the EtherChannel can carry up to 20 pps. If any switch exceeds the limit, the entire EtherChannel is placed into the error-disabled state.

Relative Priority of ARP ACLs and DHCP Snooping Entries

Dynamic ARP inspection 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.

Default Dynamic ARP Inspection Configuration

Feature	Default Settings
Dynamic ARP inspection	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 dynamic ARP inspection 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.

Relative Priority of ARP ACLs and DHCP Snooping Entries

Dynamic ARP inspection 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.

Configuring ARP ACLs for Non-DHCP Environments

This procedure shows how to configure dynamic ARP inspection when Switch B shown in Figure 2 does not support dynamic ARP inspection 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.

Follow these steps to configure an ARP ACL on Switch A. This procedure is required in non-DHCP environments.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **arp access-list** *acl-name*
4. **permit ip host** *sender-ip* **mac host** *sender-mac*
5. **exit**
6. **ip arp inspection filter** *arp-acl-name* **vlan** *vlan-range* [**static**]
7. **interface** *interface-id*
8. **no ip arp inspection trust**
9. **end**
10. Use the following show commands:

    • **show arp access-list** acl-name
    • **show ip arp inspection vlan** *vlan-range*
    • **show ip arp inspection interfaces**

11. **show running-config**
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**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.

	Command or Action	**Purpose**
Step 4	**permit ip host** *sender-ip* **mac host** *sender-mac*	Permits ARP packets from the specified host (Host 2).
		• For *sender-ip*, enter the IP address of Host 2.
		• For *sender-mac*, enter the MAC address of Host 2.
Step 5	**exit**	Returns to global configuration mode.
Step 6	**ip arp inspection filter** *arp-acl-name* **vlan** *vlan-range* [**static**]	Applies ARP ACL to the VLAN. By default, no defined ARP ACLs are applied to any VLAN.
		• For *arp-acl-name*, specify the name of the ACL created in Step 2.
		• For *vlan-range*, specify 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 4094.
		• (Optional) Specify **static** 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.
Step 7	**interface** *interface-id*	Specifies Switch A interface that is connected to Switch B, and enters the interface configuration mode.
Step 8	**no ip arp inspection trust**	Configures 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.
Step 9	**end**	Returns to privileged EXEC mode.

	Command or Action	**Purpose**

Step 10	Use the following show commands:
	
	    • **show arp access-list** acl-name
	    • **show ip arp inspection vlan** *vlan-range*
	    • **show ip arp inspection interfaces**
		Verifies your entries.

Step 11	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 12	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Dynamic ARP Inspection in DHCP Environments

Before you begin

This procedure shows how to configure dynamic ARP inspection when two switches support this feature. Host 1 is connected to Switch A, and Host 2 is connected to Switch B. Both switches are running dynamic ARP inspection 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.

Note Dynamic 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.

Follow these steps to configure dynamic ARP inspection. You must perform this procedure on both switches. This procedure is required.

SUMMARY STEPS

    1.   **enable**
    2.   **show cdp neighbors**
    3.   **configure terminal**
    4.   **ip arp inspection vlan** *vlan-range*
    5.   **Interface** *interface-id*
    6.   **ip arp inspection trust**
    7.   **end**
    8.   **show ip arp inspection interfaces**

9. **show ip arp inspection vlan** *vlan-range*
10. **show ip dhcp snooping binding**
11. **show ip arp inspection statistics vlan** *vlan-range*
12. **configure terminal**
13. **configure terminal**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**show cdp neighbors** **Example:** Switch(config-if)#**show cdp neighbors**	Verify the connection between the switches.
Step 3	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 4	**ip arp inspection vlan** *vlan-range* **Example:** Switch(config)# **ip arp inspection vlan 1**	Enable dynamic ARP inspection on a per-VLAN basis. By default, dynamic ARP inspection is disabled on all VLANs. For vlan-range, 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 4094. Specify the same VLAN ID for both switches.
Step 5	**Interface***interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	Specifies the interface connected to the other switch, and enter interface configuration mode.
Step 6	**ip arp inspection trust** **Example:** Switch(config-if)#**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 simply 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

	Command or Action	Purpose
		logging configuration specified with the ip arp inspection vlan logging global configuration command.
Step 7	**end** **Example:** Switch(config-if)#**end**	Returns to privileged EXEC mode.
Step 8	**show ip arp inspection interfaces** **Example:**	Verifies the dynamic ARP inspection configuration on interfaces.
Step 9	**show ip arp inspection vlan** *vlan-range* **Example:** Switch(config-if)#**show ip arp inspection vlan 1**	Verifies the dynamic ARP inspection configuration on VLAN.
Step 10	**show ip dhcp snooping binding** **Example:** Switch(config-if)#**show ip dhcp snooping binding**	Verifies the DHCP bindings.
Step 11	**show ip arp inspection statistics vlan** *vlan-range* **Example:** Switch(config-if)#**show ip arp inspection statistics vlan 1**	Checks the dynamic ARP inspection statistics on VLAN.
Step 12	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 13	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.

Limiting the Rate of Incoming ARP Packets

The switch CPU performs dynamic ARP inspection validation checks; therefore, the number of incoming ARP packets is rate-limited to prevent a denial- of-service attack.

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 enable error-disabled recovery so that ports automatically emerge from this state after a specified timeout period.

![note icon]

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.

Follow these steps to limit the rate of incoming ARP packets. This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip arp inspection limit {rate pps [burst interval seconds] | none}**
5. **exit**
6. Use the following commands:
   - **errdisable detect cause arp-inspection**
   - **errdisable recovery cause arp-inspection**
   - **errdisable recovery interval** *interval*
7. **exit**
8. Use the following show commands:
   - **show ip arp inspection interfaces**
   - **show errdisable recovery**
9. **show running-config**
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable**	Enables privileged EXEC mode.
	Example:	• Enter your password if prompted.
	Switch> **enable**	
Step 2	**configure terminal**	Enters global configuration mode.
	Example:	
	Switch# **configure terminal**	
Step 3	**interface** *interface-id*	Specifies the interface to be rate-limited, and enter interface configuration mode.
Step 4	**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

	Command or Action	**Purpose**
		interfaces and unlimited on trusted interfaces. The burst interval is 1 second.
		The keywords have these meanings:
		• For **rate***pps*, specify an upper limit for the number of incoming packets processed per second. The range is 0 to 2048 pps.
		• (Optional) For **burst interval***seconds*, specify the consecutive interval in seconds, over which the interface is monitored for a high rate of ARP packets. The range is 1 to 15.
		• For **rate none**, specify no upper limit for the rate of incoming ARP packets that can be processed.
Step 5	**exit**	Returns to global configuration mode.
Step 6	Use the following commands: • **errdisable detect cause arp-inspection** • **errdisable recovery cause arp-inspection** • **errdisable recovery interval** *interval*	(Optional) Enables error recovery from the dynamic ARP inspection error-disabled state, and configure the dynamic ARP inspection recover mechanism variables. By default, recovery is disabled, and the recovery interval is 300 seconds. For **interval** *interval*, specify the time in seconds to recover from the error-disabled state. The range is 30 to 86400.
Step 7	**exit**	Returns to privileged EXEC mode.
Step 8	Use the following show commands: • **show ip arp inspection interfaces** • **show errdisable recovery**	Verifies your settings.
Step 9	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 10	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Performing Dynamic ARP Inspection Validation Checks

Dynamic ARP inspection intercepts, logs, and discards ARP packets with invalid IP-to-MAC address bindings. You can configure the switch to perform additional checks on the destination MAC address, the sender and target IP addresses, and the source MAC address.

Follow these steps to perform specific checks on incoming ARP packets. This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip arp inspection validate {[src-mac] [dst-mac] [ip]}**
4. **exit**
5. **show ip arp inspection vlan** *vlan-range*
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**ip arp inspection validate {[src-mac] [dst-mac] [ip]}**	Performs a specific check on incoming ARP packets. By default, no checks are performed. The keywords have these meanings: • For **src-mac**, check 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. • For **dst-mac**, check 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.

	Command or Action	Purpose
		• For **ip**, check 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.
Step 4	**exit**	Returns to privileged EXEC mode.
Step 5	**show ip arp inspection vlan** *vlan-range*	Verifies your settings.
Step 6	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Monitoring DAI

To monitor DAI, use the following commands:

Command	Description
clear ip arp inspection statistics	Clears dynamic ARP inspection statistics.
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 dynamic ARP inspection enabled (active).
clear ip arp inspection log	Clears the dynamic ARP inspection log buffer.
show ip arp inspection log	Displays the configuration and contents of the dynamic ARP inspection log buffer.

For the **show ip arp inspection statistics** command, the switch increments the number of forwarded packets for each ARP request and response packet on a trusted dynamic ARP inspection port. The switch increments the number of ACL or DHCP permitted packets for each packet that is denied by source MAC, destination MAC, or IP validation checks, and the switch increments the appropriate.

Verifying the DAI Configuration

To display and verify the DAI configuration, use the following commands:

Command	Description
show arp access-list [*acl-name*]	Displays detailed information about ARP ACLs.
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 vlan *vlan-range*	Displays the configuration and the operating state of dynamic ARP inspection for the specified VLAN. If no VLANs are specified or if a range is specified, displays information only for VLANs with dynamic ARP inspection enabled (active).

Additional References

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Configuring IEEE 802.1x Port-Based Authentication

This chapter describes how to configure IEEE 802.1x port-based authentication. IEEE 802.1x authentication prevents unauthorized devices (clients) from gaining access to the network. Unless otherwise noted, the term *switch* refers to a standalone switch or a switch stack.

Information About 802.1x Port-Based Authentication

The 802.1x standard defines a client-server-based access control and authentication protocol that prevents unauthorized clients from connecting to a LAN through publicly accessible ports unless they are properly authenticated. The authentication server authenticates each client connected to a switch port before making available any services offered by the switch or the LAN.

Note TACACS is not supported with 802.1x authentication.

Until the client is authenticated, 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 is successful, normal traffic can pass through the port.

The table shown below lists the maximum number of each client session supported on Catalyst 3850 and Catalyst 3650 switches:

Client session	Maximum sessions supported
Maximum dot1x or MAB client sessions	2000
Maximum web-based authentication sessions	2000

Client session	Maximum sessions supported
Maximum dot1x sessions with critical-auth VLAN enabled and server re-initialized	2000
Maximum MAB sessions with various session features applied	2000
Maximum dot1x sessions with service templates or session features applied	2000

Note For complete syntax and usage information for the commands used in this chapter, see the "RADIUS Commands" section in the *Cisco IOS Security Command Reference, Release 3SE*

Port-Based Authentication Process

To configure IEEE 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.

The AAA process begins with authentication. 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.

If Multi Domain Authentication (MDA) is enabled on a port, this flow can be used with some exceptions that are applicable to voice authorization.

Figure 87: Authentication Flowchart

This figure shows the authentication process.
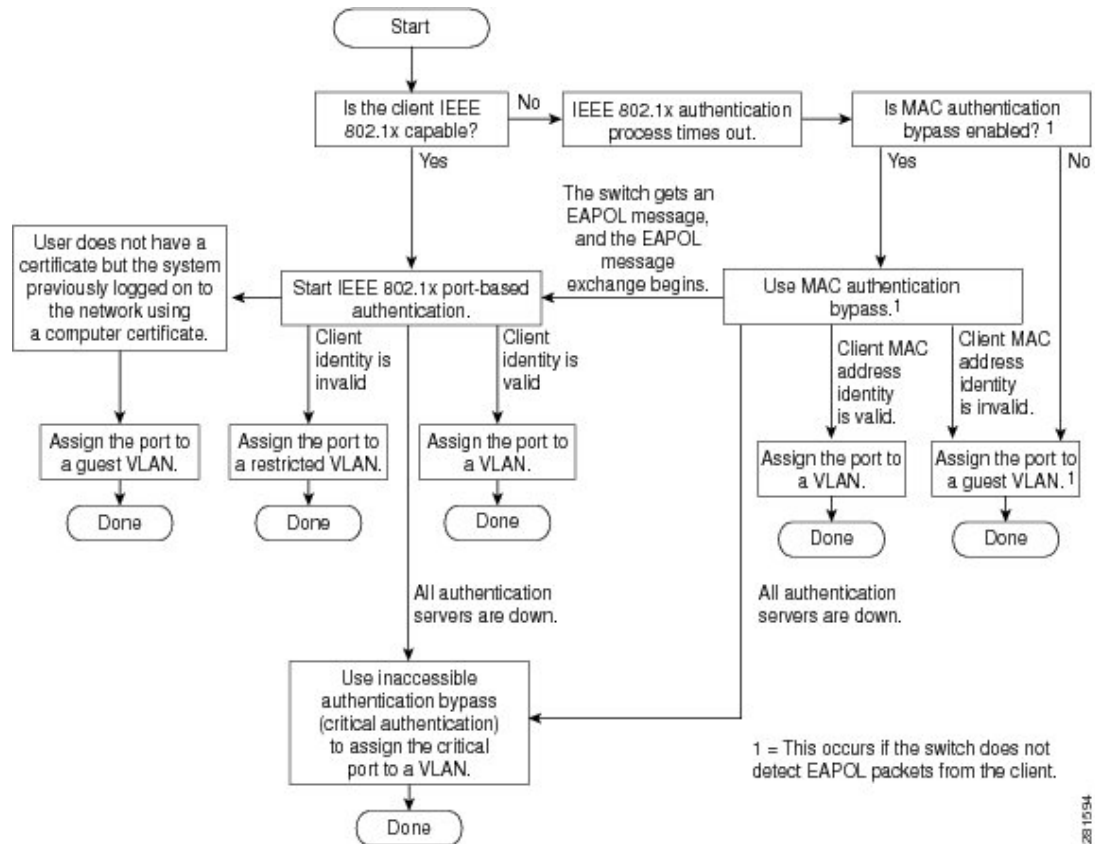
The switch re-authenticates a client when one of these situations occurs:

• Periodic re-authentication is enabled, and the re-authentication timer expires.

You can configure the re-authentication 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 re-authentication occurs.

The Termination-Action RADIUS attribute (Attribute [29]) specifies the action to take during re-authentication. 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 re-authentication. When the *ReAuthenticate* action is set (the attribute value is RADIUS-Request), the session is not affected during re-authentication.

Note	On an interface having authentication configuration and sessions, we recommend you not to *DEFAULT* the interface with session in place. The console will hang if you *DEFAULT* a range of interfaces in a stack.

• You manually re-authenticate the client by entering the **dot1x re-authenticate interface** *interface-id* privileged EXEC command.

Port-Based 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 bootup, 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.

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.

The specific exchange of EAP frames depends on the authentication method being used.

Figure 88: Message Exchange

This figure shows a message exchange initiated by the client when the client uses the One-Time-Password (OTP) authentication method with a RADIUS server.
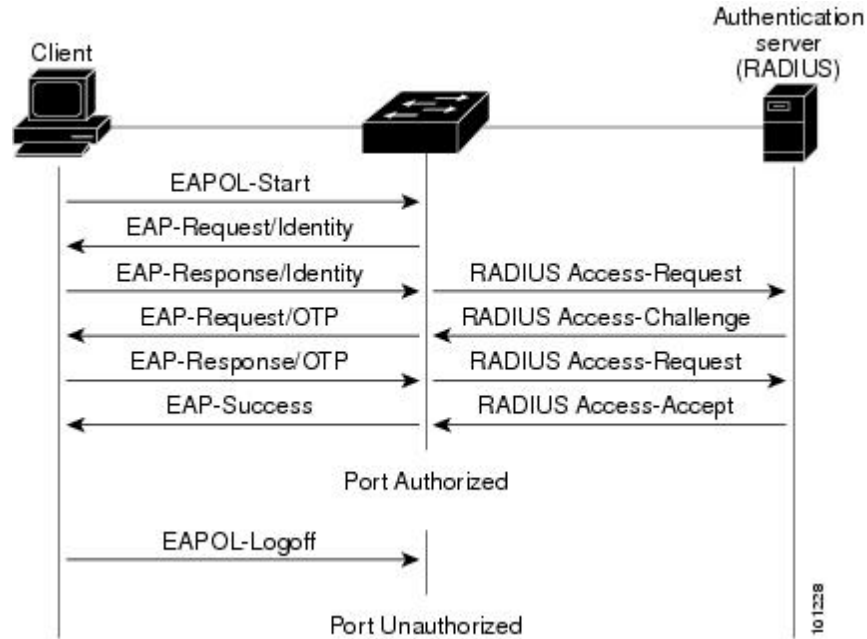
If 802.1x authentication times out while waiting for an EAPOL message exchange and MAC authentication bypass is enabled, the switch can authorize the client when the switch detects an Ethernet packet from the client. The switch uses the MAC address of the client as its identity and includes this information 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 starts 802.1x authentication.

Figure 89: Message Exchange During MAC Authentication Bypass

This figure shows the message exchange during MAC authentication bypass.
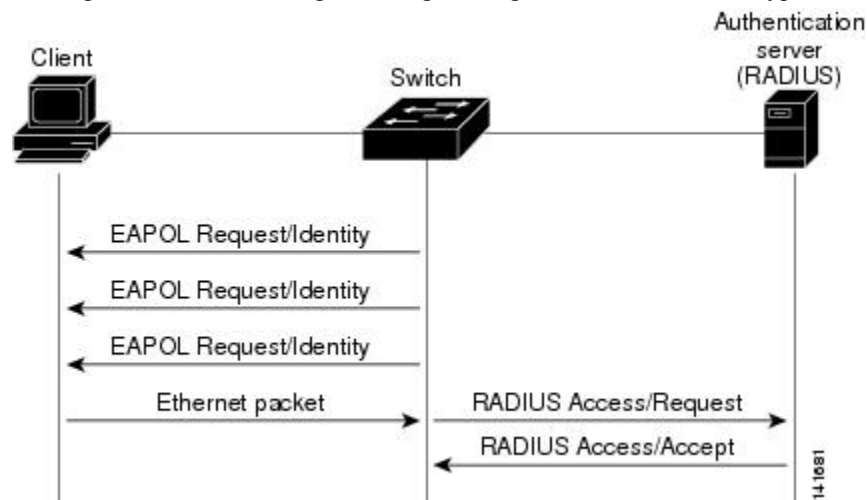
Authentication Manager for Port-Based Authentication

Port-Based Authentication Methods

Table 93: 802.1x Features

Authentication method	Mode			
	Single host	Multiple host	MDA	Multiple Authentication
802.1x	VLAN assignment Per-user ACL Filter-ID attribute Downloadable ACL Redirect URL	VLAN assignment	VLAN assignment Per-user ACL Filter-Id attribute Downloadable ACL Redirect URL	VLAN assignment 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	VLAN assignment Per-user ACL Filter-Id attribute Downloadable ACL Redirect URL	VLAN assignment Per-user ACL Filter-Id attribute Downloadable ACL Redirect URL
Standalone web authentication	Proxy ACL, Filter-Id attribute, downloadable ACL			
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 Downloadable ACL Redirect URL
Web authentication as fallback method	Proxy ACL Filter-Id attribute Downloadable ACL	Proxy ACL Filter-Id attribute Downloadable ACL	Proxy ACL Filter-Id attribute Downloadable ACL	Proxy ACL Filter-Id attribute Downloadable ACL

[15] Supported in Cisco IOS Release 12.2(50)SE and later.
[16] For clients that do not support 802.1x authentication.

Per-User ACLs and Filter-Ids

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 multi-host 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.

Port-Based 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** keyword. For example, the **authentication port-control auto** interface configuration command enables authentication on an interface. However, the **dot1x system-authentication control** global 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.

The **authentication manager** commands provide the same functionality as earlier 802.1x commands.

When filtering 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

Table 94: Authentication Manager Commands and Earlier 802.1x Commands

The authentication manager commands in Cisco IOS Release 12.2(50)SE or later	The equivalent 802.1x commands in Cisco IOS Release 12.2(46)SE and earlier	Description
authentication control-direction {**both** | **in**}	**dot1x control-direction** {**both** | **in**}	Enable 802.1x authentication with the wake-on-LAN (WoL) feature, and configure the port control as unidirectional or bidirectional.
authentication event	**dot1x auth-fail vlan** **dot1x critical (interface configuration)** **dot1x guest-vlan6**	Enable the restricted VLAN on a port. Enable the inaccessible-authentication-bypass feature. Specify an active VLAN as an 802.1x guest VLAN.
authentication fallback *fallback-profile*	**dot1x fallback** *fallback-profile*	Configure a port to use web authentication as a fallback method for clients that do not support 802.1x authentication.
authentication host-mode [**multi-auth** | **multi-domain** | **multi-host** | **single-host**]	**dot1x host-mode** {**single-host** | **multi-host** | **multi-domain**}	Allow a single host (client) or multiple hosts on an 802.1x-authorized port.
authentication order	**mab**	Provides the flexibility to define the order of authentication methods to be used.
authentication periodic	**dot1x reauthentication**	Enable periodic re-authentication of the client.
authentication port-control {**auto** | **force-authorized** | **force-un authorized**}	**dot1x port-control** {**auto** | **force-authorized** | **force-unauthorized**}	Enable manual control of the authorization state of the port.
authentication timer	**dot1x timeout**	Set the 802.1x timers.
authentication violation {**protect** | **restrict** | **shutdown**}	**dot1x violation-mode** {**shutdown** | **restrict** | **protect**}	Configure the violation modes that occur when a new device connects to a port or when a new device connects to a port after the maximum number of devices are connected to that port.

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.

Note   CDP bypass is not supported and may cause a port go into err-disabled state.

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.

Port-Based Authentication and Switch Stacks

If a switch is added to or removed from a switch stack, 802.1x authentication is not affected as long as the IP connectivity between the RADIUS server and the stack remains intact. This statement also applies if the stack's active switch is removed from the switch stack. Note that if the active switch fails, a stack member becomes the new active switch of the stack by using the election process, and the 802.1x authentication process continues as usual.

If IP connectivity to the RADIUS server is interrupted because the switch that was connected to the server is removed or fails, these events occur:

- Ports that are already authenticated and that do not have periodic re-authentication enabled remain in the authenticated state. Communication with the RADIUS server is not required.

- Ports that are already authenticated and that have periodic re-authentication enabled (with the **dot1x re-authentication** global configuration command) fail the authentication process when the re-authentication occurs. Ports return to the unauthenticated state during the re-authentication process. Communication with the RADIUS server is required.

  For an ongoing authentication, the authentication fails immediately because there is no server connectivity.

If the switch that failed comes up and rejoins the switch stack, the authentications might or might not fail depending on the boot-up time and whether the connectivity to the RADIUS server is re-established by the time the authentication is attempted.

To avoid loss of connectivity to the RADIUS server, you should ensure that there is a redundant connection to it. For example, you can have a redundant connection to the stack's active switch and another to a stack member, and if the active switch fails, the switch stack still has connectivity to the RADIUS server.

802.1x Host Mode

You can configure an 802.1x port for single-host or for multiple-hosts mode. In single-host mode, 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. 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 (re-authentication 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 90: Multiple Host Mode Example
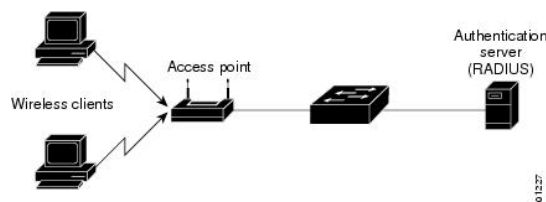
> ✎
>
> **Note** For all host modes, the line protocol stays up before authorization when port-based authentication is configured.

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.

802.1x Multiple Authentication Mode

Multiple-authentication (multiauth) mode allows multiple authenticated clients on the data VLAN and voice VLAN. Each host is individually authenticated. There is no limit to the number of data or voice device that can be authenticated on a multiauthport.

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.

> ✎
>
> **Note** When a port is in multiple-authentication mode, the authentication-failed VLAN features do not activate.

You can assign a RADIUS-server-supplied VLAN in multi-auth mode, under the following conditions:

- The host is the first host authorized on the port, and the RADIUS server supplies VLAN information

- Subsequent hosts are authorized with a VLAN that matches the operational VLAN.

- A host is authorized on the port with no VLAN assignment, and subsequent hosts either have no VLAN assignment, or their VLAN information matches the operational VLAN.

- The first host authorized on the port has a group VLAN assignment, and subsequent hosts either have no VLAN assignment, or their group VLAN matches the group VLAN on the port. Subsequent hosts must use the same VLAN from the VLAN group as the first host. If a VLAN list is used, all hosts are subject to the conditions specified in the VLAN list.

- After a VLAN is assigned to a host on the port, subsequent hosts must have matching VLAN information or be denied access to the port.

- You cannot configure a guest VLAN or an auth-fail VLAN in multi-auth mode.

- The behavior of the critical-auth VLAN is not changed for multi-auth mode. When a host tries to authenticate and the server is not reachable, all authorized hosts are reinitialized in the configured VLAN.

Multi-auth Per User VLAN assignment

The Multi-auth Per User VLAN assignment feature allows you to create multiple operational access VLANs based on VLANs assigned to the clients on the port that has a single configured access VLAN. The port configured as an access port where the traffic for all the VLANs associated with data domain is not dot1q tagged, and these VLANs are treated as native VLANs.

The number of hosts per multi-auth port is 8, however there can be more hosts.

The following scenarios are associated with the multi-auth Per User VLAN assignments:

Scenario one

When a hub is connected to an access port, and the port is configured with an access VLAN (V0).

The host (H1) is assigned to VLAN (V1) through the hub. The operational VLAN of the port is changed to V1. This behaviour is similar on a single-host or multi-domain-auth port.

When a second host (H2) is connected and gets assigned to VLAN (V2), the port will have two operational VLANs (V1 and V2). If H1 and H2 sends untagged ingress traffic, H1 traffic is mapped to VLAN (V1) and H2 traffic to VLAN (V2), all egress traffic going out of the port on VLAN (V1) and VLAN (V2) are untagged.

If both the hosts, H1 and H2 are logged out or the sessions are removed due to some reason then VLAN (V1) and VLAN (V2) are removed from the port, and the configured VLAN (V0) is restored on the port.

Scenario two

When a hub is connected to an access port, and the port is configured with an access VLAN (V0). The host (H1) is assigned to VLAN (V1) through the hub. The operational VLAN of the port is changed to V1.

When a second host (H2) is connected and gets authorized without explicit vlan policy, H2 is expected to use the configured VLAN (V0) that is restored on the port. A ll egress traffic going out of two operational VLANs, VLAN (V0) and VLAN (V1) are untagged.

If host (H2) is logged out or the session is removed due to some reason then the configured VLAN (V0) is removed from the port, and VLAN (V1) becomes the only operational VLAN on the port.

Scenario three

When a hub is connected to an access port in open mode, and the port is configured with an access VLAN (V0) .

The host (H1) is assigned to VLAN (V1) through the hub. The operational VLAN of the port is changed to V1. When a second host (H2) is connected and remains unauthorized, it still has access to operational VLAN (V1) due to open mode.

If host H1 is logged out or the session is removed due to some reason, VLAN (V1) is removed from the port and host (H2) gets assigned to VLAN (V0).

Note The combination of Open mode and VLAN assignment has an adverse affect on host (H2) because it has an IP address in the subnet that corresponds to VLAN (V1).

Limitation in Multi-auth Per User VLAN assignment

In the Multi-auth Per User VLAN assignment feature, egress traffic from multiple vlans are untagged on a port where the hosts receive traffic that is not meant for them. This can be a problem with broadcast and multicast traffic.

- **IPv4 ARPs**: Hosts receive ARP packets from other subnets. This is a problem if two subnets in different Virtual Routing and Forwarding (VRF) tables with overlapping IP address range are active on the port. The host ARP cache may get invalid entries.

- I**Pv6 control packets**: In IPv6 deployments, Router Advertisements (RA) are processed by hosts that are not supposed to receive them. When a host from one VLAN receives RA from a different VLAN, the host assign incorrect IPv6 address to itself. Such a host is unable to get access to the network.

  The workaround is to enable the IPv6 first hop security so that the broadcast ICMPv6 packets are converted to unicast and sent out from multi-auth enabled ports.. The packet is replicated for each client in multi-auth

port belonging to the VLAN and the destination MAC is set to an individual client. Ports having one VLAN, ICMPv6 packets broadcast normally.

- **IP multicast**: Multicast traffic destined to a multicast group gets replicated for different VLANs if the hosts on those VLANs join the multicast group. When two hosts in different VLANs join a multicast group (on the same mutli-auth port), two copies of each multicast packet are sent out from that port.

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.

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 multi-auth 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 multi-domain 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.

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.

- Re-authentication successfully occurs.

- Re-authentication 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

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 this command, see the *Cisco IOS Debug Command Reference, Release 12.4.*

This table lists the AV pairs and when they are sent are sent by the switch.

Table 95: Accounting AV Pairs

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[17]	Sometimes
Attribute[25]	Class	Always	Always	Always

Attribute Number	AV Pair Name	START	INTERIM	STOP
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

[17] 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.

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.

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 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 fail-over backup to the first one. The RADIUS host entries are tried in the order that they were configured.

802.1x Authentication with VLAN Assignment

The switch supports 802.1x authentication with VLAN assignment. After successful 802.1x authentication of a port, 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.

Voice device authentication is supported with multidomain host mode in Cisco IOS Release 12.2(37)SE. In Cisco IOS Release 12.2(40)SE and later, when a voice device is authorized and the RADIUS server returned 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.

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 multidomain 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.

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 or VLAN ID

    - [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 IEEE 802.1x-authenticated user.

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.

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.

To configure per-user ACLs:

- 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.

Note Per-user ACLs are supported only in single-host mode.

802.1x Authentication with Downloadable ACLs and Redirect URLs

Note IPv6 does not support 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 limit for dACL with stacking is 64 ACEs per dACL per port. The limit without stacking is the number of available TCAM entries which varies based on the other ACL features that are active.

If there is no static ACL on a port, a dynamic auth-default ACL is created, and policies are enforced before dACLs are downloaded and applied.

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 for IPv4 by using the **ip access-list extended auth-default-acl** command in global configuration mode. For IPv6, use the **ipv6 access-list extended auth-default-acl** command in the global configuration mode.

✎

Note The auth-default-ACL does not support Cisco Discovery Protocol bypass in the single host mode. You must configure a static ACL on the interface to support Cisco Discovery Protocol 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 or 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. The switch then forwards the client web browser to the specified redirect address. The url-redirect AV 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.

> **Note**
> - Traffic that matches a permit ACE in the ACL is redirected.
>
> - 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 (AV) 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 for IPv4 and #ACL#-.in.ipv6 attribute for IPv6.

- 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.

VLAN ID-based MAC Authentication

You can use VLAN ID-based MAC authentication if you wish 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.)

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 IEEE 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 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 IEEE 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 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.

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.

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, multi-auth and 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 IEEE 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.

802.1x Authentication with Restricted VLAN

You can configure a restricted VLAN (also referred to as an *authentication failed VLAN*) for each IEEE 802.1x port on a switch stack or 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 re-authentication attempt. A port in the restricted VLAN tries to re-authenticate at configured intervals (the default is 60 seconds). If re-authentication fails, the port remains in the restricted VLAN. If re-authentication is successful, the port moves either to the configured VLAN or to a VLAN sent by the RADIUS server. You can disable re-authentication. If you do this, the only way to restart the authentication process is for the port to receive a *link down* or *EAP logoff* event. We recommend that you keep re-authentication 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 on 802.1x ports in all host modes 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 port features such as dynamic ARP Inspection, DHCP snooping, and IP source guard can be configured independently on a restricted VLAN.

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.

Note   If *critical authentication* is configured on interface, then vlan used for critical authorization (*critical vlan*) should be active on the switch. If the *critical vlan* is inactive (or) down, *critical authentication* session will keep trying to enable inactive vlan and fail repeatedly. This can lead to large amount of memory holding.

Inaccessible Authentication Bypass 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 modes.

Inaccessible Authentication Bypass 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 re-authenticated.

Inaccessible Authentication Bypass 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.

In a switch stack:

- The stack's active switch checks the status of the RADIUS servers by sending keepalive packets. When the status of a RADIUS server changes, the stack's active switch sends the information to the stack members. The stack members can then check the status of RADIUS servers when re-authenticating critical ports.

- If the new active switch is elected, the link between the switch stack and RADIUS server might change, and the new stack immediately sends keepalive packets to update the status of the RADIUS servers. If the server status changes from *dead* to *alive*, the switch re-authenticates all switch ports in the critical-authentication state.

When a member is added to the stack, the stack's active switch sends the member the server status.

802.1x Critical Voice VLAN

When an IP phone connected to a port is authenticated by the access control server (ACS), the phone is put into the voice domain. If the ACS is not reachable, the switch cannot determine if the device is a voice device. If the server is unavailable, the phone cannot access the voice network and therefore cannot operate.

For data traffic, you can configure inaccessible authentication bypass, or critical authentication, to allow traffic to pass through on the native VLAN when the server is not available. If the RADIUS authentication server is unavailable (down) and inaccessible authentication bypass is enabled, the switch grants the client access to the network and puts the port in the critical-authentication state in the RADIUS-configured or the user-specified access VLAN. When the switch cannot reach the configured RADIUS servers and new hosts cannot be authenticated, the switch connects those hosts to critical ports. A new host trying to connect to the critical port is moved to a user-specified access VLAN, the critical VLAN, and granted limited authentication.

You can enter the **authentication event server dead action authorize voice** interface configuration command to configure the critical voice VLAN feature. When the ACS does not respond, the port goes into critical authentication mode. When traffic coming from the host is tagged with the voice VLAN, the connected device (the phone) is put in the configured voice VLAN for the port. The IP phones learn the voice VLAN identification through CDP (Cisco devices) or through LLDP or DHCP.

You can configure the voice VLAN for a port by entering the **switchport voice vlan** *vlan-id* interface configuration command.

This feature is supported in multidomain and multi-auth host modes. Although you can enter the command when the switch in single-host or multi-host mode, the command has no effect unless the device changes to multidomain or multi-auth host mode.

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.

IEEE 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 IEEE 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 IEEE 802.1x authentication is enabled on a voice VLAN port, the switch drops packets from unrecognized IP phones more than one hop away.

When IEEE 802.1x authentication is enabled on a switch port, you can configure an access port VLAN that is also a voice VLAN.

When IP phones are connected to an 802.1x-enabled switch port that is in single host mode, the switch grants the phones network access without authenticating them. We recommend that you use multidomain authentication (MDA) on the port to authenticate both a data device and a voice device, such as an IP phone

Note If you enable IEEE 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.

IEEE 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.

IEEE 802.1x Authentication with Wake-on-LAN

The IEEE 802.1x authentication with 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 IEEE 802.1x port and the host powers off, the IEEE 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 IEEE 802.1x authentication with WoL, the switch forwards traffic to unauthorized IEEE 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.

IEEE 802.1x Authentication with MAC Authentication Bypass

You can configure the switch to authorize clients based on the client MAC address by using the MAC authentication bypass feature. For example, you can enable this feature on IEEE 802.1x ports connected to devices such as printers.

If IEEE 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 IEEE 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 IEEE 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. This process works for most client devices; however, it does not work for clients that use an alternate MAC address format. You can configure how MAB authentication is performed for clients with MAC addresses that deviate from the standard format or where the RADIUS configuration requires the user name and password to differ.

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 IEEE 802.1x supplicant, the switch does not unauthorize the client connected to the port. When re-authentication occurs, the switch uses the authentication or re-authentication methods configured on the port, if the previous session ended because the Termination-Action RADIUS attribute value is DEFAULT.

Clients that were authorized with MAC authentication bypass can be re-authenticated. The re-authentication process is the same as that for clients that were authenticated with IEEE 802.1x. During re-authentication, the port remains in the previously assigned VLAN. If re-authentication is successful, the switch keeps the port in the same VLAN. If re-authentication fails, the switch assigns the port to the guest VLAN, if one is configured.

If re-authentication 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 re-authentication. If MAC authentication bypass is enabled and the IEEE 802.1x authentication times out, the switch uses the MAC authentication bypass feature to initiate re-authorization. For more information about these AV pairs, see RFC 3580, "IEEE 802.1X Remote Authentication Dial In User Service (RADIUS) Usage Guidelines."

MAC authentication bypass interacts with the features:

- IEEE 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 IEEE 802.lx port is authenticated with MAC authentication bypass.

- Port security

- Voice VLAN

- VLAN Membership Policy Server (VMPS)—IEEE802.1x and VMPS are mutually exclusive.

- Private VLAN—You can assign a client to a private VLAN.

- 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.

Cisco IOS Release 12.2(55)SE and later supports filtering of verbose MAB system messages

Network Admission Control Layer 2 IEEE 802.1x Validation

The switch supports the Network Admission Control (NAC) Layer 2 IEEE 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 IEEE 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 re-authentication 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 re-authenticate 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 re-authentication 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 IEEE 802.1x validation is similar to configuring IEEE 802.1x port-based authentication except that you must configure a posture token on the RADIUS server.

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.

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.

> **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.

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.

✎

Note   For all host modes, the line protocol stays up before authorization when port-based authentication is configured.

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:

- You must configure a switch port for MDA.

- You must configure the voice VLAN for the IP phone when the host mode is set to multidomain.

- Voice VLAN assignment on an MDA-enabled port is supported Cisco IOS Release 12.2(40)SE and later.

    ✎

    Note   You can assign a dynamic VLAN to a voice device on an MDA-enabled switch port, but the voice device fails authorization if a static voice VLAN configured on the switchport is the same as the dynamic VLAN assigned for the voice device in the RADIUS server.

- 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.

- You can use dynamic VLAN assignment from a RADIUS server only for data devices.

- MDA can use MAC authentication bypass as a fallback mechanism to allow the switch port to connect to devices that do not support IEEE 802.1x authentication.

- 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 is changed from single- or multihost to multidomain mode, an authorized data device remains authorized on the port. However, a Cisco IP phone that has been allowed 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- or multihost mode to multidomain mode.

- Switching a port host mode from multidomain to single- or multihost mode removes all authorized devices from the port.

- If a data domain is authorized first and placed in the guest VLAN, non-IEEE 802.1x-capable voice devices need to tag their packets on the voice VLAN to trigger authentication.

- 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 voice and data VLANs of the port. If used, only one device on the port should enforce per-user ACLs.

Limiting Login for Users

The Limiting Login feature helps Network administrators to limit the login attempt of users to a network. When a user fails to successfully login to a network within a configurable number of attempts within a configurable time limit, the user can be blocked. This feature is enabled only for local users and not for remote users. You need to configure the **aaa authentication rejected** command in global configuration mode to enable this feature.

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.

- 802.1x switch supplicant: 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.

> **Note** NEAT configuration is the only supported and qualified method to authenticate switches using 802.1x. Any other method to authenticate a network switch can result in an undefined behavior.

- If the access VLAN is configured on the authenticator switch, it becomes the native VLAN for the trunk port after successful authentication.

In the default state, when you connect a supplicant switch to an authenticator switch that has BPDU guard enabled, the authenticator port could be error-disabled if it receives a Spanning Tree Protocol (STP) bridge protocol data unit (BPDU) packets before the supplicant switch has authenticated. Beginning with Cisco IOS Release 15.0(1)SE, you can control traffic exiting the supplicant port during the authentication period. Entering the **dot1x supplicant controlled transient** global configuration command temporarily blocks the supplicant

port during authentication to ensure that the authenticator port does not shut down before authentication completes. If authentication fails, the supplicant port opens. Entering the **no dot1x supplicant controlled transient** global configuration command opens the supplicant port during the authentication period. This is the default behavior.

We strongly recommend using the **dot1x supplicant controlled transient**command on a supplicant switch when BPDU guard is enabled on the authenticator switch port with the **spanning-tree bpduguard enable** interface configuration command.

> **Note** If you globally enable BPDU guard on the authenticator switch by using the **spanning-tree portfast bpduguard default** global configuration command, entering the **dot1x supplicant controlled transient** command does not prevent the BPDU violation.

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.

- 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 91: Authenticator and Supplicant Switch using CISP
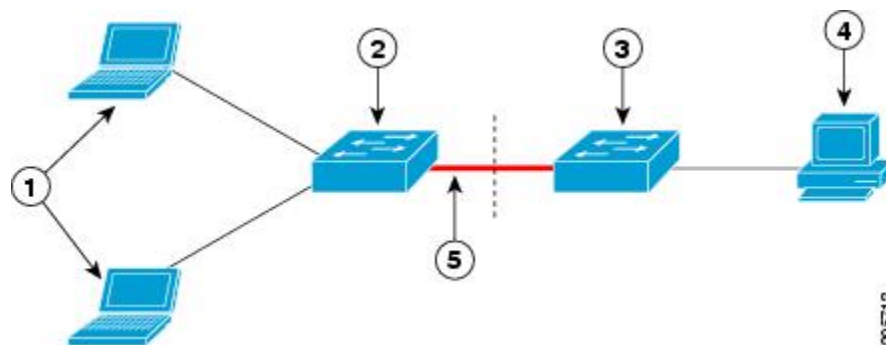
1	Workstations (clients)	2	Supplicant switch (outside wiring closet)
3	Authenticator switch	4	Access control server (ACS)
5	Trunk port		

> ✎
>
> **Note** The **switchport nonegotiate** command is not supported on supplicant and authenticator switches with NEAT. This command should not be configured at the supplicant side of the topology. If configured on the authenticator side, the internal macros will automatically remove this command from the port.

Voice Aware 802.1x Security

> ✎
>
> **Note** To use voice aware IEEE 802.1x authentication, the switch must be running the LAN base image.

You use the voice aware 802.1x security feature to configure the switch to disable only the VLAN on which a security violation occurs, whether it is a data or voice VLAN. In previous releases, when an attempt to authenticate the data client caused a security violation, the entire port shut down, resulting in a complete loss of connectivity.

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.

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)

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 also160000050000000B288508E5:

```
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.

How to Configure 802.1x Port-Based Authentication

Default 802.1x Authentication Configuration

Table 96: Default 802.1x Authentication Configuration

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.

RADIUS server
  • IP address
  • UDP authentication port
  • Key
	• None specified.
	• 1812.
	• None specified.

Host mode	Single-host mode.
Control direction	Bidirectional control.
Periodic re-authentication	Disabled.
Number of seconds between re-authentication attempts	3600 seconds.
Re-authentication 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).

Feature	Default Setting
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 dot1x timeout server-timeout 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 Authentication Configuration Guidelines

802.1x Authentication

These are the 802.1x authentication configuration guidelines:

- When 802.1x authentication is enabled, ports are authenticated before any other Layer 2 or Layer 3 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 re-authentication.

  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, voice VLAN ports, and Layer 3 routed ports, but it is not supported on these port types:

  - 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.

- Cisco IOS Release 12.2(55)SE and later supports filtering of system messages related to 802.1x authentication.

VLAN Assignment, Guest VLAN, Restricted VLAN, and Inaccessible Authentication Bypass

These are the configuration guidelines for VLAN assignment, guest VLAN, restricted VLAN, and inaccessible authentication bypass:

- When 802.1x authentication is enabled on a port, you cannot configure a port VLAN that is equal to a voice VLAN.

- You can configure any VLAN except an RSPAN 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 re-initiate 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 re-authenticate 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.

- If the CTS links are in Critical Authentication mode and the active switch reloads, the policy where SGT was configured on a device will not be available on the new active switch. This is because the internal bindings will not be synced to the standby switch in a 3750-X switch stack.

- 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.

- When wireless guest clients obtains IP from foreign client VLAN instead of anchor client VLAN, you should use the **ip dhcp required** command under the WLAN configuration to force clients to issue a new DHCP request. This prevents the clients from getting an incorrect IP at anchor.

- If the wired guest clients fail to get IP address after a Cisco WLC (foreign) reload, perform a shut/no shut on the ports used by the clients to reconnect them.

MAC Authentication Bypass

These are the MAC authentication bypass configuration guidelines:

- Unless otherwise stated, the MAC authentication bypass guidelines are the same as the 802.1x authentication guidelines.

- 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 re-authorize the port.

- If the port is in the authorized state, the port remains in this state until re-authorization 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

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 multihost 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.

Configuring 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.

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**.

Follow these steps to enable the 802.1x readiness check on the switch:

Before you begin

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.
- 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
- 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.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **dot1x test eapol-capable** [**interface** *interface-id*]
4. **dot1x test timeout** *timeout*
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode.
	Example:	• Enter your password if prompted.
	Switch> **enable**	

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**dot1x test eapol-capable** [**interface** *interface-id*] **Example:** Switch# **dot1x test eapol-capable interface gigabitethernet1/0/13** DOT1X_PORT_EAPOL_CAPABLE:DOT1X: MAC 00-01-02-4b-f1-a3 on gigabitethernet1/0/13 is EAPOL capable	Enables the 802.1x readiness check on the switch. (Optional) For *interface-id* specify the port on which to check for IEEE 802.1x readiness. **Note** If you omit the optional **interface** keyword, all interfaces on the switch are tested.
Step 4	**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.
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Voice Aware 802.1x Security

Note   To use voice aware IEEE 802.1x authentication, the switch must be running the LAN base image.

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 re-enabled. If error-disabled recovery is not configured for the port, you re-enable it by using the **shutdown** and **no shutdown** interface configuration commands.

- You can re-enable 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.

Beginning in privileged EXEC mode, follow these steps to enable voice aware 802.1x security:

SUMMARY STEPS

1. **configure terminal**
2. **errdisable detect cause security-violation shutdown vlan**
3. **errdisable recovery cause security-violation**
4. **clear errdisable interface***interface-id* **vlan** *[vlan-list]*
5. Enter the following:

   - **shutdown**
   - **no shutdown**

6. **end**
7. **show errdisable detect**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**errdisable detect cause security-violation shutdown vlan**	Shut 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.
Step 3	**errdisable recovery cause security-violation**	Enter global configuration mode.
Step 4	**clear errdisable interface***interface-id* **vlan** *[vlan-list]*	(Optional) Reenable individual VLANs that have been error disabled.
		- For interface-id specify the port on which to reenable individual VLANs.

	Command or Action	Purpose
		• (Optional) For vlan-list specify a list of VLANs to be re-enabled. If vlan-list is not specified, all VLANs are re-enabled.
Step 5	Enter the following: • **shutdown** • **no shutdown**	(Optional) Re-enable an error-disabled VLAN, and clear all error-disable indications.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show errdisable detect**	Verify your entries.

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 re-enable all VLANs that were error disabled on port Gigabit Ethernet 40/2.

```
Switch# clear errdisable interface gigabitethernet4/0/2
vlan
```

You can verify your settings by entering the **show errdisable detect** privileged EXEC command.

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

Beginning in privileged EXEC mode, follow these steps to configure the security violation actions on the switch:

SUMMARY STEPS

1. **configure terminal**
2. **aaa new-model**
3. **aaa authentication dot1x** {**default**} *method1*
4. **interface** *interface-id*
5. **switchport mode access**
6. **authentication violation** {**shutdown** | **restrict** | **protect** | **replace**}
7. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**aaa new-model** **Example:** Switch(config)# **aaa new-model**	Enables AAA.
Step 3	**aaa authentication dot1x** {**default**} *method1* **Example:** Switch(config)# **aaa authentication dot1x default group radius**	Creates an 802.1x authentication method list. To create a default list that is used 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. 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.
Step 4	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/4**	Specifies the port connected to the client that is to be enabled for IEEE 802.1x authentication, and enter interface configuration mode.
Step 5	**switchport mode access** **Example:** Switch(config-if)# **switchport mode access**	Sets the port to access mode.
Step 6	**authentication violation** {**shutdown** | **restrict** | **protect** | **replace**} **Example:** Switch(config-if)# **authentication violation restrict**	Configures the violation mode. The keywords have these meanings: • **shutdown**–Error disable the port. • **restrict**–Generate a syslog error. • **protect**–Drop packets from any new device that sends traffic to the port. • **replace**–Removes the current session and authenticates with the new host.

	Command or Action	Purpose
Step 7	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Configuring 802.1x Authentication

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 process:

Before you begin

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.

SUMMARY STEPS

1. A user connects to a port on the switch.
2. Authentication is performed.
3. VLAN assignment is enabled, as appropriate, based on the RADIUS server configuration.
4. The switch sends a start message to an accounting server.
5. Re-authentication is performed, as necessary.
6. The switch sends an interim accounting update to the accounting server that is based on the result of re-authentication.
7. The user disconnects from the port.
8. The switch sends a stop message to the accounting server.

DETAILED STEPS

	Command or Action	Purpose
Step 1	A user connects to a port on the switch.	
Step 2	Authentication is performed.	
Step 3	VLAN assignment is enabled, as appropriate, based on the RADIUS server configuration.	
Step 4	The switch sends a start message to an accounting server.	
Step 5	Re-authentication is performed, as necessary.	
Step 6	The switch sends an interim accounting update to the accounting server that is based on the result of re-authentication.	
Step 7	The user disconnects from the port.	

	Command or Action	**Purpose**
Step 8	The switch sends a stop message to the accounting server.	

Configuring 802.1x Port-Based Authentication

Beginning in privileged EXEC mode, follow these steps to configure 802.1x port-based authentication:

SUMMARY STEPS

1. **configure terminal**
2. **aaa new-model**
3. **aaa authentication dot1x** {**default**} *method1*
4. **dot1x system-auth-control**
5. **aaa authorization network** {**default**} **group radius**
6. **radius-server host** *ip-address*
7. **radius-server key** *string*
8. **interface** *interface-i*d
9. **switchport mode access**
10. **authentication port-control auto**
11. **dot1x pae authenticator**
12. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**aaa new-model** **Example:** Switch(config)# **aaa new-model**	Enables AAA.
Step 3	**aaa authentication dot1x** {**default**} *method1* **Example:** Switch(config)# **aaa authentication dot1x default group radius**	Creates an 802.1x authentication method list. To create a default list that is used 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. For *method1*, enter the **group radius** keywords to use the list of all RADIUS servers for authentication.

	Command or Action	Purpose
		Note Though other keywords are visible in the command-line help string, only the **group radius** keywords are supported.
Step 4	**dot1x system-auth-control** **Example:** Switch(config)# **dot1x system-auth-control**	Enables 802.1x authentication globally on the switch.
Step 5	**aaa authorization network** {**default**} **group radius** **Example:** Switch(config)# **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. **Note** For per-user ACLs, single-host mode must be configured. This setting is the default.
Step 6	**radius-server host** *ip-address* **Example:** Switch(config)# **radius-server host 124.2.2.12**	(Optional) Specifies the IP address of the RADIUS server.
Step 7	**radius-server key** *string* **Example:** Switch(config)# **radius-server key abc1234**	(Optional) Specifies the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server.
Step 8	**interface** *interface-i*d **Example:** Switch(config)# **interface gigabitethernet1/0/2**	Specifies the port connected to the client that is to be enabled for IEEE 802.1x authentication, and enter interface configuration mode.
Step 9	**switchport mode access** **Example:** Switch(config-if)# **switchport mode access**	(Optional) Sets the port to access mode only if you configured the RADIUS server in Step 6 and Step 7.
Step 10	**authentication port-control auto** **Example:** Switch(config-if)# **authentication port-control auto**	Enables 802.1x authentication on the port.

	Command or Action	**Purpose**
Step 11	**dot1x pae authenticator** **Example:** Switch(config-if)# **dot1x pae authenticator**	Sets the interface Port Access Entity to act only as an authenticator and ignore messages meant for a supplicant.
Step 12	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.

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**, the **radius-server retransmit**, and the **radius-server key** global configuration commands.

You also need to configure some settings on the RADIUS server. These settings include the IP address of the switch and the key string to be shared by both the server and the switch. For more information, see the RADIUS server documentation.

Follow these steps to configure the RADIUS server parameters on the switch. This procedure is required.

Before you begin

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.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **radius-server host** {*hostname* | *ip-address*} **auth-port** *port-number* **key** *string*
4. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:**	Enters global configuration mode.

	Command or Action	Purpose
	Switch# **configure terminal**	
Step 3	**radius-server host** {*hostname* | *ip-address*} **auth-port** *port-number* **key** *string* **Example:** Switch(config)# **radius-server host 125.5.5.43 auth-port 1812 key string**	Configures the RADIUS server parameters. For *hostname* | *ip-address,* specify the hostname or IP address of the remote RADIUS server. For **auth-port** *port-number*, specify the UDP destination port for authentication requests. The default is 1812. The range is 0 to 65536. For **key** *string*, specify 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, re-enter this command.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Configuring the Host Mode

Beginning in privileged EXEC mode, follow these steps to allow multiple hosts (clients) on an IEEE 802.1x-authorized port that has the **authentication port-control** interface configuration command set to **auto**. Use the **multi-domain** keyword to configure and enable multidomain authentication (MDA), which allows both a host and a voice device, such as an IP phone (Cisco or non-Cisco), on the same switch port. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication host-mode** [**multi-auth** \| **multi-domain** \| **multi-host** \| **single-host**]
4. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet2/0/1**	Specifies the port to which multiple hosts are indirectly attached, and enter interface configuration mode.

Step 3	**authentication host-mode** [**multi-auth** | **multi-domain** | **multi-host** | **single-host**]
	
	Example:
	
	Switch(config-if)# **authentication host-mode multi-host**
		Allows multiple hosts (clients) on an 802.1x-authorized port.
		
		The keywords have these meanings:
		
		• **multi-auth**–Allow multiple authenticated clients on both the voice VLAN and data VLAN.
		
		  Note The **multi-auth** keyword is only available with the **authentication host-mode** command.
		
		• **multi-host**–Allow multiple hosts on an 802.1x-authorized port after a single host has been authenticated.
		
		• **multi-domain**–Allow both a host and a voice device, such as an IP phone (Cisco or non-Cisco), to be authenticated on an IEEE 802.1x-authorized port.
		
		  Note You must configure the voice VLAN for the IP phone when the host mode is set to **multi-domain**.
		
		Make sure that the **authentication port-control** interface configuration command is set to **auto** for the specified interface.

Step 4	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.

Configuring Periodic Re-Authentication

You can enable periodic 802.1x client re-authentication and specify how often it occurs. If you do not specify a time period before enabling re-authentication, the number of seconds between attempts is 3600.

Beginning in privileged EXEC mode, follow these steps to enable periodic re-authentication of the client and to configure the number of seconds between re-authentication attempts. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication periodic**
4. **authentication timer** {{[**inactivity** | **reauthenticate** | **restart**]} {*value*}}
5. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet2/0/1**	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**authentication periodic** **Example:** Switch(config-if)# **authentication periodic**	Enables periodic re-authentication 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.
Step 4	**authentication timer** {{[**inactivity**	**reauthenticate**	**restart**]} {*value*}} **Example:** Switch(config-if)# **authentication timer reauthenticate 180**	Sets the number of seconds between re-authentication attempts. The **authentication timer** keywords have these meanings: • **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 re-authentication attempt is initiated

	Command or Action	**Purpose**
		• **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 re-authentication is enabled.
Step 5	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Changing the Quiet Period

When the switch cannot authenticate the client, the switch remains idle for a set period of time and then tries again. The **authentication timer restart** interface configuration command controls the idle period. A failed authentication of the client might occur because the client provided an invalid password. You can provide a faster response time to the user by entering a number smaller than the default.

Beginning in privileged EXEC mode, follow these steps to change the quiet period. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication timer restart** *seconds*
4. **end**
5. **show authentication sessions interface** *interface-id*
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet2/0/1`	Specifies the port to be configured, and enter interface configuration mode.

	Command or Action	Purpose
Step 3	**authentication timer restart** *seconds* **Example:** Switch(config-if)# **authentication timer restart 30**	Sets the number of seconds that the switch remains in the quiet state following a failed authentication exchange with the client. The range is 1 to 65535 seconds; the default is 60.
Step 4	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 5	**show authentication sessions interface** *interface-id* **Example:** Switch# **show authentication sessions interface gigabitethernet2/0/1**	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Changing the Switch-to-Client Retransmission Time

The client responds to the EAP-request/identity frame from the switch with an EAP-response/identity frame. If the switch does not receive this response, it waits a set period of time (known as the retransmission time) and then resends the frame.

✎

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.

Beginning in privileged EXEC mode, follow these steps to change the amount of time that the switch waits for client notification. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication timer reauthenticate** *seconds*
4. **end**
5. **show authentication sessions interface** *interface-id*
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet2/0/1**	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**authentication timer reauthenticate** *seconds* **Example:** Switch(config-if)# **authentication timer reauthenticate 60**	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.
Step 4	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 5	**show authentication sessions interface** *interface-id* **Example:** Switch# **show authentication sessions interface gigabitethernet2/0/1**	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Setting the Switch-to-Client Frame-Retransmission Number

In addition to changing the switch-to-client retransmission time, you can change the number of times that the switch sends an EAP-request/identity frame (assuming no response is received) to the client before restarting the authentication process.

| **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. |

Beginning in privileged EXEC mode, follow these steps to set the switch-to-client frame-retransmission number. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **dot1x max-reauth-req** *count*
4. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet2/0/1**	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**dot1x max-reauth-req** *count* **Example:** Switch(config-if)# **dot1x max-reauth-req 5**	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.
Step 4	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.

Setting the Re-Authentication Number

You can also change the number of times that the switch restarts the authentication process before the port changes to the unauthorized state.

> ✎
>
> | **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. |

Beginning in privileged EXEC mode, follow these steps to set the re-authentication number. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode access**
4. **dot1x max-req** *count*
5. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** Switch# **interface gigabitethernet2/0/1**	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**switchport mode access** **Example:** Switch(config-if)# **switchport mode access**	Sets the port to access mode only if you previously configured the RADIUS server.
Step 4	**dot1x max-req** *count* **Example:** Switch(config-if)# **dot1x max-req 4**	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.
Step 5	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.

Enabling MAC Move

MAC move allows an authenticated host to move from one port on the switch to another.

Beginning in privileged EXEC mode, follow these steps to globally enable MAC move on the switch. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **authentication mac-move permit**
3. **end**
4. **show running-config**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**authentication mac-move permit** **Example:** Switch(config)# **authentication mac-move permit**	Enables MAC move on the switch. Default is deny. In Session Aware Networking mode, the default CLI is **access-session mac-move deny**. To enable Mac Move in Session Aware Networking, use the **no access-session mac-move** global configuration command.
Step 3	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 4	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 5	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Disabling MAC Move

To disable MAC move from a secure port to an unsecured port on a switch, beginning in privileged EXEC mode, follow these steps. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **authentication mac-move deny-uncontrolled**
3. **end**
4. **show running-config**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**authentication mac-move deny-uncontrolled** **Example:** Switch(config)# **authentication mac-move deny-uncontrolled**	Disables MAC move on the switch.
Step 3	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 4	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 5	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Enabling MAC Replace

MAC replace allows a host to replace an authenticated host on a port.

Beginning in privileged EXEC mode, follow these steps to enable MAC replace on an interface. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication violation** {**protect** | **replace** | **restrict** | **shutdown**}
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet2/0/2**	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**authentication violation** {**protect**	**replace**	**restrict**	**shutdown**} **Example:** Switch(config-if)# **authentication violation replace**	Use the **replace** keyword to enable 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**: the port drops packets with unexpected MAC addresses without generating a system message. • **restrict**: violating packets are dropped by the CPU and a system message is generated. • **shutdown**: the port is error disabled when it receives an unexpected MAC address.
Step 4	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.

	Command or Action	**Purpose**
Step 5	**show running-config** Example: Switch# **show running-config**	Verifies your entries.
Step 6	**copy running-config startup-config** Example: Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring 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.

Beginning in privileged EXEC mode, follow these steps to configure 802.1x accounting after AAA is enabled on your switch. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **aaa accounting dot1x default start-stop group radius**
4. **aaa accounting system default start-stop group radius**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/3**	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**aaa accounting dot1x default start-stop group radius** **Example:** Switch(config-if)# **aaa accounting dot1x default start-stop group radius**	Enables 802.1x accounting using the list of all RADIUS servers.
Step 4	**aaa accounting system default start-stop group radius** **Example:** Switch(config-if)# **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.
Step 5	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEc mode.
Step 6	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** Switch# **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.

Beginning in privileged EXEC mode, follow these steps to configure a guest VLAN. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. Use one of the following:
   - **switchport mode access**
   - **switchport mode private-vlan host**
4. **authentication event no-response action authorize vlan** *vlan-id*
5. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet2/0/2**	Specifies the port to be configured, and enter interface configuration mode.
Step 3	Use one of the following: • **switchport mode access** • **switchport mode private-vlan host** **Example:** Switch(config-if)# **switchport mode private-vlan host**	• Sets the port to access mode. • Configures the Layer 2 port as a private-VLAN host port.
Step 4	**authentication event no-response action authorize vlan** *vlan-id* **Example:** Switch(config-if)# **authentication event no-response**	Specifies an active VLAN as an 802.1x guest VLAN. The range is 1 to 4094. 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.

	Command or Action	Purpose
	`action authorize vlan 2`	
Step 5	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.

Configuring a Restricted VLAN

When you configure a restricted VLAN on a switch stack or a switch, clients that are IEEE 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.

Beginning in privileged EXEC mode, follow these steps to configure a restricted VLAN. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. Use one of the following:
   - **switchport mode access**
   - **switchport mode private-vlan host**
4. **authentication port-control auto**
5. **authentication event fail action authorize vlan** *vlan-id*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet2/0/2**	Specifies the port to be configured, and enter interface configuration mode.
Step 3	Use one of the following: • **switchport mode access**	• Sets the port to access mode.

	Command or Action	**Purpose**
	• **switchport mode private-vlan host** **Example:** Switch(config-if)# **switchport mode access**	• Configures the Layer 2 port as a private-VLAN host port.
Step 4	**authentication port-control auto** **Example:** Switch(config-if)# **authentication port-control auto**	Enables 802.1x authentication on the port.
Step 5	**authentication event fail action authorize vlan** *vlan-id* **Example:** Switch(config-if)# **authentication event fail action authorize vlan 2**	Specifies an active VLAN as an 802.1x restricted VLAN. The range is 1 to 4094. You can configure any active VLAN except an internal VLAN (routed port), an RSPAN VLAN or a voice VLAN as an 802.1x restricted VLAN.
Step 6	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.

Configuring Number of Authentication Attempts on a Restricted VLAN

You can configure the maximum number of authentication attempts allowed before a user is assigned to the restricted VLAN by using the **authentication event retry** *retry count* interface configuration command. The range of allowable authentication attempts is 1 to 3. The default is 3 attempts.

Beginning in privileged EXEC mode, follow these steps to configure the maximum number of allowed authentication attempts. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. Use one of the following:

    • **switchport mode access**
    • **switchport mode private-vlan host**

4. **authentication port-control auto**
5. **authentication event fail action authorize vlan** *vlan-id*
6. **authentication event retry** *retry count*
7. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet2/0/3**	Specifies the port to be configured, and enter interface configuration mode.
Step 3	Use one of the following: • **switchport mode access** • **switchport mode private-vlan host** **Example:** or Switch(config-if)# **switchport mode access**	• Sets the port to access mode. • Configures the Layer 2 port as a private-VLAN host port.
Step 4	**authentication port-control auto** **Example:** Switch(config-if)# **authentication port-control auto**	Enables 802.1x authentication on the port.
Step 5	**authentication event fail action authorize vlan** *vlan-id* **Example:** Switch(config-if)# **authentication event fail action authorize vlan 8**	Specifies an active VLAN as an 802.1x restricted VLAN. The range is 1 to 4094. You can configure any active VLAN except an internal VLAN (routed port), an RSPAN VLAN or a voice VLAN as an 802.1x restricted VLAN.
Step 6	**authentication event retry** *retry count* **Example:** Switch(config-if)# **authentication event retry 2**	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.
Step 7	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.

Configuring 802.1x Inaccessible Authentication Bypass with Critical Voice VLAN

Beginning in privileged EXEC mode, follow these steps to configure critical voice VLAN on a port and enable the inaccessible authentication bypass feature.

SUMMARY STEPS

1. **configure terminal**
2. **aaa new-model**
3. **radius-server dead-criteria**{**time** *seconds* } [**tries** *number*]
4. **radius-serverdeadtime***minutes*
5. **radius-server host ip-address** *address*[**acct-port** *udp-port*][**auth-port** *udp-port*] [**testusername** *name*[**idle-time** *time*] [**ignore-acct-port**][**ignore auth-port**]] [**key** *string*]
6. **dot1x critical** {**eapol** | **recovery delay** *milliseconds*}
7. **interface** *interface-id*
8. **authentication event server dead action** {**authorize** | **reinitialize**} **vlan** *vlan-id*]
9. **switchport voice vlan** *vlan-id*
10. **authentication event server dead action authorize voice**
11. **show authentication interface** *interface-id*
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**aaa new-model** **Example:** Switch(config)# **aaa new-model**	Enables AAA.
Step 3	**radius-server dead-criteria**{**time** *seconds* } [**tries** *number*] **Example:** Switch(config)# **radius-server dead-criteria time 20 tries 10**	Sets the conditions that determine when a RADIUS server is considered un-available or down (dead). • **time**— 1 to 120 seconds. The switch dynamically determines a default *seconds* value between 10 and 60. • **number**—1 to 100 tries. The switch dynamically determines a default **tries***number* between 10 and 100.

	Command or Action	**Purpose**
Step 4	**radius-serverdeadtime***minutes* **Example:** Switch(config)# **radius-server deadtime 60**	(Optional) Sets the number of minutes during which a RADIUS server is not sent requests. The range is from 0 to 1440 minutes (24 hours). The default is 0 minutes.
Step 5	**radius-server host ip-address** *address*[**acct-port** *udp-port*][**auth-port** *udp-port*] [**testusername** *name*[**idle-time** *time*] [**ignore-acct-port**][**ignore auth-port**]] [**key** *string*] **Example:** Switch(config)# **radius-server host 10.0.0.10 acct-port 1550 auth-port 1560 test username user1 idle-time 30 key abc1234**	(Optional) Configure the RADIUS server parameters by using these keywords: • **acct-port***udp-port*—Specify 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*—Specify 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*—Enable automated testing of the RADIUS server status, and specify the username to be used. • **idle-time** *time*—Set 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**—Disable testing on the RADIUS-server accounting port. • **ignore-auth-port**—Disable testing on the RADIUS-server authentication port. • For **key***string*, specify 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.

	Command or Action	**Purpose**
		You can also configure the authentication and encryption key by using the**radius-server key** {**0**string | **7***string* | *string*} global configuration command.
Step 6	**dot1x critical** {**eapol** | **recovery delay** *milliseconds*} **Example:** Switch(config)# **dot1x critical eapol** (config)# **dot1x critical recovery delay 2000**	(Optional) Configure the parameters for inaccessible authentication bypass: • **eapol**—Specify that the switch sends an EAPOL-Success message when the switch successfully authenticates the critical port. • **recovery delay***milliseconds*—Set the recovery delay period during which the switch waits to re-initialize 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 re-initialized every second).
Step 7	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet 1/0/1**	Specify the port to be configured, and enter interface configuration mode.
Step 8	**authentication event server dead action** {**authorize** | **reinitialize**} **vlan** *vlan-id*] **Example:** Switch(config-if)# **authentication event server dead action reinitialicze vlan 20**	Use these keywords to move hosts on the port if the RADIUS server is unreachable: • **authorize**—Move any new hosts trying to authenticate to the user-specified critical VLAN. • **reinitialize**—Move all authorized hosts on the port to the user-specified critical VLAN.
Step 9	**switchport voice vlan** *vlan-id* **Example:** Switch(config-if)# **switchport voice vlan**	Specifies the voice VLAN for the port. The voice VLAN cannot be the same as the critical data VLAN configured in Step 6.
Step 10	**authentication event server dead action authorize voice** **Example:** Switch(config-if)# **authentication event server dead action authorize voice**	Configures critical voice VLAN to move data traffic on the port to the voice VLAN if the RADIUS server is unreachable.
Step 11	**show authentication interface** *interface-id* **Example:**	(Optional) Verify your entries.

	Command or Action	Purpose
	`Switch(config-if)#  do show authentication interface gigabit 1/0/1`	
Step 12	**copy running-config startup-config** **Example:** `Switch(config-if)#  do copy running-config startup-config`	(Optional) Verify your entries.

Example

To return to the RADIUS server default settings, use the **no radius-server dead-criteria**, the **no radius-server deadtime**, and the **no radius-server host** global configuration commands. To disable inaccessible authentication bypass, use the **no authentication event server dead action** interface configuration command. To disable critical voice VLAN, use the no **authentication event server dead action authorize voice** interface configuration command.

Example of Configuring Inaccessible Authentication Bypass

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 10.0.0.10 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 1/0/1
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 802.1x Authentication with WoL

Beginning in privileged EXEC mode, follow these steps to enable 802.1x authentication with WoL. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication control-direction** {**both** | **in**}
4. **end**
5. **show authentication sessions interface** *interface-id*
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* Example: Switch(config)# **interface gigabitethernet2/0/3**	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**authentication control-direction** {**both** | **in**} Example: Switch(config-if)# **authentication control-direction both**	Enables 802.1x authentication with WoL on the port, and use 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.
Step 4	**end** Example: Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 5	**show authentication sessions interface** *interface-id* Example: Switch# **show authentication sessions interface gigabitethernet2/0/3**	Verifies your entries.
Step 6	**copy running-config startup-config** Example: Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring MAC Authentication Bypass

Beginning in privileged EXEC mode, follow these steps to enable MAC authentication bypass. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication port-control auto**
4. **mab** [**eap**]
5. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet2/0/1**	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**authentication port-control auto** **Example:** Switch(config-if)# **authentication port-control auto**	Enables 802.1x authentication on the port.
Step 4	**mab** [**eap**] **Example:** Switch(config-if)# **mab**	Enables MAC authentication bypass. (Optional) Use the **eap** keyword to configure the switch to use EAP for authorization.
Step 5	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.

Formatting a MAC Authentication Bypass Username and Password

Use the optional **mab request format** command to format the MAB username and password in a style accepted by the authentication server. The username and password are usually the MAC address of the client. Some authentication server configurations require the password to be different from the username.

Beginning in privileged EXEC mode, follow these steps to format MAC authentication bypass username and passwords.

SUMMARY STEPS

1. **configure terminal**
2. **mab request format attribute 1 groupsize** {**1** | **2** | **4** |**12**} [**separator** {**-** | **:** | **.**} {**lowercase** | **uppercase**}]
3. **mab request format attribute2** {**0** | **7**} *text*
4. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**mab request format attribute 1 groupsize** {**1**	**2**	**4** |**12**} [**separator** {**-**	**:**	**.**} {**lowercase**	**uppercase**}] **Example:** Switch(config)# **mab request format attribute 1 groupsize 12**	Specifies the format of the MAC address in the User-Name attribute of MAB-generated Access-Request packets. 1—Sets the username format of the 12 hex digits of the MAC address. group size—The number of hex nibbles to concatenate before insertion of a separator. A valid groupsize must be either 1, 2, 4, or 12. separator—The character that separates the hex nibbles according to group size. A valid separator must be either a hyphen, colon, or period. No separator is used for a group size of 12. {lowercase	uppercase}—Specifies if nonnumeric hex nibbles should be in lowercase or uppercase.
Step 3	**mab request format attribute2** {**0**	**7**} *text* **Example:** Switch(config)# **mab request format attribute 2 7 A02f44E18B12**	2—Specifies a custom (nondefault) value for the User-Password attribute in MAB-generated Access-Request packets. 0—Specifies a cleartext password to follow. 7—Specifies an encrypted password to follow. *text*—Specifies the password to be used in the User-Password attribute. **Note** When you send configuration information in e-mail, remove type 7 password information. The **show tech-support** command removes this information from its output by default.
Step 4	**end** **Example:**	Returns to privileged EXEC mode.

Command or Action	Purpose
Switch(config)# end	

Configuring 802.1x User Distribution

Beginning in privileged EXEC mode, follow these steps to configure a VLAN group and to map a VLAN to it:

SUMMARY STEPS

1. **configure terminal**
2. **vlan group** *vlan-group-name* **vlan-list** *vlan-list*
3. **end**
4. **no vlan group** *vlan-group-name* **vlan-list** *vlan-list*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**vlan group** *vlan-group-name* **vlan-list** *vlan-list* **Example:** Switch(config)# **vlan group eng-dept vlan-list 10**	Configures a VLAN group, and maps a single VLAN or a range of VLANs to it.
Step 3	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 4	**no vlan group** *vlan-group-name* **vlan-list** *vlan-list* **Example:** Switch(config)# **no vlan group eng-dept vlan-list 10**	Clears the VLAN group configuration or elements of the VLAN group configuration.

Example of Configuring VLAN Groups

This example shows how to configure the VLAN groups, to map the VLANs to the groups, to and 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(config)# 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

You can configure NAC Layer 2 802.1x validation, which is also referred to as 802.1x authentication with a RADIUS server.

Beginning in privileged EXEC mode, follow these steps to configure NAC Layer 2 802.1x validation. The procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode access**

4. **authentication event no-response action authorize vlan** *vlan-id*
5. **authentication periodic**
6. **authentication timer reauthenticate**
7. **end**
8. **show authentication sessions interface** *interface-id*
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet2/0/3**	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**switchport mode access** **Example:** Switch(config-if)# **switchport mode access**	Sets the port to access mode only if you configured the RADIUS server.
Step 4	**authentication event no-response action authorize vlan** *vlan-id* **Example:** Switch(config-if)# **authentication event no-response action authorize vlan 8**	Specifies an active VLAN as an 802.1x guest VLAN. The range is 1 to 4094. 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.
Step 5	**authentication periodic** **Example:** Switch(config-if)# **authentication periodic**	Enables periodic re-authentication of the client, which is disabled by default.
Step 6	**authentication timer reauthenticate** **Example:** Switch(config-if)# **authentication timer reauthenticate**	Sets re-authentication attempt for the client (set to one hour). This command affects the behavior of the switch only if periodic re-authentication is enabled.

	Command or Action	Purpose
Step 7	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 8	**show authentication sessions interface** *interface-id* **Example:** Switch# **show authentication sessions interface gigabitethernet2/0/3**	Verifies your entries.
Step 9	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Limiting Login for Users

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa authentication login default local**
5. **aaa authentication rejected** *n* **in** *m* **ban** *x*
6. **end**
7. **show aaa local user blocked**
8. **clear aaa local user blocked username** *username*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Device# configure terminal	Enters global configuration mode.
Step 3	**aaa new-model** **Example:**	Enables the authentication, authorization, and accounting (AAA) access control model.

	Command or Action	Purpose
	Device(config)# aaa new-model	
Step 4	**aaa authentication login default local** **Example:** Device(config)# aaa authentication login default local	Sets the authentication, authorization, and accounting (AAA) authentication by using the default authentication methods.
Step 5	**aaa authentication rejected** *n* **in** *m* **ban** *x* **Example:** Device(config)# aaa authentication rejected 3 in 20 ban 300	Configures the time period for which an user is blocked, if the user fails to successfully login within the specified time and login attempts. • *n*—Specifies the number of times a user can try to login. • *m*—Specifies the number of seconds within which an user can try to login. • *x*—Specifies the time period an user is banned if the user fails to successfully login.
Step 6	**end** **Example:** Device(config)# end	Exits global configuration mode and returns to privileged EXEC mode.
Step 7	**show aaa local user blocked** **Example:** Device# show aaa local user blocked	Displays the list of local users who were blocked.
Step 8	**clear aaa local user blocked username** *username* **Example:** Device# clear aaa local user blocked username user1	Clears the information about the blocked local user.

Example

The following is sample output from the **show aaa local user blocked** command:

```
Device# show aaa local user blocked

      Local-user               State

      user1                    Watched (till 11:34:42 IST Feb 5 2015)
```

Configuring an Authenticator Switch with NEAT

Configuring this feature requires that one switch outside a wiring closet is configured as a supplicant and is connected to an authenticator switch.

> ✎
>
> | **Note** | • The authenticator switch interface configuration must be restored to access mode by explicitly flapping it if a line card is removed and inserted in the chassis when CISP or NEAT session is active. |
> | | • The *cisco-av-pairs* must be configured as *device-traffic-class=switch* on the ISE, which sets the interface as a trunk after the supplicant is successfully authenticated. |

Beginning in privileged EXEC mode, follow these steps to configure a switch as an authenticator:

SUMMARY STEPS

1. **configure terminal**
2. **cisp enable**
3. **interface** *interface-id*
4. **switchport mode access**
5. **authentication port-control auto**
6. **dot1x pae authenticator**
7. **spanning-tree portfast**
8. **end**
9. **show running-config interface** *interface-id*
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**cisp enable** **Example:** Switch(config)# **cisp enable**	Enables CISP.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet 2/0/1**	Specifies the port to be configured, and enter interface configuration mode.
Step 4	**switchport mode access** **Example:** Switch(config-if)# **switchport mode access**	Sets the port mode to **access**.

		Command or Action	**Purpose**
Step 5		**authentication port-control auto**	Sets the port-authentication mode to auto.
		Example:	
		Switch(config-if)# **authentication port-control auto**	
Step 6		**dot1x pae authenticator**	Configures the interface as a port access entity (PAE) authenticator.
		Example:	
		Switch(config-if)# **dot1x pae authenticator**	
Step 7		**spanning-tree portfast**	Enables Port Fast on an access port connected to a single workstation or server..
		Example:	
		Switch(config-if)# **spanning-tree portfast trunk**	
Step 8		**end**	Returns to privileged EXEC mode.
		Example:	
		Switch(config-if)# **end**	
Step 9		**show running-config interface** *interface-id*	Verifies your configuration.
		Example:	
		Switch# **show running-config interface gigabitethernet 2/0/1**	
Step 10		**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.
		Example:	**Note** Saving changes to the configuration file will mean that the authenticator interface will continue to be in trunk mode after reload. If you want the authenticator interface to remain as an access port, do not save your changes to the configuration file.
		Switch# **copy running-config startup-config**	

Configuring a Supplicant Switch with NEAT

Beginning in privileged EXEC mode, follow these steps to configure a switch as a supplicant:

SUMMARY STEPS

1. **configure terminal**
2. **cisp enable**

3. **dot1x credential**s *profile*

4. **username** *suppswitch*

5. **password** *password*

6. **dot1x supplicant force-multicast**

7. **interface** *interface-id*

8. **switchport trunk encapsulation dot1q**

9. **switchport mode trunk**

10. **dot1x pae supplicant**

11. **dot1x credentials** *profile-name*

12. **end**

13. **show running-config interface** *interface-id*

14. **copy running-config startup-config**

15. Configuring NEAT with Auto Smartports Macros

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**cisp enable** **Example:** Switch(config)# **cisp enable**	Enables CISP.
Step 3	**dot1x credential**s *profile* **Example:** Switch(config)# **dot1x credentials test**	Creates 802.1x credentials profile. This must be attached to the port that is configured as supplicant.
Step 4	**username** *suppswitch* **Example:** Switch(config)# **username suppswitch**	Creates a username.
Step 5	**password** *password* **Example:** Switch(config)# **password myswitch**	Creates a password for the new username.

	Command or Action	Purpose
Step 6	**dot1x supplicant force-multicast** **Example:** Switch(config)# **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.
Step 7	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	Specifies the port to be configured, and enter interface configuration mode.
Step 8	**switchport trunk encapsulation dot1q** **Example:** Switch(config-if)# **switchport trunk encapsulation dot1q**	Sets the port to trunk mode.
Step 9	**switchport mode trunk** **Example:** Switch(config-if)# **switchport mode trunk**	Configures the interface as a VLAN trunk port.
Step 10	**dot1x pae supplicant** **Example:** Switch(config-if)# **dot1x pae supplicant**	Configures the interface as a port access entity (PAE) supplicant.
Step 11	**dot1x credentials** *profile-name* **Example:** Switch(config-if)# **dot1x credentials test**	Attaches the 802.1x credentials profile to the interface.
Step 12	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 13	**show running-config interface** *interface-id* **Example:** Switch# **show running-config interface gigabitethernet1/0/1**	Verifies your configuration.

	Command or Action	**Purpose**
Step 14	**copy running-config startup-config** **Example:** `Switch# ` **`copy running-config startup-config`**	(Optional) Saves your entries in the configuration file.
Step 15	Configuring NEAT with Auto Smartports Macros	You can also use an Auto Smartports user-defined macro instead of the switch VSA to configure the authenticator switch. For more information, see the *Auto Smartports Configuration Guide* for this release.

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 *Configuration Guide for Cisco Secure ACS 4.2*:
http://www.cisco.com/en/US/docs/net_mgmt/cisco_secure_access_control_server_for_windows/4.2/configuration/guide/acs_config.pdf

> **Note**   You must configure a downloadable ACL on the ACS before downloading it to the switch.

After authentication on the port, you can use the **show ip access-list** privileged EXEC command to display the downloaded ACLs on the port.

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.

Beginning in privileged EXEC mode:

SUMMARY STEPS

1. **configure terminal**
2. **ip device tracking**
3. **aaa new-model**
4. **aaa authorization network default local group radius**
5. **radius-server vsa send authentication**
6. **interface** *interface-id*
7. **ip access-group** *acl-id* **in**
8. **show running-config interface** *interface-id*
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**ip device tracking** **Example:** Switch(config)# **ip device tracking**	Sets the ip device tracking table.
Step 3	**aaa new-model** **Example:** Switch(config)# **aaa new-model**	Enables AAA.
Step 4	**aaa authorization network default local group radius** **Example:** Switch(config)# **aaa authorization network default local group radius**	Sets the authorization method to local. To remove the authorization method, use the **no aaa authorization network default local group radius** command.
Step 5	**radius-server vsa send authentication** **Example:** Switch(config)# **radius-server vsa send authentication**	Configures the radius vsa send authentication.
Step 6	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet2/0/4**	Specifies the port to be configured, and enter interface configuration mode.
Step 7	**ip access-group** *acl-id* **in** **Example:** Switch(config-if)# **ip access-group default_acl in**	Configures the default ACL on the port in the input direction. **Note** The *acl-id* is an access list name or number.
Step 8	**show running-config interface** *interface-id* **Example:**	Verifies your configuration.

Command or Action	**Purpose**
Switch(config-if)# **show running-config interface gigabitethernet2/0/4**	
Step 9 **copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring a Downloadable Policy

Beginning in privileged EXEC mode:

SUMMARY STEPS

1. **configure terminal**
2. **access-list** *access-list-number* **{ deny | permit } { hostname | any | host } log**
3. **interface** *interface-id*
4. **ip access-group** *acl-id* **in**
5. **exit**
6. **aaa new-model**
7. **aaa authorization network default group radius**
8. **ip device tracking**
9. **ip device tracking probe** [**count** | **interval** | **use-svi**]
10. **radius-server vsa send authentication**
11. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**access-list** *access-list-number* **{ deny	permit } { hostname	any	host } log** **Example:** Switch(config)# **access-list 1 deny any log**	Defines the default port ACL. 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 that sends a packet, such as this:

	Command or Action	**Purpose**
		• **hostname**: The 32-bit quantity in dotted-decimal format.
		• **any**: 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.
		• **host**: The keyword host as an abbreviation for source and source-wildcard of source 0.0.0.0.
		(Optional) Applies the source-wildcard wildcard bits to the source.
		(Optional) Enters log to cause an informational logging message about the packet that matches the entry to be sent to the console.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet2/0/2**	Enters interface configuration mode.
Step 4	**ip access-group** *acl-id* **in** **Example:** Switch(config-if)# **ip access-group default_acl in**	Configures the default ACL on the port in the input direction. **Note** The acl-id is an access list name or number.
Step 5	**exit** **Example:** Switch(config-if)# **exit**	Returns to global configuration mode.
Step 6	**aaa new-model** **Example:** Switch(config)# **aaa new-model**	Enables AAA.
Step 7	**aaa authorization network default group radius** **Example:** Switch(config)# **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.
Step 8	**ip device tracking** **Example:**	Enables the IP device tracking table. To disable the IP device tracking table, use the **no ip device tracking** global configuration commands.

	Command or Action	Purpose
	Switch(config)# **ip device tracking**	
Step 9	**ip device tracking probe** [**count** | **interval** | **use-svi**] **Example:** Switch(config)# **ip device tracking probe count**	(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.
Step 10	**radius-server vsa send authentication** **Example:** Switch(config)# **radius-server vsa send authentication**	Configures the network access server to recognize and use vendor-specific attributes. **Note** The downloadable ACL must be operational.
Step 11	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Configuring VLAN ID-based MAC Authentication

Beginning in privileged EXEC mode, follow these steps:

SUMMARY STEPS

1. **configure terminal**
2. **mab request format attribute 32 vlan access-vlan**
3. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**mab request format attribute 32 vlan access-vlan** **Example:** Switch(config)# **mab request format attribute 32 vlan access-vlan**	Enables VLAN ID-based MAC authentication.
Step 3	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Flexible Authentication Ordering

The examples used in the instructions below changes the order of Flexible Authentication Ordering so that MAB is attempted before IEEE 802.1X authentication (dot1x). MAB is configured as the first authentication method, so MAB will have priority over all other authentication methods.

Beginning in privileged EXEC mode, follow these steps:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode access**
4. **authentication order** [**dot1x** | **mab**] | {**webauth**}
5. **authentication priority** [**dot1x** | **mab**] | {**webauth**}
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet 1/0/1**	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**switchport mode access** **Example:**	Sets the port to access mode only if you previously configured the RADIUS server.

	Command or Action	Purpose
	Switch(config-if)# **switchport mode access**	
Step 4	**authentication order** [**dot1x**	**mab**]	{**webauth**} **Example:** Switch(config-if)# **authentication order mab dot1x**	(Optional) Sets the order of authentication methods used on a port.
Step 5	**authentication priority** [**dot1x**	**mab**]	{**webauth**} **Example:** Switch(config-if)# **authentication priority mab dot1x**	(Optional) Adds an authentication method to the port-priority list.
Step 6	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.

Configuring Open1x

Beginning in privileged EXEC mode, follow these steps to enable manual control of the port authorization state:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode access**
4. **authentication control-direction** {**both** | **in**}
5. **authentication fallback** *name*
6. **authentication host-mode** [**multi-auth** | **multi-domain** | **multi-host** | **single-host**]
7. **authentication open**
8. **authentication order** [**dot1x** | **mab**] | {**webauth**}
9. **authentication periodic**
10. **authentication port-control** {**auto** | **force-authorized** | **force-un authorized**}
11. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:**	Enters global configuration mode.

	Command or Action	Purpose
	Switch# **configure terminal**	
Step 2	**interface** *interface-id* Example: Switch(config)# **interface gigabitethernet 1/0/1**	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**switchport mode access** Example: Switch(config-if)# **switchport mode access**	Sets the port to access mode only if you configured the RADIUS server.
Step 4	**authentication control-direction** {**both** | **in**} Example: Switch(config-if)# **authentication control-direction both**	(Optional) Configures the port control as unidirectional or bidirectional.
Step 5	**authentication fallback** *name* Example: Switch(config-if)# **authentication fallback profile1**	(Optional) Configures a port to use web authentication as a fallback method for clients that do not support 802.1x authentication.
Step 6	**authentication host-mode** [**multi-auth** | **multi-domain** | **multi-host** | **single-host**] Example: Switch(config-if)# **authentication host-mode multi-auth**	(Optional) Sets the authorization manager mode on a port.
Step 7	**authentication open** Example: Switch(config-if)# **authentication open**	(Optional) Enables or disable open access on a port.
Step 8	**authentication order** [**dot1x** | **mab**] | {**webauth**} Example: Switch(config-if)# **authentication order dot1x webauth**	(Optional) Sets the order of authentication methods used on a port.

	Command or Action	Purpose
Step 9	**authentication periodic**	(Optional) Enables or disable reauthentication on a port.
	Example:	
	Switch(config-if)# **authentication periodic**	
Step 10	**authentication port-control** {**auto** | **force-authorized** | **force-un authorized**}	(Optional) Enables manual control of the port authorization state.
	Example:	
	Switch(config-if)# **authentication port-control auto**	
Step 11	**end**	Returns to privileged EXEC mode.
	Example:	
	Switch(config-if)# **end**	

Disabling 802.1x Authentication on the Port

You can disable 802.1x authentication on the port by using the **no dot1x pae** interface configuration command.

Beginning in privileged EXEC mode, follow these steps to disable 802.1x authentication on the port. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode access**
4. **no dot1x pae authenticator**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
	Example:	
	Switch# **configure terminal**	
Step 2	**interface** *interface-id*	Specifies the port to be configured, and enter interface configuration mode.
	Example:	

	Command or Action	Purpose
	`Switch(config)# interface gigabitethernet2/0/1`	
Step 3	**switchport mode access** **Example:** `Switch(config-if)# switchport mode access`	(Optional) Sets the port to access mode only if you configured the RADIUS server.
Step 4	**no dot1x pae authenticator** **Example:** `Switch(config-if)# no dot1x pae authenticator`	Disables 802.1x authentication on the port.
Step 5	**end** **Example:** `Switch(config-if)# end`	Returns to privileged EXEC mode.

Resetting the 802.1x Authentication Configuration to the Default Values

Beginning in privileged EXEC mode, follow these steps to reset the 802.1x authentication configuration to the default values. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **dot1x default**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/2`	Enters interface configuration mode, and specify the port to be configured.

	Command or Action	**Purpose**
Step 3	**dot1x default** **Example:** Switch(config-if)# **dot1x default**	Resets the 802.1x parameters to the default values.
Step 4	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.

Monitoring 802.1x Statistics and Status

Table 97: Privileged EXEC show Commands

Command	**Purpose**
show dot1x all statistics	Displays 802.1x statistics for all ports
show dot1x interface *interface-id* **statistics**	Displays 802.1x statistics for a specific port
show dot1x all [**count** | **details** | **statistics** | **summary**]	Displays the 802.1x administrative and operational status for a switch
show dot1x interface *interface-id*	Displays the 802.1x administrative and operational status for a specific port

Table 98: Global Configuration Commands

Command	**Purpose**
no dot1x logging verbose	Filters verbose 802.1x authentication messages (beginning with Cisco IOS Release 12.2(55)SE)

For detailed information about the fields in these displays, see the command reference for this release.

Additional References

Related Documents

Related Topic	Document Title
Configuring Identity Control policies and Identity Service templates for Session Aware networking.	Session Aware Networking Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches) http://www.cisco.com/en/US/docs/ios-xml/ios/san/configuration/xe-3se/3850/san-xe-3se-3850-book.htm
Configuring RADIUS, TACACS+, Secure Shell, 802.1X and AAA.	Securing User Services Configuration Guide Library, Cisco IOS XE Release 3SE (Catalyst 3850 Switch http://www.cisco.com/en/US/docs/ios-xml/ios/security/config_library/xe-3se/3850/secuser-xe-3se-3850-l

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information for 802.1x Port-Based Authentication

Release	Feature Information
Cisco IOS Release 15.0(2)EX1	This feature was introduced.
	Supports the use of same authorization methods on all the Catalyst switches in a network.
	Supports filtering verbose system messages from the authentication manager.

Configuring Web-Based Authentication

This chapter describes how to configure web-based authentication on the switch. It contains these sections:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About 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 and Layer 3 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.

Note HTTPS traffic interception for central web authentication redirect is not supported.

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 92: Web-Based Authentication Device Roles

This figure shows the roles of these devices in a

network.
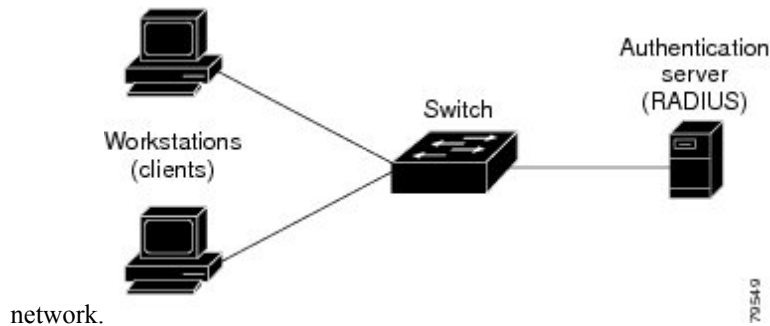
Host Detection

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.

- 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

With Web Authentication, you can create a default and customized web-browser banners that appears when you log in to a switch.

The banner appears on both the login page and the authentication-result pop-up pages. The default banner messages are as follows:

- *Authentication Successful*

- *Authentication Failed*

- *Authentication Expired*

The Local Web Authentication Banner can be configured in legacy and new-style (Session-aware) CLIs as follows:

- Legacy mode—Use the **ip admission auth-proxy-banner http** global configuration command.

- New-style mode—Use the **parameter-map type webauth global banner** 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.

Figure 93: Authentication Successful Banner
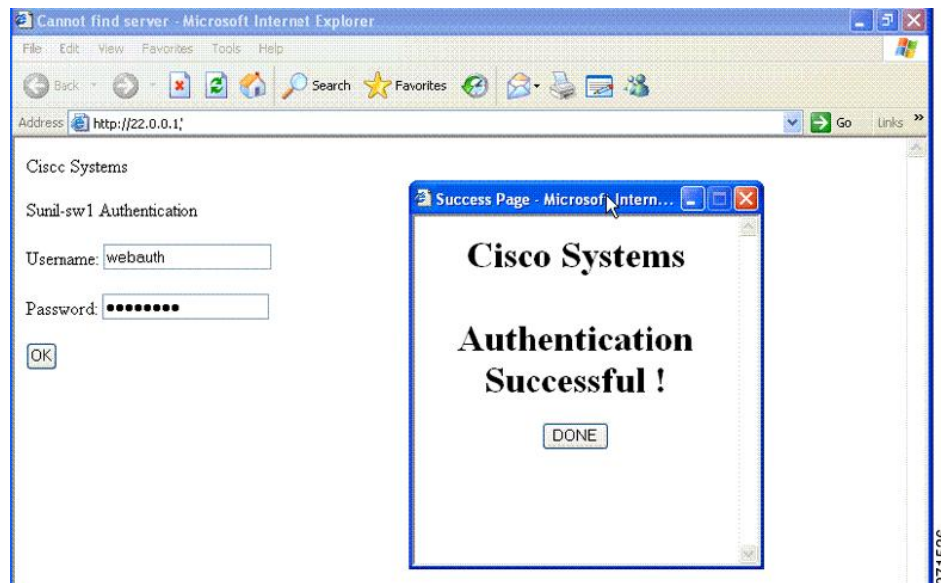
The banner can be customized as follows:

- Add a message, such as switch, router, or company name to the banner:

  - Legacy mode—Use the **ip admission auth-proxy-banner http** *banner-text* global configuration command.

  - New-style mode—Use the **parameter-map type webauth global banner** global configuration command

- Add a logo or text file to the banner :

- Legacy mode—Use the **ip admission auth-proxy-banner http** *file-path* global configuration command.

- New-style mode—Use the **parameter-map type webauth global banner** global configuration command

Figure 94: Customized Web Banner
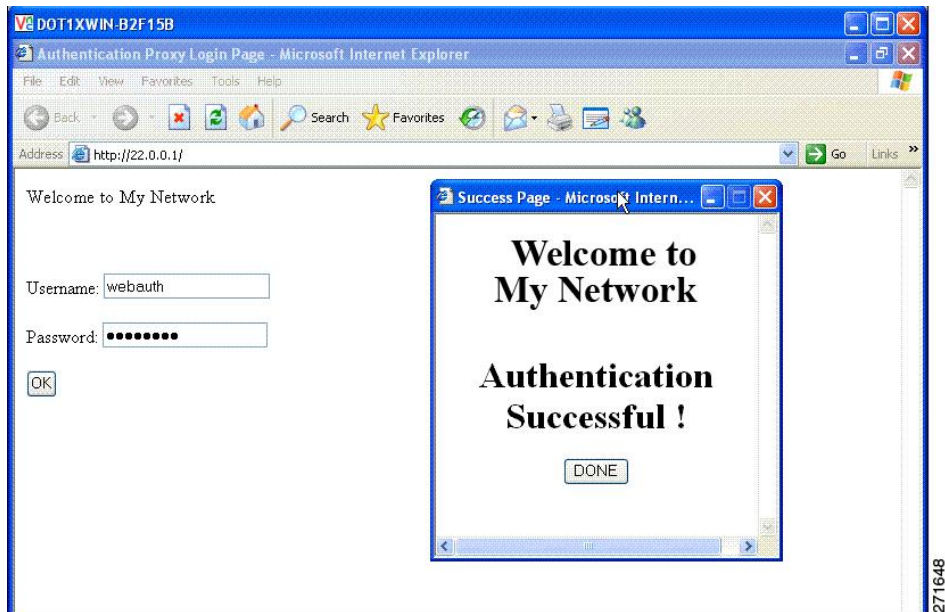
If you do not enable a banner, only the username and password dialog boxes appear in the web authentication login screen, and no banner appears when you log into the switch.

Figure 95: Login Screen With No Banner
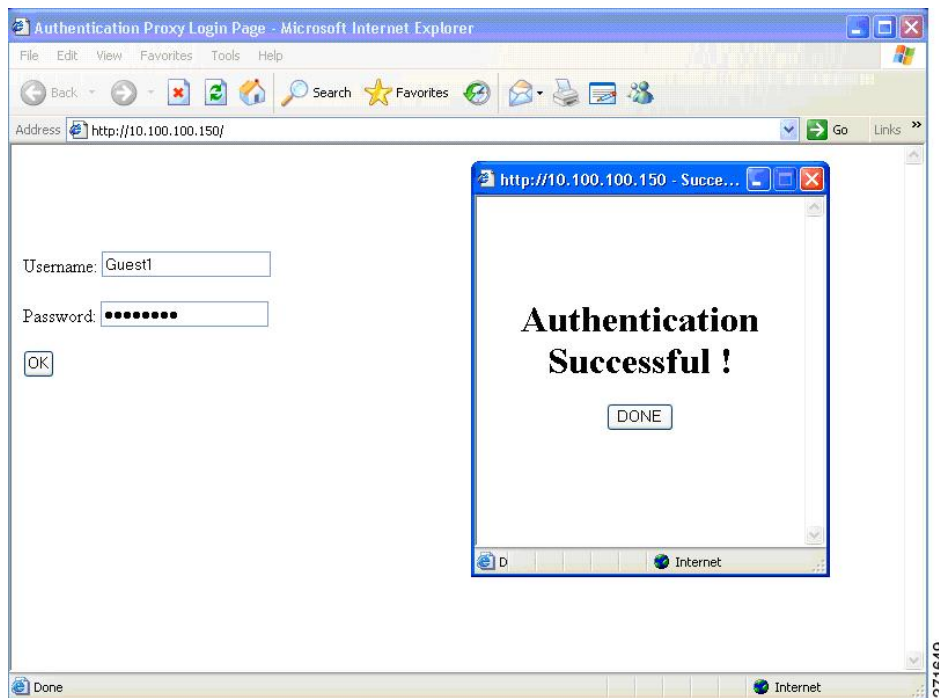
For more information, see the *Session Aware Networking Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches) Session Aware Networking Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)* and the *Web Authentication Enhancements - Customizing Authentication Proxy Web Pages*.

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.

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* 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 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.

- On stackable switches, 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 file name.

- The configured authentication proxy feature supports both HTTP and SSL.

You can substitute your HTML pages 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 96: Customizable Authentication Page
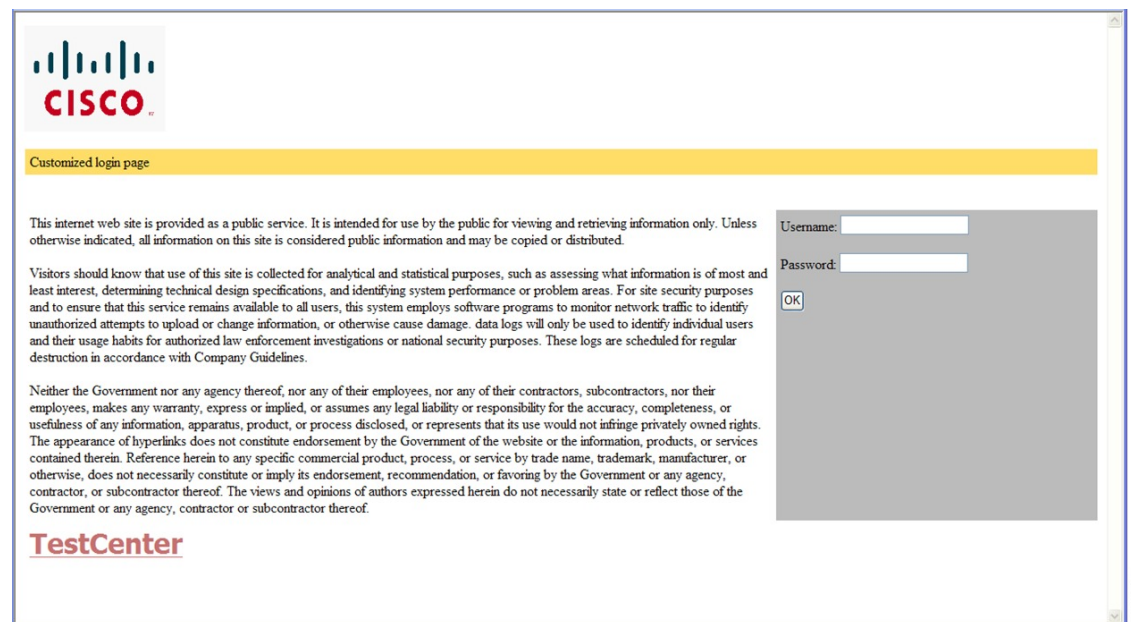
Authentication Proxy Web Page Guidelines

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.

Related Topics

Customizing the Authentication Proxy Web Pages, on page 1012

Redirection URL for Successful Login Guidelines

When configuring a redirection URL for successful login, consider these guidelines:

- If the custom authentication proxy web pages feature is enabled, the redirection URL feature is disabled and is not available in the CLI. You can perform redirection in the custom-login success page.

- If the redirection URL feature is enabled, a configured auth-proxy-banner is not used.

- To remove the specification of a redirection URL, use the **no** form of the command.

- If the redirection URL is required after the web-based authentication client is successfully authenticated, then the URL string must start with a valid URL (for example, http://) followed by the URL information. If only the URL is given without http://, then the redirection URL on successful authentication might cause page not found or similar errors on a web browser.

Related Topics

Specifying a Redirection URL for Successful Login, on page 1013

Web-based Authentication Interactions with Other Features

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.

Related Topics

Enabling and Configuring Port Security, on page 1041

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, it is more secure, though not required, to 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. The Policy ACL is applied to the session even if there is no ACL configured on the port.

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.

EtherChannel

You can configure web-based authentication on a Layer 2 EtherChannel interface. The web-based authentication configuration applies to all member channels.

How to Configure Web-Based Authentication

Default Web-Based Authentication Configuration

The following table shows the default web-based authentication configuration.

Table 99: Default Web-based Authentication Configuration

Feature	Default Setting
AAA	Disabled

RADIUS server
 	
  • IP address
 	
  • UDP authentication port
 	
  • Key
	• None specified
	
	• 1645
	
	• None specified

Default value of inactivity timeout	3600 seconds
Inactivity timeout	Enabled

Web-Based Authentication Configuration Guidelines and Restrictions

- 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.

- 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.

- 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 supports IPv6 in Session-aware policy mode. IPv6 Web-authentication requires at least one IPv6 address configured on the switch and IPv6 Snooping configured on the switchport.

- 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 NRH (Non-Responsive Host) is not supported for voice devices.

- Only the Password Authentication Protocol (PAP) is supported for web-based RADIUS authentication on controllers. The Challenge Handshake Authentication Protocol (CHAP) is not supported for web-based RADIUS authentication on controllers.

- Identify the following RADIUS security server settings that will be used while configuring switch-to-RADIUS-server communication:

    - 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.

- When you configure the RADIUS server parameters:

    - Specify the **key** *string* on a separate command line.

    - For **key** *string*, specify 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.

    - When you specify the **key** *string*, use spaces within and at the end of the key. 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 globally configure the timeout, retransmission, and encryption key values for all RADIUS servers by using with 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 transmit, and the radius-server key global configuration commands. For more information, see the *Cisco IOS Security Configuration Guide*, Release 12.4 and the *Cisco IOS Security Command Reference*, Release 12.4.

    Note   You need to configure some settings on the RADIUS server, including: the switch IP address, the key string to be shared by both the server and the switch, and the downloadable ACL (DACL). For more information, see the RADIUS server documentation.

Web-Based Authentication Configuration Task List

Configuring the Authentication Rule and Interfaces

Examples in this section are legacy-style configurations. For new-style configurations, see the *Session Aware Networking Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*

Follow these steps to configure the authentication rule and interfaces:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip admission name** *name* **proxy http**
4. **interface** *type slot/port*
5. **ip access-group** *name*
6. **ip admission** *name*
7. **exit**
8. **ip device tracking**
9. **end**
10. **show ip admission status**
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**ip admission name** *name* **proxy http** **Example:** Switch(config)# **ip admission name webauth1 proxy http**	Configures an authentication rule for web-based authorization.

	Command or Action	**Purpose**
Step 4	**interface** *type slot/port*	Enters interface configuration mode and specifies the ingress Layer 2 or Layer 3 interface to be enabled for web-based authentication.
	Example:	
	Switch(config)# **interface gigabitEthernet1/0/1**	*type* can be fastethernet, gigabit ethernet, or tengigabitethernet.
Step 5	**ip access-group** *name*	Applies the default ACL.
	Example:	
	Switch(config-if)# **ip access-group webauthag**	
Step 6	**ip admission** *name*	Configures web-based authentication on the specified interface.
	Example:	
	Switch(config-if)# **ip admission webauth1**	
Step 7	**exit**	Returns to configuration mode.
	Example:	
	Switch(config-if)# **exit**	
Step 8	**ip device tracking**	Enables the IP device tracking table.
	Example:	
	Switch(config)# **ip device tracking**	
Step 9	**end**	Returns to privileged EXEC mode.
	Example:	
	Switch(config)# **end**	
Step 10	**show ip admission status**	Displays the configuration.
	Example:	
	Switch# **show ip admission *status***	
Step 11	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.
	Example:	
	Switch# **copy running-config startup-config**	

Configuring AAA Authentication

Follow these steps to configure AAA authentication:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa authentication login default group** {**tacacs+** | **radius**}
5. **aaa authorization auth-proxy default group** {**tacacs+** | **radius**}
6. **tacacs-server host** {*hostname* | *ip_address*}
7. **tacacs-server key** {*key-data*}
8. **end**
9. **show running-config**
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**aaa new-model** **Example:** Switch(config)# **aaa new-model**	Enables AAA functionality.
Step 4	**aaa authentication login default group** {**tacacs+** | **radius**} **Example:** Switch(config)# **aaa authentication login default group tacacs+**	Defines the list of authentication methods at login.
Step 5	**aaa authorization auth-proxy default group** {**tacacs+** | **radius**} **Example:**	Creates an authorization method list for web-based authorization.

	Command or Action	Purpose
	Switch(config)# **aaa authorization auth-proxy default group tacacs+**	
Step 6	**tacacs-server host** {*hostname* | *ip_address*} **Example:** Switch(config)# **tacacs-server host 10.1.1.1**	Specifies an AAA server.
Step 7	**tacacs-server key** {*key-data*} **Example:** Switch(config)# **tacacs-server key**	Configures the authorization and encryption key used between the switch and the TACACS server.
Step 8	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 9	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 10	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Switch-to-RADIUS-Server Communication

Follow these steps to configure the RADIUS server parameters:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip radius source-interface vlan** *vlan interface number*
4. **radius-server host** {*hostname* \| *ip-address*} **test username** *username*
5. **radius-server key** *string*
6. **radius-server dead-criteria tries** *num-tries*
7. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**ip radius source-interface vlan** *vlan interface number* **Example:** Switch(config)# **ip radius source-interface vlan 80**	Specifies that the RADIUS packets have the IP address of the indicated interface.
Step 4	**radius-server host** {*hostname* | *ip-address*} **test username** *username* **Example:** Switch(config)# **radius-server host 172.120.39.46 test username user1**	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.
Step 5	**radius-server key** *string* **Example:** Switch(config)# **radius-server key rad123**	Configures the authorization and encryption key used between the switch and the RADIUS daemon running on the RADIUS server.
Step 6	**radius-server dead-criteria tries** *num-tries* **Example:** Switch(config)# **radius-server dead-criteria tries 30**	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.
Step 7	**end** **Example:**	Returns to privileged EXEC mode.

Command or Action	Purpose
Switch(config)# **end**	

Configuring the HTTP Server

To use web-based authentication, you must enable the HTTP server within the Switch. You can enable the server for either HTTP or HTTPS.

Note	The Apple psuedo-browser will not open if you configure only the **ip http secure-server** command. You should also configure the **ip http server** command.

Follow these steps to enable the server for either HTTP or HTTPS:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip http server**
4. **ip http secure-server**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**ip http server** **Example:** Switch(config)# **ip http server**	Enables the HTTP server. The web-based authentication feature uses the HTTP server to communicate with the hosts for user authentication.
Step 4	**ip http secure-server** **Example:**	Enables HTTPS. You can configure custom authentication proxy web pages or specify a redirection URL for successful login.

	Command or Action	Purpose
	Switch(config)# **ip http secure-server**	**Note** To ensure secure authentication when you enter the **ip http secure-server** command, the login page is always in HTTPS (secure HTTP) even if the user sends an HTTP request.
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Customizing the Authentication Proxy Web Pages

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.

For the equivalent Session Aware Networking configuration example for this feature, see the section "Configuring a Parameter Map for Web-Based Authentication" in the chapter, "Configuring Identity Control Policies." of the book, "*Session Aware Networking Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches).*"

Follow these steps to specify the use of your custom authentication proxy web pages:

Before you begin

Store your custom HTML files on the Switch flash memory.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip admission proxy http login page file** *device:login-filename*
4. **ip admission proxy http success page file** *device:success-filename*
5. **ip admission proxy http failure page file** *device:fail-filename*
6. **ip admission proxy http login expired page file** *device:expired-filename*
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:**	Enters global configuration mode.

	Command or Action	Purpose
	Switch# **configure terminal**	
Step 3	**ip admission proxy http login page file** *device:login-filename* **Example:** Switch(config)# **ip admission proxy http login page file disk1:login.htm**	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.
Step 4	**ip admission proxy http success page file** *device:success-filename* **Example:** Switch(config)# **ip admission proxy http success page file disk1:success.htm**	Specifies the location of the custom HTML file to use in place of the default login success page.
Step 5	**ip admission proxy http failure page file** *device:fail-filename* **Example:** Switch(config)# **ip admission proxy http fail page file disk1:fail.htm**	Specifies the location of the custom HTML file to use in place of the default login failure page.
Step 6	**ip admission proxy http login expired page file** *device:expired-filename* **Example:** Switch(config)# **ip admission proxy http login expired page file disk1:expired.htm**	Specifies the location of the custom HTML file to use in place of the default login expired page.
Step 7	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Related Topics

Authentication Proxy Web Page Guidelines, on page 1002

Specifying a Redirection URL for Successful Login

Follow these steps to specify a URL to which the user is redirected after authentication, effectively replacing the internal Success HTML page:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip admission proxy http success redirect** *url-string*
4. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**ip admission proxy http success redirect** *url-string* **Example:** Switch(config)# **ip admission proxy http success redirect www.example.com**	Specifies a URL for redirection of the user in place of the default login success page.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Related Topics

Configuring the Web-Based Authentication Parameters

Follow these steps to configure the maximum number of failed login attempts before the client is placed in a watch list for a waiting period:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip admission max-login-attempts** *number*
4. **end**

5. show running-config

6. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**ip admission max-login-attempts** *number* **Example:** Switch(config)# **ip admission max-login-attempts 10**	Sets the maximum number of failed login attempts. The range is 1 to 2147483647 attempts. The default is 5.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring a Web Authentication Local Banner

Follow these steps to configure a local banner on a switch that has web authentication configured.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip admission auth-proxy-banner http** [*banner-text* | *file-path*]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**ip admission auth-proxy-banner http** [*banner-text* | *file-path*] **Example:** Switch(config)# **ip admission auth-proxy-banner http C My Switch C**	Enables the local banner. (Optional) Create a custom banner by entering *C banner-text C* (where *C* is a delimiting character), or *file-path* that indicates a file (for example, a logo or text file) that appears in the banner.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Web-Based Authentication without SVI

You configure the web-based authentication without SVI feature to redirect the HTML login page to the client without creating an IP address in the routing table. These steps are optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **parameter-map type webauth global**
4. **l2-webauth-enabled**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**parameter-map type webauth global** **Example:** Switch (config)# **parameter-map type webauth global**	Creates a parameter map and enters parameter-map webauth configuration mode. The specific configuration commands supported for a global parameter map defined with the global keyword differ from the commands supported for a named parameter map defined with the parameter-map-name argument.
Step 4	**l2-webauth-enabled** **Example:** Switch (config-params-parameter-map)# **l2-webauth-enabled**	Enables the web-based authentication without SVI feature
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

	Command or Action	**Purpose**
Step 6	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Web-Based Authentication with VRF Aware

You configure the web-based authentication with VRF aware to redirect the HTML login page to the client. These steps are optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **parameter-map type webauth global**
4. **webauth-vrf-aware**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**parameter-map type webauth global** **Example:** Switch (config)# **parameter-map type webauth global**	Creates a parameter map and enters parameter-map webauth configuration mode. The specific configuration commands supported for a global parameter map defined with the global keyword differ from the commands supported for a

	Command or Action	Purpose
		named parameter map defined with the parameter-map-name argument.
Step 4	**webauth-vrf-aware** **Example:** Switch (config-params-parameter-map)# **webauth-vrf-aware**	Enables the web-based authentication VRF aware feature on SVI.
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Removing Web-Based Authentication Cache Entries

Follow these steps to remove web-based authentication cache entries:

SUMMARY STEPS

1. **enable**
2. **clear ip auth-proxy cache** {* | *host ip address*}
3. **clear ip admission cache** {* | *host ip address*}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	**Purpose**
Step 2	**clear ip auth-proxy cache** {*| *host ip address*} **Example:** Switch# **clear ip auth-proxy cache 192.168.4.5**	Delete authentication proxy entries. Use an asterisk to delete all cache entries. Enter a specific IP address to delete the entry for a single host.
Step 3	**clear ip admission cache** {*| *host ip address*} **Example:** Switch# **clear ip admission cache 192.168.4.5**	Delete authentication proxy entries. Use an asterisk to delete all cache entries. Enter a specific IP address to delete the entry for a single host.

Monitoring Web-Based Authentication Status

Use the commands in this topic to display the web-based authentication settings for all interfaces or for specific ports.

Table 100: Privileged EXEC show Commands

Command	**Purpose**
show authentication sessions method webauth	Displays the web-based authentication settings for all interfaces for fastethernet, gigabitethernet, or tengigabitethernet
show authentication sessions interface *type slot/port*[**details**]	Displays the web-based authentication settings for the specified interface for fastethernet, gigabitethernet, or tengigabitethernet. In Session Aware Networking mode, use the **show access-session interface** command.

Feature Information for Web-Based Authentication

Release	**Feature Information**
Cisco IOS Release 15.0(2)EX1	This feature is introduced.

Configuring Port-Based Traffic Control

Overview of Port-Based Traffic Control

Port-based traffic control is a set of Layer 2 features on the Cisco Catalyst switches used to filter or block packets at the port level in response to specific traffic conditions. The following port-based traffic control features are supported in the Cisco IOS Release for which this guide is written:

- Storm Control

- Protected Ports

- Port Blocking

- Port Security

- Protocol Storm Protection

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About Storm 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.

How Traffic Activity is Measured

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.

Traffic Patterns

Figure 97: Broadcast Storm Control Example

This example shows broadcast traffic patterns on an interface over a given period of time.
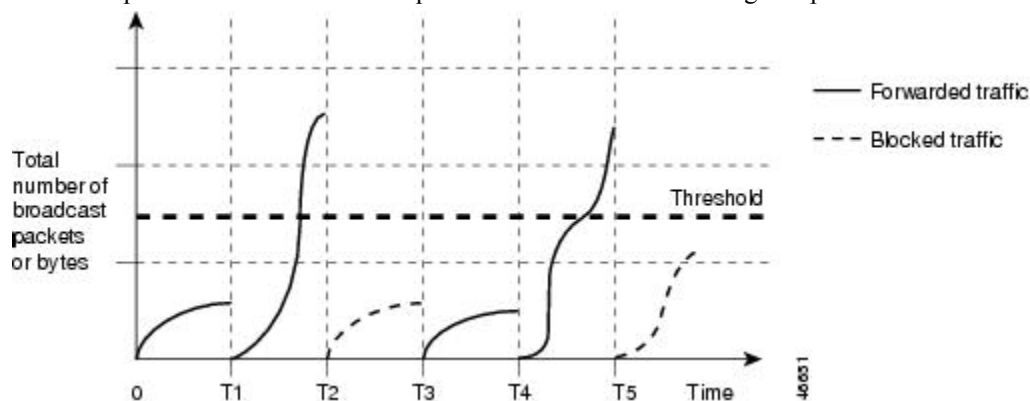
Broadcast traffic being forwarded exceeded the configured threshold between time intervals T1 and T2 and between T4 and T5. When the amount of specified traffic exceeds the threshold, all traffic of that kind is dropped for the next time period. Therefore, broadcast traffic is blocked during the intervals following T2 and T5. At the next time interval (for example, T3), if broadcast traffic does not exceed the threshold, it is again forwarded.

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.

How to Configure Storm Control

Configuring 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.

Follow these steps to storm control and threshold levels:

Before you begin

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.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **storm-control** {**broadcast** | **multicast** | **unicast**} **level** {*level* [*level-low*] | **bps** *bps* [*bps-low*] | **pps** *pps* [*pps-low*]}
5. **storm-control action** {**shutdown** | **trap**}
6. **end**
7. **show storm-control** [*interface-id*] [**broadcast** | **multicast** | **unicast**]
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable**	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Switch> **enable**	• Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	Specifies the interface to be configured, and enter interface configuration mode.
Step 4	**storm-control** {**broadcast** | **multicast** | **unicast**} **level** {*level* [*level-low*] | **bps** *bps* [*bps-low*] | **pps** *pps* [*pps-low*]} **Example:** Switch(config-if)# **storm-control unicast level 87 65**	Configures broadcast, multicast, or unicast storm control. By default, storm control is disabled. The keywords have these meanings: • For *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) For *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. • For **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) For *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 or Action	Purpose
		• For **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) For *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.
Step 5	**storm-control action** {**shutdown** | **trap**} **Example:** Switch(config-if)# **storm-control action 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. • Select the **shutdown** keyword to error-disable the port during a storm. • Select the **trap** keyword to generate an SNMP trap when a storm is detected.
Step 6	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 7	**show storm-control** [*interface-id*] [**broadcast** | **multicast** | **unicast**] **Example:** Switch# **show storm-control gigabitethernet1/0/1 unicast**	Verifies 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.
Step 8	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring 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.

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.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **errdisable detect cause small-frame**
4. **errdisable recovery interval** *interval*
5. **errdisable recovery cause small-frame**
6. **interface** *interface-id*
7. **small-frame violation-rate** *pps*
8. **end**
9. **show interfaces** *interface-id*
10. **show running-config**
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**errdisable detect cause small-frame** **Example:** Switch(config)# **errdisable detect cause small-frame**	Enables the small-frame rate-arrival feature on the switch.
Step 4	**errdisable recovery interval** *interval* **Example:** Switch(config)# **errdisable recovery interval 60**	(Optional) Specifies the time to recover from the specified error-disabled state.
Step 5	**errdisable recovery cause small-frame** **Example:** Switch(config)# **errdisable recovery cause small-frame**	(Optional) Configures the recovery time for error-disabled ports to be automatically re-enabled after they are error disabled by the arrival of small frames 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

	Command or Action	**Purpose**
		control settings propagate to the EtherChannel physical interfaces.
Step 6	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/2**	Enters interface configuration mode, and specify the interface to be configured.
Step 7	**small-frame violation-rate** *pps* **Example:** Switch(config-if)# **small-frame violation rate 10000**	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)
Step 8	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 9	**show interfaces** *interface-id* **Example:** Switch# **show interfaces gigabitethernet1/0/2**	Verifies the configuration.
Step 10	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 11	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About Protected Ports

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.

Because a switch stack represents a single logical switch, Layer 2 traffic is not forwarded between any protected ports in the switch stack, whether they are on the same or different switches in the stack.

Default Protected Port Configuration

The default is to have no protected ports defined.

Protected Ports 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.

How to Configure Protected Ports

Configuring a Protected Port

Before you begin

Protected ports are not pre-defined. This is the task to configure one.

SUMMARY STEPS

1. **enable**

2. **configure terminal**

3. **interface** *interface-id*

4. **switchport protected**

5. **end**

6. **show interfaces** *interface-id* **switchport**

7. **show running-config**

8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	Specifies the interface to be configured, and enter interface configuration mode.
Step 4	**switchport protected** **Example:** Switch(config-if)# **switchport protected**	Configures the interface to be a protected port.
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**show interfaces** *interface-id* **switchport** **Example:** Switch# **show interfaces gigabitethernet1/0/1 switchport**	Verifies your entries.

	Command or Action	**Purpose**
Step 7	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 8	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Monitoring Protected Ports

Table 101: Commands for Displaying Protected Port Settings

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.

Where to Go Next

•

Additional References

Error Message Decoder

Description	**Link**
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information

Release	Feature Information
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About Port Blocking

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.

How to Configure Port Blocking

Blocking Flooded Traffic on an Interface

Before you begin

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.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport block multicast**
5. **switchport block unicast**
6. **end**
7. **show interfaces** *interface-id* **switchport**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:**	Specifies the interface to be configured, and enter interface configuration mode.

	Command or Action	Purpose
	Switch(config)# **interface gigabitethernet1/0/1**	
Step 4	**switchport block multicast** **Example:** Switch(config-if)# **switchport block multicast**	Blocks unknown multicast forwarding out of the port.
Step 5	**switchport block unicast** **Example:** Switch(config-if)# **switchport block unicast**	Blocks unknown unicast forwarding out of the port.
Step 6	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 7	**show interfaces** *interface-id* **switchport** **Example:** Switch# **show interfaces gigabitethernet1/0/1** **switchport**	Verifies your entries.
Step 8	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 9	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Monitoring Port Blocking

Table 102: Commands for Displaying Port Blocking Settings

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.

Where to Go Next

-

Additional References

Related Documents

Related Topic	Document Title

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title

MIBs

MIB	MIBs Link
	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature Information

Release	Feature Information
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Prerequisites for Port Security

> **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.

Restrictions for Port Security

The maximum number of secure MAC addresses that you can configure on a switch or switch stack 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.

Information About Port Security

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.

Related Topics

Types of Secure MAC Addresses

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.

Sticky Secure MAC Addresses

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. 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.

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.

- Running diagnostic tests with port security enabled.

You can configure the interface for one of three 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.

> **Note** 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. 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 re-enable 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

This table shows the violation mode and the actions taken when you configure an interface for port security.

Table 103: Security Violation Mode Actions

Violation Mode	Traffic is forwarded [18]	Sends SNMP trap	Sends syslog message	Displays error message [19]	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 [20]

[18] Packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses.

[19] The switch returns an error message if you manually configure an address that would cause a security violation.

[20] Shuts down only the VLAN on which the violation occurred.

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.

Related Topics

Port Security and Switch Stacks

When a switch joins a stack, the new switch will get the configured secure addresses. All dynamic secure addresses are downloaded by the new stack member from the other stack members.

When a switch (either the active switch or a stack member) leaves the stack, the remaining stack members are notified, and the secure MAC addresses configured or learned by that switch are deleted from the secure MAC address table.

Default Port Security Configuration

Table 104: 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).

-

Note	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 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.

This table summarizes port security compatibility with other port-based features.

Table 105: Port Security Compatibility with Other Switch Features

Type of Port or Feature on Port	Compatible with Port Security
DTP [21] port [22]	No
Trunk port	Yes
Dynamic-access port [23]	No
Routed port	No
SPAN source port	Yes
SPAN destination port	No
EtherChannel	Yes
Tunneling port	Yes
Protected port	Yes
IEEE 802.1x port	Yes
Voice VLAN port [24]	Yes
IP source guard	Yes

Type of Port or Feature on Port	Compatible with Port Security
Dynamic Address Resolution Protocol (ARP) inspection	Yes
Flex Links	Yes

[21] DTP=Dynamic Trunking Protocol

[22] A port configured with the **switchport mode dynamic** interface configuration command.

[23] A VLAN Query Protocol (VQP) port configured with the **switchport access vlan dynamic** interface configuration command.

[24] 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.

Overview of Port-Based Traffic Control

Port-based traffic control is a set of Layer 2 features on the Cisco Catalyst switches used to filter or block packets at the port level in response to specific traffic conditions. The following port-based traffic control features are supported in the Cisco IOS Release for which this guide is written:

- Storm Control

- Protected Ports

- Port Blocking

- Port Security

- Protocol Storm Protection

How to Configure Port Security

Enabling and Configuring Port Security

Before you begin

This task restricts input to an interface by limiting and identifying MAC addresses of the stations allowed to access the port:

SUMMARY STEPS

1. **enable**

2. **configure terminal**

3. **interface** *interface-id*

4. **switchport mode** {**access** | **trunk**}

5. **switchport voice vlan** *vlan-id*

6. **switchport port-security**

7. **switchport port-security [maximum** *value* [**vlan** {*vlan-list* | {**access** | **voice**}}]]

8. **switchport port-security violation** {**protect** | **restrict** | **shutdown** | **shutdown vlan**}

9. **switchport port-security [mac-address** *mac-address* [**vlan** {*vlan-id* | {**access** | **voice**}}]
10. **switchport port-security mac-address sticky**
11. **switchport port-security mac-address sticky** [*mac-address* | **vlan** {*vlan-id* | {**access** | **voice**}}]
12. **end**
13. **show port-security**
14. **show running-config**
15. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	Specifies the interface to be configured, and enter interface configuration mode.
Step 4	**switchport mode** {**access**	**trunk**} **Example:** Switch(config-if)# **switchport mode access**	Sets the interface switchport mode as access or trunk; an interface in the default mode (dynamic auto) cannot be configured as a secure port.
Step 5	**switchport voice vlan** *vlan-id* **Example:** Switch(config-if)# **switchport voice vlan 22**	Enables voice VLAN on a port. vlan-id—Specifies the VLAN to be used for voice traffic.
Step 6	**switchport port-security** **Example:** Switch(config-if)# **switchport port-security**	Enable port security on the interface.
Step 7	**switchport port-security [maximum** *value* [**vlan** {*vlan-list*	{**access**	**voice**}}]]	(Optional) Sets the maximum number of secure MAC addresses for the interface. The maximum number of

	Command or Action	Purpose
	Example: Switch(config-if)# **switchport port-security maximum 20**	secure MAC addresses that you can configure on a switch or switch stack 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, you can set 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.
Step 8	**switchport port-security violation** {**protect** | **restrict** | **shutdown** | **shutdown vlan**} **Example:** Switch(config-if)# **switchport port-security violation restrict**	(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

	Command or Action	**Purpose**
		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**—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.
		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 re-enable it by entering the **shutdown** and **no shutdown** interface configuration commands or by using the **clear errdisable interface vlan** privileged EXEC command.

Step 9	**switchport port-security [mac-address** *mac-address* **[vlan** {*vlan-id* | {**access** | **voice**}}]**
	
	Example:
	
	`Switch(config-if)# switchport port-security`
	`mac-address 00:A0:C7:12:C9:25 vlan 3 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, you can specify 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.

	Command or Action	Purpose
		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.
Step 10	**switchport port-security mac-address sticky** **Example:** Switch(config-if)# **switchport port-security mac-address sticky**	(Optional) Enables sticky learning on the interface.
Step 11	**switchport port-security mac-address sticky** [*mac-address* | **vlan** {*vlan-id* | {**access** | **voice**}}] **Example:** Switch(config-if)# **switchport port-security mac-address sticky 00:A0:C7:12:C9:25 vlan 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, you can specify 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.
Step 12	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 13	**show port-security** **Example:**	Verifies your entries.

	Command or Action	Purpose
	Switch# **show port-security**	
Step 14	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 15	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Related Topics

Enabling and Configuring Port Security Aging

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.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport port-security aging** {**static** | **time** *time* | **type** {**absolute** | **inactivity**}}
5. **end**
6. **show port-security** [**interface** *interface-id*] [**address**]
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	**Purpose**
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	Specifies the interface to be configured, and enter interface configuration mode.
Step 4	**switchport port-security aging** {**static** | **time** *time* | **type** {**absolute** | **inactivity**}} **Example:** Switch(config-if)# **switchport port-security aging time 120**	Enables or disable static aging for the secure port, or set the aging time or type. **Note** The switch does not support port security aging of sticky secure addresses. Enter **static** to enable aging for statically configured secure addresses on this port. For *time*, specifies the aging time for this port. The valid range is from 0 to 1440 minutes. For **type**, select one of these keywords: • **absolute**—Sets the aging type as absolute aging. 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**—Sets the aging type as inactivity aging. 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.
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**show port-security** [**interface** *interface-id*] [**address**] **Example:** Switch# **show port-security interface gigabitethernet1/0/1**	Verifies your entries.
Step 7	**show running-config** **Example:**	Verifies your entries.

	Command or Action	Purpose
	`Switch# `**`show running-config`**	
Step 8	**copy running-config startup-config** **Example:** `Switch# `**`copy running-config startup-config`**	(Optional) Saves your entries in the configuration file.

Related Topics

Configuration Examples for Port Security

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 gigabitethernet1/0/1
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 gigabitethernet1/0/2
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport port-security
Switch(config-if)# switchport port-security mac-address 0000.0200.0004 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 tengigabitethernet1/0/1
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
```

Related Topics

Port Security, on page 1036

Enabling and Configuring Port Security, on page 1041

Additional References

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About Protocol Storm Protection

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 re-enabling 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 Flexlink interfaces

Default Protocol Storm Protection Configuration

Protocol storm protection is disabled by default. When it is enabled, auto-recovery of the virtual port is disabled by default.

How to Configure Protocol Storm Protection

Enabling Protocol Storm Protection

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **psp** {**arp** | **dhcp** | **igmp**} pps *value*

4. **errdisable detect cause psp**
5. **errdisable recovery interval** *time*
6. **end**
7. **show psp config** {**arp** | **dhcp** | **igmp**}

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**psp** {**arp**	**dhcp**	**igmp**} pps *value* **Example:** Switch(config)# **psp dhcp pps 35**	Configures protocol storm protection for ARP, IGMP, or DHCP. For *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.
Step 4	**errdisable detect cause psp** **Example:** Switch(config)# **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.
Step 5	**errdisable recovery interval** *time* **Example:** Switch	(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.
Step 6	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 7	**show psp config** {**arp**	**dhcp**	**igmp**} **Example:**	Verifies your entries.

Command or Action	Purpose
Switch# **show psp config dhcp**	

Monitoring Protocol Storm Protection

Command	Purpose
show psp config {**arp** | **dhcp** | **igmp**}	Verify your entries.

Additional References

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

C H A P T E R 47

Configuring IPv6 First Hop Security

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for First Hop Security in IPv6

- You have configured the necessary IPv6 enabled SDM template.

- You should be familiar with the IPv6 neighbor discovery feature.

Restrictions for First Hop Security in IPv6

- The following restrictions apply when applying FHS policies to EtherChannel interfaces (Port Channels):

  - A physical port with an FHS policy attached cannot join an EtherChannel group.

> > > • An FHS policy cannot be attached to an physical port when it is a member of an EtherChannel group.

> > • By default, a snooping policy has a security-level of guard. When such a snooping policy is configured on an access switch, external IPv6 Router Advertisement (RA) or Dynamic Host Configuration Protocol for IPv6 (DHCPv6) server packets are blocked, even though the uplink port facing the router or DHCP server/relay is configured as a trusted port. To allow IPv6 RA or DHCPv6 server messages, do the following:

> > > • Apply an IPv6 RA-guard policy (for RA) or IPv6 DHCP-guard policy (for DHCP server messages) on the uplink port.

> > > • Configure a snooping policy with a lower security-level, for example glean or inspect. However; configuring a lower security level is not recommended with such a snooping policy, because benefits of First Hop security features are not effective.

Information about First Hop Security in IPv6

First Hop Security in IPv6 (FHS IPv6) is a set of IPv6 security features, the policies of which can be attached to a physical interface, an EtherChannel interface, or a VLAN. An IPv6 software policy database service stores and accesses these policies. When a policy is configured or modified, the attributes of the policy are stored or updated in the software policy database, then applied as was specified. The following IPv6 policies are currently supported:

- IPv6 Snooping Policy—IPv6 Snooping Policy acts as a container policy that enables most of the features available with FHS in IPv6.

- IPv6 FHS Binding Table Content—A database table of IPv6 neighbors connected to the switch is created from information sources such as Neighbor Discovery (ND) protocol snooping. This database, or binding, table is used by various IPv6 guard features (such as IPv6 ND Inspection) to validate the link-layer address (LLA), the IPv4 or IPv6 address, and prefix binding of the neighbors to prevent spoofing and redirect attacks.
- IPv6 Neighbor Discovery Inspection—IPv6 ND inspection learns and secures bindings for stateless autoconfiguration addresses in Layer 2 neighbor tables. IPv6 ND inspection analyzes neighbor discovery messages in order to build a trusted binding table database and IPv6 neighbor discovery messages that do not conform are dropped. An ND message is considered trustworthy if its IPv6-to-Media Access Control (MAC) mapping is verifiable.

  This feature mitigates some of the inherent vulnerabilities of the ND mechanism, such as attacks on DAD, address resolution, router discovery, and the neighbor cache.

- IPv6 Router Advertisement Guard—The IPv6 Router Advertisement (RA) guard feature enables the network administrator to block or reject unwanted or rogue RA guard messages that arrive at the network switch platform. RAs are used by routers to announce themselves on the link. The RA Guard feature analyzes the RAs and filters out bogus RAs sent by unauthorized routers. In host mode, all router advertisement and router redirect messages are disallowed on the port. The RA guard feature compares configuration information on the Layer 2 device with the information found in the received RA frame. Once the Layer 2 device has validated the content of the RA frame and router redirect frame against the configuration, it forwards the RA to its unicast or multicast destination. If the RA frame content is not validated, the RA is dropped.

- IPv6 DHCP Guard—The IPv6 DHCP Guard feature blocks reply and advertisement messages that come from unauthorized DHCPv6 servers and relay agents. IPv6 DHCP guard can prevent forged messages from being entered in the binding table and block DHCPv6 server messages when they are received on ports that are not explicitly configured as facing a DHCPv6 server or DHCP relay. To use this feature, configure a policy and attach it to an interface or a VLAN. To debug DHCP guard packets, use the **debug ipv6 snooping dhcp-guard** privileged EXEC command.

- IPv6 Source Guard—Like IPv4 Source Guard, IPv6 Source Guard validates the source address or prefix to prevent source address spoofing.

  A source guard programs the hardware to allow or deny traffic based on source or destination addresses. It deals exclusively with data packet traffic.

  The IPv6 source guard feature provides the ability to use the IPv6 binding table to install PACLs to prevent a host from sending packets with an invalid IPv6 source address.

  To debug source-guard packets, use the debug ipv6 snooping source-guard privileged EXEC command.

  > **Note** The IPv6 PACL feature is supported only in the ingress direction; it is not supported in the egress direction.

  The following restrictions apply:

  - An FHS policy cannot be attached to an physical port when it is a member of an EtherChannel group.

  - When IPv6 source guard is enabled on a switch port, NDP or DHCP snooping must be enabled on the interface to which the switch port belongs. Otherwise, all data traffic from this port will be blocked.

  - An IPv6 source guard policy cannot be attached to a VLAN. It is supported only at the interface level.

  - You cannot use IPv6 Source Guard and Prefix Guard together. When you attach the policy to an interface, it should be "validate address" or "validate prefix" but not both.

  - PVLAN and Source/Prefix Guard cannot be applied together.

  For more information on IPv6 Source Guard, see the IPv6 Source Guard chapter of the Cisco IOS IPv6 Configuration Guide Library on Cisco.com.

- IPv6 Prefix Guard—The IPv6 prefix guard feature works within the IPv6 source guard feature, to enable the device to deny traffic originated from non-topologically correct addresses. IPv6 prefix guard is often used when IPv6 prefixes are delegated to devices (for example, home gateways) using DHCP prefix delegation. The feature discovers ranges of addresses assigned to the link and blocks any traffic sourced with an address outside this range.

  For more information on IPv6 Prefix Guard, see the IPv6 Prefix Guard chapter of the Cisco IOS IPv6 Configuration Guide Library on Cisco.com.

- IPv6 Destination Guard—The IPv6 destination guard feature works with IPv6 neighbor discovery to ensure that the device performs address resolution only for those addresses that are known to be active on the link. It relies on the address glean functionality to populate all destinations active on the link into the binding table and then blocks resolutions before they happen when the destination is not found in the binding table.

| **Note** | IPv6 Destination Guard is recommended only on Layer 3. It is not recommended on Layer2. |

For more information about IPv6 Destination Guard, see the IPv6 Destination Guard chapter of the Cisco IOS IPv6 Configuration Guide Library on Cisco.com.

- IPv6 Neighbor Discovery Multicast Suppress—The IPv6 Neighbor Discovery multicast suppress feature is an IPv6 snooping feature that runs on a switch or a wireless controller and is used to reduce the amount of control traffic necessary for proper link operations.

- DHCPv6 Relay—Lightweight DHCPv6 Relay Agent—The DHCPv6 Relay—Lightweight DHCPv6 Relay Agent feature allows relay agent information to be inserted by an access node that performs a link-layer bridging (non-routing) function. Lightweight DHCPv6 Relay Agent (LDRA) functionality can be implemented in existing access nodes, such as DSL access multiplexers (DSLAMs) and Ethernet switches, that do not support IPv6 control or routing functions. LDRA is used to insert relay-agent options in DHCP version 6 (DHCPv6) message exchanges primarily to identify client-facing interfaces. LDRA functionality can be enabled on an interface and on a VLAN.

For more information about DHCPv6 Relay, See the DHCPv6 Relay—Lightweight DHCPv6 Relay Agent section of the IP Addressing: DHCP Configuration Guide, Cisco IOS Release 15.1SG.

How to Configure an IPv6 Snooping Policy

Beginning in privileged EXEC mode, follow these steps to configure IPv6 Snooping Policy :

SUMMARY STEPS

1. **configure terminal**
2. **ipv6 snooping policy** *policy-name*
3. {[**default**] | [**device-role** {**node** | **switch**}] | [**limit address-count** *value*] | [**no**] | [**protocol** {**dhcp** | **ndp**}] | [**security-level** {**glean** | **guard** | **inspect**}] | [**tracking** {**disable** [**stale-lifetime** [*seconds* | **infinite**] | **enable** [**reachable-lifetime** [*seconds* | **infinite**] }] | [**trusted-port**] }
4. **end**
5. **show ipv6 snooping policy** *policy-name*

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters the global configuration mode.
Step 2	**ipv6 snooping policy** *policy-name* **Example:** Switch(config)# **ipv6 snooping policy example_policy**	Creates a snooping policy and enters IPv6 Snooping Policy Configuration mode.

	Command or Action	Purpose
Step 3	{[**default**]	[**device-role** {**node**	**switch**}]	[**limit address-count** *value*]	[**no**]	[**protocol** {**dhcp**	**ndp**}]	[**security-level** {**glean**	**guard**	**inspect**}]	[**tracking** {**disable** [**stale-lifetime** [*seconds*	**infinite**]	**enable** [**reachable-lifetime** [*seconds*	**infinite**] }]	[**trusted-port**] } **Example:** Switch**(config-ipv6-snooping)# security-level inspect** **Example:** Switch**(config-ipv6-snooping)# trusted-port**	Enables data address gleaning, validates messages against various criteria, specifies the security level for messages. • (Optional) **default**—Sets all to default options. • (Optional) **device-role**{**node**]	**switch**}—Specifies the role of the device attached to the port. Default is **node**. • (Optional) **limit address-count** *value*—Limits the number of addresses allowed per target. • (Optional) **no**—Negates a command or sets it to defaults. • (Optional) **protocol**{**dhcp**	**ndp**}—Specifies which protocol should be redirected to the snooping feature for analysis. The default, is **dhcp** and **ndp**. To change the default, use the **no protocol** command. • (Optional) **security-level**{**glean**|**guard**|**inspect**}—Specifies the level of security enforced by the feature. Default is **guard.** **glean**—Gleans addresses from messages and populates the binding table without any verification. **guard**—Gleans addresses and inspects messages. In addition, it rejects RA and DHCP server messages. This is the default option. **inspect**—Gleans addresses, validates messages for consistency and conformance, and enforces address ownership. • (Optional) **tracking** {**disable**	**enable**}—Overrides the default tracking behavior and specifies a tracking option. • (Optional) **trusted-port**—Sets up a trusted port. It disables the guard on applicable targets. Bindings learned through a trusted port have preference over bindings learned through any other port. A trusted port is given preference in case of a collision while making an entry in the table.
Step 4	**end** **Example:** Switch(config-ipv6-snooping)# **exit**	Exits configuration modes to Privileged EXEC mode.
Step 5	**show ipv6 snooping policy** *policy-name* **Example:**	Displays the snooping policy configuration.

Command or Action	Purpose
Switch#**show ipv6 snooping policy example_policy**	

What to do next

Attach an IPv6 Snooping policy to interfaces or VLANs.

How to Attach an IPv6 Snooping Policy to an Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Snooping policy on an interface or VLAN:

SUMMARY STEPS

1. **configure terminal**
2. **interface** Interface_type *stack/module/port*
3. **switchport**
4. **ipv6 snooping** [**attach-policy** *policy_name* [**vlan** {*vlan_id* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids*}] | **vlan** {*vlan_id* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
5. **do show running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters the global configuration mode.
Step 2	**interface** Interface_type *stack/module/port* **Example:** Switch(config)# **interface gigabitethernet 1/1/4**	Specifies an interface type and identifier; enters the interface configuration mode.
Step 3	**switchport** **Example:** Switch(config-if)# **switchport**	Enters the Switchport mode. **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. The command prompt displays as (config-if)# in Switchport configuration mode.

	Command or Action	Purpose

Step 4	**ipv6 snooping** [**attach-policy** *policy_name* [**vlan** {*vlan_id* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids*}] | **vlan** {*vlan_id* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
	
	Example:
	`Switch(config-if)# ipv6 snooping`
	
	`or`
	
	`Switch(config-if)# ipv6 snooping attach-policy`
	`example_policy`
	
	`or`
	`Switch(config-if)# ipv6 snooping vlan 111,112`
	
	`or`
	
	`Switch(config-if)# ipv6 snooping attach-policy`
	`example_policy vlan 111,112`
		Attaches a custom ipv6 snooping policy to the interface or the specified VLANs on the interface. To attach the default policy to the interface, use the **ipv6 snooping** command without the **attach-policy** keyword. To attach the default policy to VLANs on the interface, use the **ipv6 snooping vlan** command. The default policy is, security-level **guard**, device-role **node**, protocol **ndp** and **dhcp.**

Step 5	**do show running-config** **Example:** `Switch#(config-if)#  do show running-config`	Verifies that the policy is attached to the specified interface without exiting the interface configuration mode.

How to Attach an IPv6 Snooping Policy to a Layer 2 EtherChannel Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Snooping policy on an EtherChannel interface or VLAN:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**interface range** *Interface_name* **Example:** `Switch(config)#  interface range Po11`	Specify the port-channel interface name assigned when the EtherChannel was created. Enters the interface range configuration mode. **Tip** Enter the **do show interfaces summary** command for quick reference to interface names and types.
Step 3	**ipv6 snooping** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_id*s | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]	Attaches the IPv6 Snooping policy to the interface or the specified VLANs on that interface. The default policy is attached if the **attach-policy** option is not used.

	Command or Action	**Purpose**

	Example:
	
	`Switch(config-if-range)# ipv6 snooping`
	`attach-policy example_policy`
	
	`or`
	
	`Switch(config-if-range)# ipv6 snooping`
	`attach-policy example_policy vlan 222,223,224`
	
	`or`
	
	`Switch(config-if-range)#ipv6 snooping vlan 222,`
	`223,224`	

Step 4	**do show running-config**
	interface_portchannel_interface_name_
	
	Example:
	
	`Switch#(config-if-range)#  do show running-config`
	` int po11`
		Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 Snooping Policy to VLANs Globally

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Snooping Policy to VLANs across multiple interfaces:

SUMMARY STEPS

1. **configure terminal**
2. **vlan configuration** _vlan_list_
3. **ipv6 snooping** [**attach-policy** _policy_name_]
4. **do show running-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.
Step 2	**vlan configuration** _vlan_list_ **Example:** `Switch(config)#  vlan configuration 333`	Specifies the VLANs to which the IPv6 Snooping policy will be attached ; enters the VLAN interface configuration mode.
Step 3	**ipv6 snooping** [**attach-policy** _policy_name_] **Example:**	Attaches the IPv6 Snooping policy to the specified VLANs across all switch and stack interfaces. The default policy is attached if the **attach-policy** option is not used. The default

	Command or Action	Purpose
	Switch(config-vlan-config)#`ipv6 snooping attach-policy example_policy`	policy is, security-level **guard**, device-role **node**, protocol **ndp** and **dhcp.**
Step 4	**do show running-config** **Example:** Switch#(config-if)# `do show running-config`	Verifies that the policy is attached to the specified VLANs without exiting the interface configuration mode.

How to Configure the IPv6 Binding Table Content

Beginning in privileged EXEC mode, follow these steps to configure IPv6 Binding Table Content :

SUMMARY STEPS

1. **configure terminal**
2. [**no**] **ipv6 neighbor binding** [**vlan** *vlan-id* {*ipv6-address* **interface** interface_type *stack/module/port hw_address* [**reachable-lifetimevalue** [*seconds* | **default** | **infinite**] | [**tracking**{ [default | disable] [**reachable-lifetimevalue** [*seconds* | **default** | **infinite**] | [**enable** [**reachable-lifetimevalue** [*seconds* | **default** | **infinite**] | [**retry-interval** {*seconds*| **default** [**reachable-lifetimevalue** [*seconds* | **default** | **infinite**] }]
3. [**no**] **ipv6 neighbor binding max-entries** *number* [**mac-limit** *number* | **port-limit** *number* [**mac-limit** *number*] | **vlan-limit** *number* [[**mac-limit** *number*] | [**port-limit** *number* [**mac-limit***number*]]]]
4. **ipv6 neighbor binding logging**
5. **exit**
6. **show ipv6 neighbor binding**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# `configure terminal`	Enters the global configuration mode.
Step 2	[**no**] **ipv6 neighbor binding** [**vlan** *vlan-id* {*ipv6-address* **interface** interface_type *stack/module/port hw_address* [**reachable-lifetimevalue** [*seconds*	**default**	**infinite**]	[**tracking**{ [default	disable] [**reachable-lifetimevalue** [*seconds*	**default**	**infinite**]	[**enable** [**reachable-lifetimevalue** [*seconds*	**default**	**infinite**]	[**retry-interval** {*seconds*| **default** [**reachable-lifetimevalue** [*seconds*	**default**	**infinite**] }] **Example:** Switch(config)# `ipv6 neighbor binding`	Adds a static entry to the binding table database.

	Command or Action	**Purpose**
Step 3	[**no**] **ipv6 neighbor binding max-entries** *number* [**mac-limit** *number*	**port-limit** *number* [**mac-limit** *number*]	**vlan-limit** *number* [[**mac-limit** *number*]	[**port-limit** *number* [**mac-limit***number*]]]] **Example:** Switch(config)# **ipv6 neighbor binding max-entries 30000**	Specifies the maximum number of entries that are allowed to be inserted in the binding table cache.
Step 4	**ipv6 neighbor binding logging** **Example:** Switch(config)# **ipv6 neighbor binding logging**	Enables the logging of binding table main events.
Step 5	**exit** **Example:** Switch(config)# **exit**	Exits global configuration mode, and places the router in privileged EXEC mode.
Step 6	**show ipv6 neighbor binding** **Example:** Switch# **show ipv6 neighbor binding**	Displays contents of a binding table.

How to Configure an IPv6 Neighbor Discovery Inspection Policy

Beginning in privileged EXEC mode, follow these steps to configure an IPv6 ND Inspection Policy:

SUMMARY STEPS

1. **configure terminal**
2. [**no**]**ipv6 nd inspection policy** *policy-name*
3. **device-role** {**host** | **monitor** | **router** | **switch**}
4. **drop-unsecure**
5. **limit address-count** *value*
6. **sec-level minimum** *value*
7. **tracking** {**enable** [**reachable-lifetime** {*value* | **infinite**}] | **disable** [**stale-lifetime** {*value* | **infinite**}]}
8. **trusted-port**
9. **validate source-mac**
10. **no** {**device-role** | **drop-unsecure** | **limit address-count** | **sec-level minimum** | **tracking** | **trusted-port** | **validate source-mac**}
11. **default** {**device-role** | **drop-unsecure** | **limit address-count** | **sec-level minimum** | **tracking** | **trusted-port** | **validate source-mac**}
12. **do show ipv6 nd inspection policy** *policy_name*

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters the global configuration mode.
Step 2	[**no**]**ipv6 nd inspection policy** *policy-name* **Example:** Switch(config)# **ipv6 nd inspection policy example_policy**	Specifies the ND inspection policy name and enters ND Inspection Policy configuration mode.
Step 3	**device-role** {**host**	**monitor**	**router**	**switch**} **Example:** Switch(config-nd-inspection)# **device-role switch**	Specifies the role of the device attached to the port. The default is **host**.
Step 4	**drop-unsecure** **Example:** Switch(config-nd-inspection)# **drop-unsecure**	Drops messages with no or invalid options or an invalid signature.
Step 5	**limit address-count** *value* **Example:** Switch(config-nd-inspection)# **limit address-count 1000**	Enter 1–10,000.
Step 6	**sec-level minimum** *value* **Example:** Switch(config-nd-inspection)# **limit address-count 1000**	Specifies the minimum security level parameter value when Cryptographically Generated Address (CGA) options are used.
Step 7	**tracking** {**enable** [**reachable-lifetime** {*value*	**infinite**}]	**disable** [**stale-lifetime** {*value*	**infinite**}]} **Example:** Switch(config-nd-inspection)# **tracking disable stale-lifetime infinite**	Overrides the default tracking policy on a port.
Step 8	**trusted-port** **Example:** Switch(config-nd-inspection)# **trusted-port**	Configures a port to become a trusted port.
Step 9	**validate source-mac** **Example:** Switch(config-nd-inspection)# **validate source-mac**	Checks the source media access control (MAC) address against the link-layer address.
Step 10	**no** {**device-role**	**drop-unsecure**	**limit address-count**	**sec-level minimum**	**tracking**	**trusted-port**	**validate source-mac**}	Remove the current configuration of a parameter with the **no** form of the command.

	Command or Action	Purpose
	Example: Switch(config-nd-inspection)# **no validate source-mac**	
Step 11	**default** {**device-role** | **drop-unsecure** | **limit address-count** | **sec-level minimum** | **tracking** | **trusted-port** | **validate source-mac**} **Example:** Switch(config-nd-inspection)# **default limit address-count**	Restores configuration to the default values.
Step 12	**do show ipv6 nd inspection policy** *policy_name* **Example:** Switch(config-nd-inspection)# **do show ipv6 nd inspection policy example_policy**	Verifies the ND Inspection Configuration without exiting ND inspection configuration mode.

How to Attach an IPv6 Neighbor Discovery Inspection Policy to an Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 ND Inspection policy to an interface or VLANs on an interface :

SUMMARY STEPS

1. **configure terminal**
2. **interface** Interface_type *stack/module/port*
3. **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* \| **add** *vlan_ids* \| **except** *vlan_ids* \| **none** \| **remove** *vlan_ids* \| **all**}] \| **vlan** [{*vlan_id*s \| **add** *vlan_ids* \| **except***vlan_ids* \| **none** \| **remove** *vlan_ids* \| **all**}]
4. **do show running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters the global configuration mode.
Step 2	**interface** Interface_type *stack/module/port* **Example:** Switch(config)# **interface gigabitethernet 1/1/4**	Specifies an interface type and identifier; enters the interface configuration mode.
Step 3	**ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_id*s | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]	Attaches the Neighbor Discovery Inspection policy to the interface or the specified VLANs on that interface. The default policy is attached if the **attach-policy** option is not used.

	Command or Action	Purpose

	Example:
	
	`Switch(config-if)# ipv6 nd inspection attach-policy example_policy`
	
	`or`
	
	`Switch(config-if)# ipv6 nd inspection attach-policy example_policy vlan 222,223,224`
	
	`or`
	
	`Switch(config-if)# ipv6 nd inspection vlan 222, 223,224`	

Step 4	**do show running-config** **Example:** `Switch#(config-if)#  do show running-config`	Verifies that the policy is attached to the specified interface without exiting the interface configuration mode.

How to Attach an IPv6 Neighbor Discovery Inspection Policy to a Layer 2 EtherChannel Interface

Starting with Cisco IOS XE Amsterdam 17.1.1 the IPv6 ND Inspection feature is deprecated and the SISF-based device tracking feature replaces it. For the corresponding replacement task, see *Attaching a Device Tracking Policy to an Interface* under the *Configuring SISF-Based Device Tracking* chapter in this document.

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Neighbor Discovery Inspection policy on an EtherChannel interface or VLAN:

SUMMARY STEPS

1. **configure terminal**
2. **interface range** *Interface_name*
3. **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]] | **vlan** [{*vlan_id*s | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
4. **do show running-config interface***portchannel_interface_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters the global configuration mode.

	Command or Action	**Purpose**
Step 2	**interface range** *Interface_name* **Example:** Switch(config)# **interface Po11**	Specify the port-channel interface name assigned when the EtherChannel was created. Enters the interface range configuration mode. **Tip** Enter the **do show interfaces summary** command for quick reference to interface names and types.
Step 3	**ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_id*s | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] **Example:** Switch(config-if-range)# **ipv6 nd inspection attach-policy example_policy** or Switch(config-if-range)# **ipv6 nd inspection attach-policy example_policy vlan 222,223,224** **or** Switch(config-if-range)#**ipv6 nd inspection vlan 222, 223,224**	Attaches the ND Inspection policy to the interface or the specified VLANs on that interface. The default policy is attached if the **attach-policy** option is not used.
Step 4	**do show running-config interface***portchannel_interface_name* **Example:** Switch#(config-if-range)# **do show running-config int po11**	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 Neighbor Discovery Inspection Policy to VLANs Globally

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 ND Inspection policy to VLANs across multiple interfaces:

SUMMARY STEPS

1. **configure terminal**
2. **vlan configuration** *vlan_list*
3. **ipv6 nd inspection** [**attach-policy** *policy_name*]
4. **do show running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters the global configuration mode.
Step 2	**vlan configuration** *vlan_list* **Example:** Switch(config)# **vlan configuration 334**	Specifies the VLANs to which the IPv6 Snooping policy will be attached ; enters the VLAN interface configuration mode.
Step 3	**ipv6 nd inspection** [**attach-policy** *policy_name*] **Example:** Switch(config-vlan-config)#**ipv6 nd inspection attach-policy example_policy**	Attaches the IPv6 Neighbor Discovery policy to the specified VLANs across all switch and stack interfaces. The default policy is attached if the **attach-policy** option is not used. The default policy is, device-role **host**, no drop-unsecure, limit address-count disabled, sec-level minimum is disabled, tracking is disabled, no trusted-port, no validate source-mac.
Step 4	**do show running-config** **Example:** Switch#(config-if)# **do show running-config**	Confirms that the policy is attached to the specified VLANs without exiting the configuration mode.

How to Configure an IPv6 Router Advertisement Guard Policy

Beginning in privileged EXEC mode, follow these steps to configure an IPv6 Router Advertisement policy :

SUMMARY STEPS

1. **configure terminal**
2. [**no**]**ipv6 nd raguard policy** *policy-name*
3. [**no**]**device-role** {**host** | **monitor** | **router** | **switch**}
4. [**no**]**hop-limit** {**maximum** | **minimum**} *value*
5. [**no**]**managed-config-flag** {**off** | **on**}
6. [**no**]**match** {**ipv6 access-list** *list* | **ra prefix-list** *list*}
7. [**no**]**other-config-flag** {**on** | **off**}
8. [**no**]**router-preference maximum** {**high** | **medium** | **low**}
9. [**no**]**trusted-port**
10. **default** {**device-role** | **hop-limit** {**maximum** | **minimum**} | **managed-config-flag** | **match** {**ipv6 access-list** | **ra prefix-list** } | **other-config-flag** | **router-preference maximum**| **trusted-port**}
11. **do show ipv6 nd raguard policy** *policy_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters the global configuration mode.
Step 2	[**no**]**ipv6 nd raguard policy** *policy-name* **Example:** Switch(config)# **ipv6 nd raguard policy example_policy**	Specifies the RA Guard policy name and enters RA Guard Policy configuration mode.
Step 3	[**no**]**device-role** {**host**	**monitor**	**router**	**switch**} **Example:** Switch(config-nd-raguard)# **device-role switch**	Specifies the role of the device attached to the port. The default is **host**.
Step 4	[**no**]**hop-limit** {**maximum**	**minimum**} *value* **Example:** Switch(config-nd-raguard)# **hop-limit maximum 33**	(1–255) Range for Maximum and Minimum Hop Limit values. Enables filtering of Router Advertisement messages by the Hop Limit value. A rogue RA message may have a low Hop Limit value (equivalent to the IPv4 Time to Live) that when accepted by the host, prevents the host from generating traffic to destinations beyond the rogue RA message generator. An RA message with an unspecified Hop Limit value is blocked. If not configured, this filter is disabled. Configure **minimum** to block RA messages with Hop Limit values lower than the value you specify. Configure **maximum** to block RA messages with Hop Limit values greater than the value you specify.
Step 5	[**no**]**managed-config-flag** {**off**	**on**} **Example:** Switch(config-nd-raguard)# **managed-config-flag on**	Enables filtering of Router Advertisement messages by the Managed Address Configuration, or "M" flag field. A rouge RA message with an M field of 1 can cause a host to use a rogue DHCPv6 server. If not configured, this filter is disabled. **On**—Accepts and forwards RA messages with an M value of 1, blocks those with 0. **Off**—Accepts and forwards RA messages with an M value of 0, blocks those with 1.
Step 6	[**no**]**match** {**ipv6 access-list** *list*	**ra prefix-list** *list*} **Example:** Switch(config-nd-raguard)# **match ipv6 access-list example_list**	Matches a specified prefix list or access list.

	Command or Action	**Purpose**
Step 7	**[no]other-config-flag** {**on** | **off**} **Example:** Switch(config-nd-raguard)# **other-config-flag on**	Enables filtering of Router Advertisement messages by the Other Configuration, or "O" flag field. A rouge RA message with an O field of 1 can cause a host to use a rogue DHCPv6 server. If not configured, this filter is disabled. **On**—Accepts and forwards RA messages with an O value of 1, blocks those with 0. **Off**—Accepts and forwards RA messages with an O value of 0, blocks those with 1.
Step 8	**[no]router-preference maximum** {**high** | **medium** | **low**} **Example:** Switch(config-nd-raguard)# **router-preference maximum high**	Enables filtering of Router Advertisement messages by the Router Preference flag. If not configured, this filter is disabled. • **high**—Accepts RA messages with the Router Preference set to high, medium, or low. • **medium**—Blocks RA messages with the Router Preference set to high. • **low**—Blocks RA messages with the Router Preference set to medium and high.
Step 9	**[no]trusted-port** **Example:** Switch(config-nd-raguard)# **trusted-port**	When configured as a trusted port, all attached devices are trusted, and no further message verification is performed.
Step 10	**default** {**device-role** | **hop-limit** {**maximum** | **minimum**} | **managed-config-flag** | **match** {**ipv6 access-list** | **ra prefix-list** } | **other-config-flag** | **router-preference maximum** | **trusted-port**} **Example:** Switch(config-nd-raguard)# **default hop-limit**	Restores a command to its default value.
Step 11	**do show ipv6 nd raguard policy** *policy_name* **Example:** Switch(config-nd-raguard)# **do show ipv6 nd raguard policy example_policy**	(Optional)—Displays the ND Guard Policy configuration without exiting the RA Guard policy configuration mode.

How to Attach an IPv6 Router Advertisement Guard Policy to an Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Router Advertisement policy to an interface or to VLANs on the interface :

SUMMARY STEPS

    1. **configure terminal**

2. **interface** Interface_type *stack/module/port*
3. **ipv6 nd raguard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_id*s | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
4. **do show running-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters the global configuration mode.
Step 2	**interface** Interface_type *stack/module/port* **Example:** Switch(config)# **interface gigabitethernet 1/1/4**	Specifies an interface type and identifier; enters the interface configuration mode.
Step 3	**ipv6 nd raguard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids*	**add** *vlan_ids*	**except** *vlan_ids*	**none**	**remove** *vlan_ids*	**all**}]	**vlan** [{*vlan_id*s	**add** *vlan_ids*	**except***vlan_ids*	**none**	**remove** *vlan_ids*	**all**}] **Example:** Switch(config-if)# **ipv6 nd raguard attach-policy example_policy** or Switch(config-if)# **ipv6 nd raguard attach-policy example_policy vlan 222,223,224** or Switch(config-if)# **ipv6 nd raguard vlan 222, 223,224**	Attaches the Neighbor Discovery Inspection policy to the interface or the specified VLANs on that interface. The default policy is attached if the **attach-policy** option is not used.
Step 4	**do show running-config** **Example:** Switch#(config-if)# **do show running-config**	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 Router Advertisement Guard Policy to a Layer 2 EtherChannel Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Router Advertisement Guard Policy on an EtherChannel interface or VLAN:

SUMMARY STEPS

1. **configure terminal**
2. **interface range** *Interface_name*
3. **ipv6 nd raguard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_id*s | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
4. **do show running-config interface***portchannel_interface_name*

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters the global configuration mode.
Step 2	**interface range** *Interface_name* **Example:** Switch(config)# **interface Po11**	Specify the port-channel interface name assigned when the EtherChannel was created. Enters the interface range configuration mode. **Tip** Enter the **do show interfaces summary** command for quick reference to interface names and types.
Step 3	**ipv6 nd raguard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids*	**add** *vlan_ids*	**except** *vlan_ids*	**none**	**remove** *vlan_ids*	**all**}]	**vlan** [{*vlan_id*s	**add** *vlan_ids*	**except***vlan_ids*	**none**	**remove** *vlan_ids*	**all**}] **Example:** Switch(config-if-range)# **ipv6 nd raguard attach-policy example_policy** or Switch(config-if-range)# **ipv6 nd raguard attach-policy example_policy vlan 222,223,224** **or** Switch(config-if-range)#**ipv6 nd raguard vlan 222, 223,224**	Attaches the RA Guard policy to the interface or the specified VLANs on that interface. The default policy is attached if the **attach-policy** option is not used.
Step 4	**do show running-config interface***portchannel_interface_name* **Example:** Switch#(config-if-range)# **do show running-config int po11**	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 Router Advertisement Guard Policy to VLANs Globally

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Router Advertisement policy to VLANs regardless of interface:

SUMMARY STEPS

1. **configure terminal**
2. **vlan configuration** *vlan_list*
3. **ipv6 dhcp guard** [**attach-policy** *policy_name*]
4. **do show running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**vlan configuration** *vlan_list* **Example:** Switch(config)# **vlan configuration 335**	Specifies the VLANs to which the IPv6 RA Guard policy will be attached ; enters the VLAN interface configuration mode.
Step 3	**ipv6 dhcp guard** [**attach-policy** *policy_name*] **Example:** Switch(config-vlan-config)#**ipv6 nd raguard attach-policy example_policy**	Attaches the IPv6 RA Guard policy to the specified VLANs across all switch and stack interfaces. The default policy is attached if the **attach-policy** option is not used.
Step 4	**do show running-config** **Example:** Switch#(config-if)# **do show running-config**	Confirms that the policy is attached to the specified VLANs without exiting the configuration mode.

How to Configure an IPv6 DHCP Guard Policy

Beginning in privileged EXEC mode, follow these steps to configure an IPv6 DHCP (DHCPv6) Guard policy:

SUMMARY STEPS

1. **configure terminal**
2. [**no**]**ipv6 dhcp guard policy** *policy-name*
3. [**no**]**device-role** {**client** | **server**}
4. [**no**] **match server access-list** *ipv6-access-list-name*
5. [**no**] **match reply prefix-list** *ipv6-prefix-list-name*
6. [**no**]**preference**{ **max** *limit* | **min** *limit* }
7. [**no**] **trusted-port**

8. **default** {**device-role** | **trusted-port**}

9. **do show ipv6 dhcp guard policy** *policy_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters the global configuration mode.
Step 2	[**no**]**ipv6 dhcp guard policy** *policy-name* **Example:** Switch(config)# **ipv6 dhcp guard policy example_policy**	Specifies the DHCPv6 Guard policy name and enters DHCPv6 Guard Policy configuration mode.
Step 3	[**no**]**device-role** {**client**	**server**} **Example:** Switch(config-dhcp-guard)# **device-role server**	(Optional) Filters out DHCPv6 replies and DHCPv6 advertisements on the port that are not from a device of the specified role. Default is **client**. • **client**—Default value, specifies that the attached device is a client. Server messages are dropped on this port. • **server**—Specifies that the attached device is a DHCPv6 server. Server messages are allowed on this port.
Step 4	[**no**] **match server access-list** *ipv6-access-list-name* **Example:** ;;Assume a preconfigured IPv6 Access List as follows: Switch(config)# **ipv6 access-list my_acls** Switch(config-ipv6-acl)# **permit host FE80::A8BB:CCFF:FE01:F700 any** ;;configure DCHPv6 Guard to match approved access list. Switch(config-dhcp-guard)# **match server access-list my_acls**	(Optional). Enables verification that the advertised DHCPv6 server or relay address is from an authorized server access list (The destination address in the access list is 'any'). If not configured, this check will be bypassed. An empty access list is treated as a permit all.
Step 5	[**no**] **match reply prefix-list** *ipv6-prefix-list-name* **Example:** ;;Assume a preconfigured IPv6 prefix list as follows: Switch(config)# **ipv6 prefix-list my_prefix permit 2001:0DB8::/64 le 128** ;; Configure DCHPv6 Guard to match prefix Switch(config-dhcp-guard)# **match reply prefix-list my_prefix**	(Optional) Enables verification of the advertised prefixes in DHCPv6 reply messages from the configured authorized prefix list. If not configured, this check will be bypassed. An empty prefix list is treated as a permit.

	Command or Action	**Purpose**
Step 6	[no]preference{ max *limit* | min *limit* } **Example:** Switch(config-dhcp-guard)# **preference max 250** Switch(config-dhcp-guard)#**preference min 150**	Configure **max** and **min** when **device-role** is **server**to filter DCHPv6 server advertisements by the server preference value. The defaults permit all advertisements. **max** *limit*—(0 to 255) (Optional) Enables verification that the advertised preference (in preference option) is less than the specified limit. Default is 255. If not specified, this check will be bypassed. **min** *limit*—(0 to 255) (Optional) Enables verification that the advertised preference (in preference option) is greater than the specified limit. Default is 0. If not specified, this check will be bypassed.
Step 7	[no] trusted-port **Example:** Switch(config-dhcp-guard)# **trusted-port**	(Optional) **trusted-port**—Sets the port to a trusted mode. No further policing takes place on the port. **Note** If you configure a trusted port then the device-role option is not available.
Step 8	default {device-role | trusted-port} **Example:** Switch(config-dhcp-guard)# **default device-role**	(Optional) **default**—Sets a command to its defaults.
Step 9	do show ipv6 dhcp guard policy *policy_name* **Example:** Switch(config-dhcp-guard)# **do show ipv6 dhcp guard policy example_policy**	(Optional) Displays the configuration of the IPv6 DHCP guard policy without leaving the configuration submode. Omitting the *policy_name* variable displays all DHCPv6 policies.

Example of DHCPv6 Guard Configuration

```
enable
configure terminal
ipv6 access-list acl1
 permit host FE80::A8BB:CCFF:FE01:F700 any
ipv6 prefix-list abc permit 2001:0DB8::/64 le 128
ipv6 dhcp guard policy pol1
 device-role server
 match server access-list acl1
 match reply prefix-list abc
 preference min 0
 preference max 255
 trusted-port
interface GigabitEthernet 0/2/0
 switchport
 ipv6 dhcp guard attach-policy pol1 vlan add 1
 vlan 1
  ipv6 dhcp guard attach-policy pol1
show ipv6 dhcp guard policy pol1
```

How to Attach an IPv6 DHCP Guard Policy to an Interface or a VLAN on an Interface

Beginning in privileged EXEC mode, follow these steps to configure IPv6 Binding Table Content :

SUMMARY STEPS

1. **configure terminal**
2. **interface** Interface_type *stack/module/port*
3. **ipv6 dhcp guard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_id*s | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
4. **do show running-config interface** Interface_type *stack/module/port*

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters the global configuration mode.
Step 2	**interface** Interface_type *stack/module/port* **Example:** Switch(config)# **interface gigabitethernet 1/1/4**	Specifies an interface type and identifier; enters the interface configuration mode.
Step 3	**ipv6 dhcp guard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids*	**add** *vlan_ids*	**except** *vlan_ids*	**none**	**remove** *vlan_ids*	**all**}]	**vlan** [{*vlan_id*s	**add** *vlan_ids*	**except***vlan_ids*	**none**	**remove** *vlan_ids*	**all**}] **Example:** Switch(config-if)# **ipv6 dhcp guard attach-policy example_policy** or Switch(config-if)# **ipv6 dhcp guard attach-policy example_policy vlan 222,223,224** **or** Switch(config-if)# **ipv6 dhcp guard vlan 222, 223,224**	Attaches the DHCP Guard policy to the interface or the specified VLANs on that interface. The default policy is attached if the **attach-policy** option is not used.
Step 4	**do show running-config interface** Interface_type *stack/module/port* **Example:**	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

Command or Action	Purpose
Switch#(config-if)# **do show running-config gig 1/1/4**	

How to Attach an IPv6 DHCP Guard Policy to a Layer 2 EtherChannel Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 DHCP Guard policy on an EtherChannel interface or VLAN:

SUMMARY STEPS

1. **configure terminal**
2. **interface range** *Interface_name*
3. **ipv6 dhcp guard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_id*s | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
4. **do show running-config interface***portchannel_interface_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters the global configuration mode.
Step 2	**interface range** *Interface_name* **Example:** Switch(config)# **interface Po11**	Specify the port-channel interface name assigned when the EtherChannel was created. Enters the interface range configuration mode. **Tip** Enter the **do show interfaces summary** command for quick reference to interface names and types.
Step 3	**ipv6 dhcp guard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids*	**add** *vlan_ids*	**except** *vlan_ids*	**none**	**remove** *vlan_ids*	**all**}]	**vlan** [{*vlan_id*s	**add** *vlan_ids*	**except***vlan_ids*	**none**	**remove** *vlan_ids*	**all**}] **Example:** Switch(config-if-range)# **ipv6 dhcp guard attach-policy example_policy** or Switch(config-if-range)# **ipv6 dhcp guard attach-policy example_policy vlan 222,223,224** or Switch(config-if-range)#**ipv6 dhcp guard vlan 222, 223,224**	Attaches the DHCP Guard policy to the interface or the specified VLANs on that interface. The default policy is attached if the **attach-policy** option is not used.

	Command or Action	Purpose
Step 4	**do show running-config interface***portchannel_interface_name* **Example:** Switch#(config-if-range)# **do show running-config int po11**	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 DHCP Guard Policy to VLANs Globally

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 DHCP Guard policy to VLANs across multiple interfaces:

SUMMARY STEPS

1. **configure terminal**
2. **vlan configuration** *vlan_list*
3. **ipv6 dhcp guard** [**attach-policy** *policy_name*]
4. **do show running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters the global configuration mode.
Step 2	**vlan configuration** *vlan_list* **Example:** Switch(config)# **vlan configuration 334**	Specifies the VLANs to which the IPv6 Snooping policy will be attached ; enters the VLAN interface configuration mode.
Step 3	**ipv6 dhcp guard** [**attach-policy** *policy_name*] **Example:** Switch(config-vlan-config)#**ipv6 dhcp guard attach-policy example_policy**	Attaches the IPv6 Neighbor Discovery policy to the specified VLANs across all switch and stack interfaces. The default policy is attached if the **attach-policy** option is not used. The default policy is, device-role **client**, **no** trusted-port.
Step 4	**do show running-config** **Example:** Switch#(config-if)# **do show running-config**	Confirms that the policy is attached to the specified VLANs without exiting the configuration mode.

Additional References

Related Documents

Related Topic	Document Title
Implementing IPv6 Addressing and Basic Connectivity	[http://www.cisco.com/Slash.info.config.v6dEx.html](#)
IPv6 network management and security topics	IPv6 Configuration Library, Cisco IOS XE Release 3SE (Catalyst 3850 Switches) [http://www.cisco.com/Slash.info.config.y.3SPe6.2SSe.html](#)
IPv6 Command Reference	IPv6 Command Reference, Cisco IOS XE Release 3SE (Catalyst 3850 Switches) [http://www.cisco.com/Slash.info.command.v6.3SSe.html](#)

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Configuring Cisco TrustSec

Information about Cisco TrustSec

Cisco TrustSec provides security improvements to Cisco network devices based on the capability to strongly identify users, hosts, and network devices within a network. TrustSec provides topology-independent and scalable access controls by uniquely classifying data traffic for a particular role. TrustSec ensures data confidentiality and integrity by establishing trust among authenticated peers and encrypting links with those peers.

The key component of Cisco TrustSec is the Cisco Identity Services Engine (ISE). Cisco ISE can provision switches with TrustSec Identities and Security Group ACLs (SGACLs), though these may be configured manually on the switch.

Cisco TrustSec Features

The table below lists the Cisco TrustSec features implemented on Cisco TrustSec-enabled Catalyst 2960-X and 2960-XR Series Switches:

Cisco TrustSec Feature	Description
Endpoint Admission Control (EAC)	EAC is an authentication process for an endpoint user or a device connecting to the TrustSec domain. Usually EAC takes place at the access level switch. Successful authentication and authorization in the EAC process results in Security Group Tag assignment for the user or device. Currently EAC can be 802.1X, MAC Authentication Bypass (MAB), and Web Authentication Proxy (WebAuth).

Cisco TrustSec Feature	Description
SGT Exchange Protocol (SXP)	Security Group Tag Exchange Protocol (SXP). With SXP, devices that are not TrustSec-hardware-capable can receive SGT attributes for authenticated users and devices from the Cisco Identity Services Engine (ISE) or the Cisco Secure Access Control System (ACS). The devices can then forward a sourceIP-to-SGT binding to a TrustSec-hardware-capable device will tag the source traffic for SGACL enforcement.

Feature Information for Cisco TrustSec

Table 106: Feature Information for Cisco TrustSec

Feature Name	Release	Feature Information
SXPv1 and SXPv2	Cisco IOS XE 15.0(2)EX	SXP is introduced on the Catalyst 2960-X switch.
SXPv1 and SXPv2	Cisco IOS XE 15.0(2)EX1	SXP is introduced on the Catalyst 2960-XR switch.

PART IX

Configuring Cisco IOS IP SLAs

Configuring Cisco IP SLAs

Restrictions on SLAs

This section lists the restrictions on SLAs.

The following are restrictions on IP SLAs network performance measurement:

• The switch does not support VoIP service levels using the gatekeeper registration delay operations measurements.

• Only a Cisco IOS device can be a source for a destination IP SLAs responder.

• 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.

Information About SLAs

Cisco IOS IP Service Level Agreements (SLAs)

Cisco IOS IP SLAs send data across the network to measure performance between multiple network locations or across multiple network paths. They simulate network data and IP services and collect network performance information in real time. Cisco IOS IP SLAs generate and analyze 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 SLA operations can be used for troubleshooting, for problem analysis, and for designing network topologies.

Depending on the specific Cisco IOS IP SLA operations, 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 SLA 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 are 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 collect and analyze the following 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 Cisco Prime Internetwork Performance Monitor (IPM) and other third-party Cisco partner performance management products.

Using IP SLAs can provide the following benefits:

- Service-level agreement monitoring, measurement, and verification.

- Network performance monitoring

    - Measurement of jitter, latency, or packet loss in the network.

    - 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).

- Network operation troubleshooting 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).

Network Performance Measurement with Cisco IOS IP SLAs

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 98: Cisco IOS IP SLAs Operation

The following figure shows how IP SLAs begin 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.
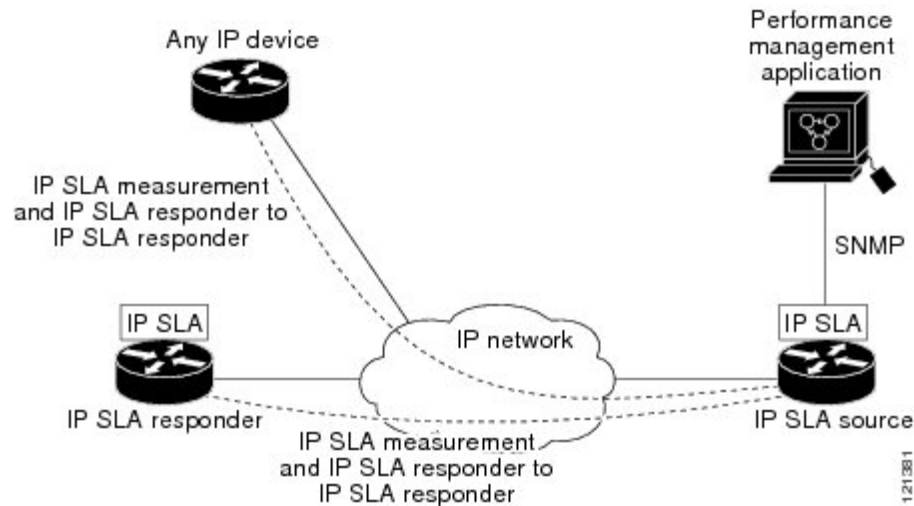
IP SLA Responder and IP SLA Control Protocol

The IP SLA responder is a component embedded in the destination Cisco device that allows the system to anticipate and respond to IP SLA request packets. The responder provides accurate measurements without the need for dedicated probes. The responder uses the Cisco IOS IP SLA Control Protocol to provide a mechanism through which it can be notified on which port it should listen and respond.

Note   The IP SLA responder can be a Cisco IOS Layer 2, responder-configurable switch. The responder does not need to support full IP SLA functionality.

The following figure shows where the Cisco IOS IP SLA responder fits in the IP network. The responder listens on a specific port for control protocol messages sent by an IP SLA 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 SLA packet, or when the specified time expires. MD5 authentication for control messages is available for added security.

Figure 99: Cisco IOS IP SLAs Operation
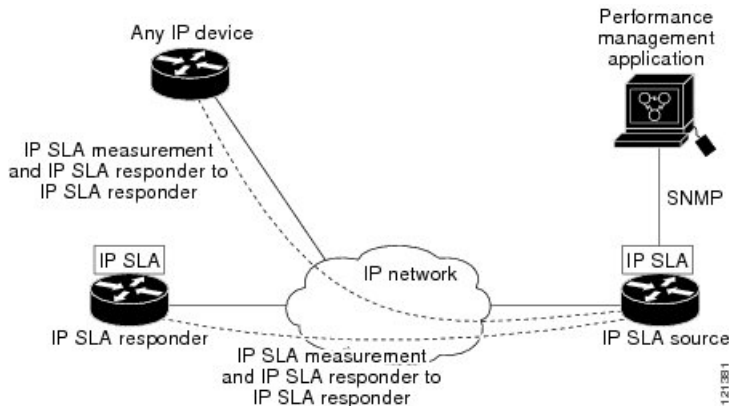
You do not need to enable the responder on the destination device for all IP SLA operations. For example, a responder is not required for services that are already provided by the destination router (such as Telnet or HTTP).

Response Time Computation for IP SLAs

Switches, controllers, 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 minimize 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 SLA test packets use time stamping to minimize the processing delays.

When the IP SLA 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 100: Cisco IOS IP SLA Responder Time Stamping

The following figure 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
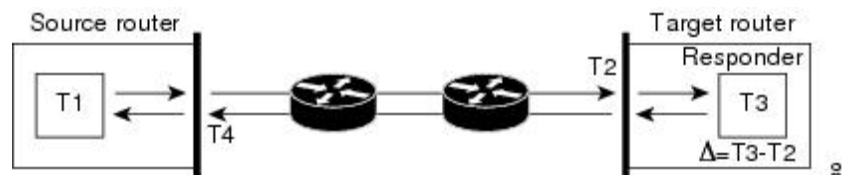
level to allow for greater accuracy.

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.

How to Configure IP SLAs Operations

This section does not include configuration information for all available operations as the configuration information details are included in the *Cisco IOS IP SLAs Configuration Guide*. It does include several operations as examples, including configuring the responder, configuring a UDP jitter operation, which requires a responder, and configuring an ICMP echo operation, which does not require a responder. For details about configuring other operations, see the *Cisco IOS IP SLAs Configuration Guide*.

Default Configuration

No IP SLAs operations are configured.

Configuration Guidelines

For information on the IP SLA commands, see the *Cisco IOS IP SLAs Command Reference, Release 12.4T* command reference.

For detailed descriptions and configuration procedures, see the *Cisco IOS IP SLAs Configuration Guide, Release 12.4TL*.

Configuring the IP SLA Responder

The IP SLA responder is available only on Cisco IOS software-based devices, including some Layer 2 switches that do not support full IP SLA functionality.

Follow these steps to configure the IP SLA responder on the target device (the operational target):

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip sla responder** {**tcp-connect** | **udp-echo**} **ipaddress** *ip-address* **port** *port-number*
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:**	Enters global configuration mode.

	Command or Action	Purpose
	Switch# **configure terminal**	
Step 3	**ip sla responder** {**tcp-connect**	**udp-echo**} **ipaddress** *ip-address* **port** *port-number* **Example:** Switch(config)# **ip sla responder udp-echo 172.29.139.134 5000**	Configures the switch as an IP SLA responder. The 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*—Enter the destination IP address. **port** *port-number*—Enter the destination port number. **Note** The IP address and port number must match those configured on the source device for the IP SLA operation.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Monitoring IP SLA Operations

The following table describes the commands used to display IP SLA operation configurations and results:

Table 107: Monitoring IP SLA Operations

show ip sla authentication	Displays IP SLA authentication information.
show ip sla responder	Displays information about the IP SLA responder.

Additional References

Related Documents

Related Topic	Document Title
Cisco Medianet Metadata Guide	http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/mdata/configuration/15-sy/mdata-15sy-book/metadata-framework.pdf
Cisco Media Services Proxy Configuration Guide	http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/msp/configuration/15-mt/msp-15-mt-book.pdf
Cisco Mediatrace and Cisco Performance Monitor Configuration Guide	http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/media_monitoring/configuration/15-mt/mm-15-mt-book/mm-mediatrace.html

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	-

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Service Level Agreements

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Stack Manager and High Availability

Managing Switch Stacks

Prerequisites for Switch Stacks

All stack members must run the same Cisco IOS software image to ensure compatibility among stack members.

For switch stack hardware considerations, see the *Catalyst 2960-XR Switch Hardware Installation Guide*.

Restrictions for Switch Stacks

The following are restrictions for your switch stack configuration:

- The switch does not support mixed stacking. Stacking is supported only with other Catalyst 2960-XR switches.

- Auto-upgrade of stack can not be done when one of the switches in stack is with version Cisco IOS 15.2(3)E. This means that whenever any of the switches in the stack goes into a version mismatch, and if either the active stack is running Cisco IOS 15.2(3)E, or if a member is running Cisco 15.2(3)E, the member can not be auto-upgraded to the required version.

Information About Switch Stacks

Switch Stack Overview

A switch stack is a set of up to eight stacking-capable switches connected through their stack ports. You can connect only one switch type in a stack, or you can connect a mix of Catalyst 2960-XR, Catalyst 2960-X and Catalyst 2960-S switches in the stack. The stack can have one of these configurations:

- Homogeneous stack—A Catalyst 2960-XR only stack with only Catalyst 2960-XR switches as stack members. A homogenous stack can have up to 8 stack members.

- Mixed stack—A stack with a mix of Catalyst 2960-XR, Catalyst 2960-X and Catalyst 2960-S switches as stack members. A mixed stack of Catalyst 2960-XR and Catalyst 2960-X switches can have up to 8 stack members. A mixed stack that includes a Catalyst 2960-S switch can have up to 4 stack members.

The active stack controls the operation of the switch stack, and is the single point of stack-wide management. From the active stack, you configure:

- System-level (global) features that apply to all stack members

- Interface-level features for each stack member

The active stack contains the saved and running configuration files for the switch stack. The configuration files include the system-level settings for the switch stack and the interface-level settings for each stack member. Each stack member has a current copy of these files for back-up purposes.

Supported Features in a Switch Stack

The system-level features supported on the active switch are supported on the entire switch stack.

Encryption Features

If the active switch is running the cryptographic universal software image (supports encryption), the encryption features are available on the switch stack.

FlexStack-Plus

The stack members use the Cisco FlexStack-Plus technology to work together as a unified system. Layer 2 protocols support the entire switch stack as a single entity in the network.

> **Note** Switch stacks running the LAN Base image do not support Layer 3 features.

The FlexStack-Plus bandwidth for a single stack port is 20 Gbps. With FlexStack-Plus technology, up to eight members can be joined into a single stack. In a mixed stack of Catalyst 2960-X and Catalyst 2960-S switches, FlexStack-Plus reverts to FlexStack capabilities of 10 Gbps stack port bandwidth and a maximum of four members per stack.

Switch Stack Membership

A switch stack has up to eight stack members connected through their stack ports. A switch stack always has one active switch.

A standalone device is a device stack with one stack member that also operates as the active switch. You can connect one standalone device to another to create a stack containing two stack members, with one of them as the active switch. You can connect standalone devices to an existing device stack to increase the stack membership.

Figure 101: Creating a Switch Stack from Two Standalone Switches
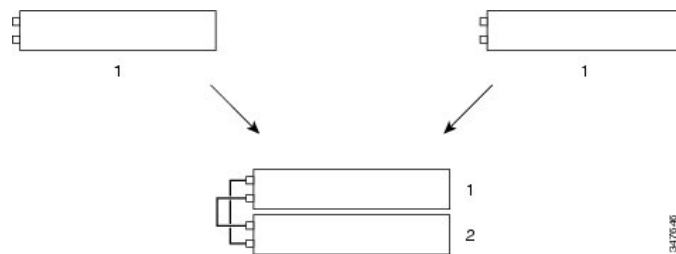
Figure 102: Adding a Standalone Switch to a Switch Stack
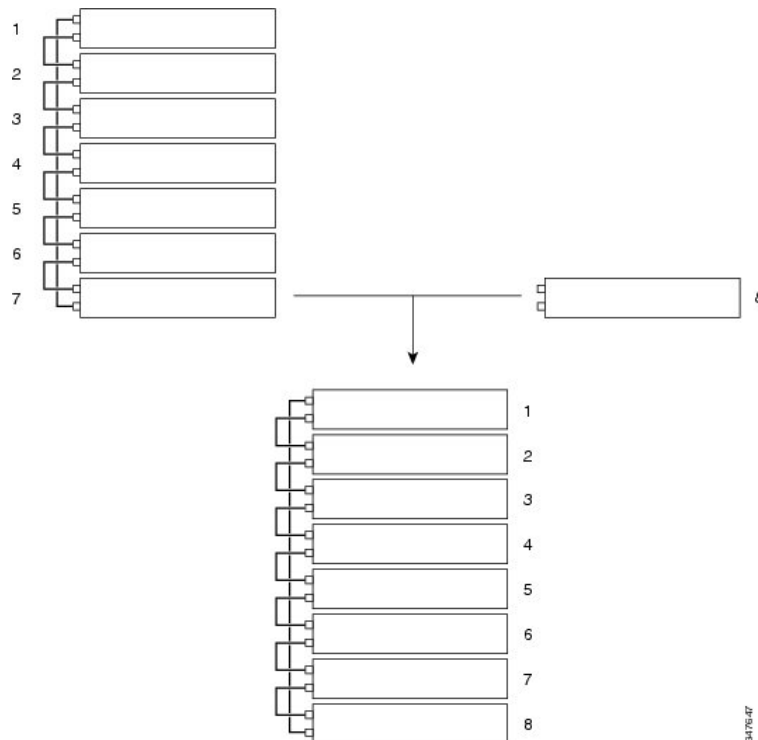
Hello messages are sent and received by all stack members.

- If a stack member does not respond, that member is removed from the stack.

- If the standby device does not respond, a new standby device is elected.

- If the active device does not respond, the standby device becomes the active device.

In addition, keepalive messages are sent and received between the active and standby devices.

- If the standby device does not respond, a new standby device is elected.

- If the active device does not respond, the standby device becomes the active device.

Changes to Switch Stack Membership

If you replace a stack member with an identical model, the new switch functions with exactly the same configuration as the replaced switch, assuming that the new switch (referred to as the provisioned switch) is using the same member number as the replaced switch.

The operation of the switch stack continues uninterrupted during membership changes unless you remove the active switch or you add powered-on standalone switches or switch stacks.

- Adding powered-on switches (merging) causes all switches to reload and elect a new active switch from among themselves. The newly elected active switch retains its role and configuration. All other switches retain their stack member numbers and use the stack configuration of the newly elected active switch.

> **Note** In Cisco IOS XE 3.6.4E and later versions, when a new switch is powered-on as a standalone switch before it is added as part of the switch stack, only this switch is reloaded and not the whole switch stack.

Adding powered-on switches (merging) causes the active stack of the merging switch stacks to elect an active stack from among themselves. The reelected active stack retains its role and configuration as do its stack members. All remaining switches, including the former active stacks, reload and join the switch stack as stack members. They change their stack member numbers to the lowest available numbers and use the stack configuration of the reelected active stack.

- Removing powered-on stack members causes the switch stack to divide (partition) into two or more switch stacks, each with the same configuration. This can cause:

  - An IP address conflict in your network. If you want the switch stacks to remain separate, change the IP address or addresses of the newly created switch stacks.

  - A MAC address conflict between two members in the stack. You can use the **stack-mac update force** command to resolve the conflict.

If a newly created switch stack does not have an active switch or standby switch, the switch stack will reload and elect a new active switch.

> **Note** Make sure that you power off the switches that you add to or remove from the switch stack.
>
> After adding or removing stack members, make sure that the switch stack is operating at full bandwidth (40 Gbps). Press the Mode button on a stack member until the Stack mode LED is on. The last two right port LEDs on all switches in the stack should be green. Depending on the switch model, the last two right ports are 10-Gigabit Ethernet ports or small form-factor pluggable (SFP) module ports (10/100/1000 ports). If one or both of these LEDs are not green on any of the switches, the stack is not operating at full bandwidth.
>
> It may take upto 4 seconds for stack convergence when a new stack member is added to the existing switch stack.

If you remove powered-on members but do not want to partition the stack:

- Power off the switches in the newly created switch stacks.

- Reconnect them to the original switch stack through their stack ports.

- Power on the switches.

For cabling and power considerations that affect switch stacks, see the *Catalyst 2960-XR Switch Hardware Installation Guide*.

Stack Member Numbers

The stack member number (1 to 8) identifies each member in the switch stack. The member number also determines the interface-level configuration that a stack member uses. You can display the stack member number by using the **show switch** EXEC command.

A new, out-of-the-box device (one that has not joined a device stack or has not been manually assigned a stack member number) ships with a default stack member number of 1. When it joins a device stack, its default stack member number changes to the lowest available member number in the stack.

Stack members in the same stack cannot have the same stack member number. Every stack member, including a standalone device, retains its member number until you manually change the number or unless the number is already being used by another member in the stack.

- If you manually change the stack member number by using the **switch** *current-stack-member-number* **renumber** *new-stack-member-number* command, the new number goes into effect after that stack member resets (or after you use the **reload slot** *stack-member-number* privileged EXEC command) and only if that number is not already assigned to any other members in the stack. Another way to change the stack member number is by changing the device_NUMBER environment variable.

  If the number is being used by another member in the stack, the device selects the lowest available number in the stack.

  If you manually change the number of a stack member and no interface-level configuration is associated with that new member number, that stack member resets to its default configuration.

  You cannot use the **switch** *current-stack-member-number* **renumber** *new-stack-member-number* command on a provisioned device. If you do, the command is rejected.

- If you move a stack member to a different device stack, the stack member retains its number only if the number is not being used by another member in the stack. If it is being used, the device selects the lowest available number in the stack.

- If you merge device stacks, the device that join the device stack of a new active device select the lowest available numbers in the stack.

As described in the hardware installation guide, you can use the device port LEDs in Stack mode to visually determine the stack member number of each stack member.

In the **default** mode Stack LED will blink in green color only on the active switch. However, when we scroll the Mode button to **Stack** option - Stack LED will glow green on all the stack members.

When mode button is scrolled to **Stack** option, the switch number of each stack member will be displayed as LEDs on the first five ports of that switch. The switch number is displayed in binary format for all stack members. On the switch, the amber LED indicates value 0 and green LED indicates value 1.

Example for switch number 5 (Binary - 00101):

First five LEDs glow as follows on stack member with switch number 5.

- Port-1 : Amber

- Port-2 : Amber

- Port-3 : Green

- Port-4 : Amber

- Port-5 : Green

Similarly, the first five LEDs glow amber or green, depending on the switch number on all stack members.

Note
- If you connect a Horizontal stack port to a normal network port on other end, stack port transmission/reception will be disabled within 30 seconds if no SDP packets are received from the other end.

- Stack port will not go down but only transmission/reception will be disabled. The log message shown below will be displayed on the console. Once the peer end network port is converted to stack port, transmission/reception on this stack port will be enabled.

```
%STACKMGR-4-HSTACK_LINK_CONFIG: Verify peer stack port setting for hstack
 StackPort-1 switch 5 (hostname-switchnumber)
```

Stack Member Priority Values

A higher priority value for a stack member increases the probability of it being elected active switch and retaining its stack member number. The priority value can be 1 to 15. The default priority value is 1. You can display the stack member priority value by using the **show switch** EXEC command.

Note
We recommend assigning the highest priority value to the device that you prefer to be the active device. This ensures that the device is reelected as the active device if a reelection occurs.

To change the priority value for a stack member, use the **switch** *stack-member-number* **priority** *new priority-value* command. For more information, see the "Setting the Stack Member Priority Value" section.

The new priority value takes effect immediately but does not affect the current active device. The new priority value helps determine which stack member is elected as the new active device when the current active device or the device stack resets.

Switch Stack Bridge ID and MAC Address

The MAC address of the active switch determines the stack MAC address.

When the stack initializes, the MAC address of the active switch determines the bridge ID that identifies the stack in the network.

If the active switch changes, the MAC address of the new active switch determines the new bridge ID and stack MAC address.

If the entire switch stack reloads, the switch stack uses the MAC address of the active switch.

Persistent MAC Address on the Switch Stack

You can use the persistent MAC address feature to set a time delay before the stack MAC address changes to the MAC address of the new active stack. When this feature is enabled, the stack MAC address changes in approximately 4 minutes. During this time, if the previous active stack rejoins the stack, the stack continues to use its MAC address as the stack MAC address, even if the switch is now a stack member and not the active stack. If the previous active stack does not rejoin the stack during this period, the switch stack takes the MAC address of the new active stack as the stack MAC address.

You can also configure stack MAC persistency so that the stack MAC address never changes to the new active switch MAC address.

Active and Standby Switch Election and Reelection

All stack members are eligible to be the active stack. If the active stack becomes unavailable, the remaining members elect a new active stack from among themselves.

The active switch is elected or reelected based on one of these factors and in the order listed:

1. The switch that is currently the active switch.

2. The switch with the highest stack member priority value.

> **Note** We recommend assigning the highest priority value to the switch that you prefer to be the active switch ensures that the switch is reelected as active switch if a reelection occurs.

3. The switch that has the configuration file.

4. The switch with the lowest MAC address.

An active stack retains its role unless one of these events occurs:

- The switch stack is reset.*

- The active stack is removed from the switch stack.

- The active stack is reset or powered off.

- The active stack fails.

- The switch stack membership is increased by adding powered-on standalone switches or switch stacks.*

In the events marked by an asterisk (*), the current active stack *might* be reelected based on the listed factors.

When you power on or reset an entire switch stack, some stack members *might not* participate in the active stack election. Stack members that are powered on within the same 20-second time frame participate in the active stack election and have a chance to become the active stack. Stack members that are powered on after the 20-second time frame do not participate in this initial election and become stack members. All stack

members participate in reelections. For all powering considerations that affect active-stack elections, see the "Switch Installation" chapter in the hardware installation guide.

The new active stack becomes available after a few seconds. In the meantime, the switch stack uses the forwarding tables in memory to minimize network disruption. The physical interfaces on the other available stack members are not affected during a new active stack election and reset.

After a new active stack is elected and the previous active stack becomes available, the previous active stack *does not* resume its role as the active stack.

For all powering considerations that affect active-stack elections, see the *Catalyst 2960-XR Switch Hardware Installation Guide*.

Switch Stack Configuration Files

The active switch has the saved and running configuration files for the switch stack. All stack members periodically receive synchronized copies of the configuration files from the active switch. If the active switch becomes unavailable, any stack member assuming the role of active switch has the latest configuration files.

The configuration files record these settings:

- System-level (global) configuration settings such as IP, STP, VLAN, and SNMP settings that apply to all stack members
- Stack member interface-specific configuration settings that are specific for each stack member

Note   The interface-specific settings of the active switch are saved if the active switch is replaced without saving the running configuration to the startup configuration.

A new, out-of-box device joining a switch stack uses the system-level settings of that switch stack. If a device is moved to a different switch stack before it is powered on, that device loses its saved configuration file and uses the system-level configuration of the new switch stack. If the device is powered on as a standalone device before it joins the new switch stack, the stack will reload. When the stack reloads, the new device may become the device, retain its configuration and overwrite the configuration files of the other stack members.

The interface-specific configuration of each stack member is associated with the stack member number. Stack members retain their numbers unless they are manually changed or they are already used by another member in the same switch stack. If the stack member number changes, the new number goes into effect after that stack member resets.

- If an interface-specific configuration does not exist for that member number, the stack member uses its default interface-specific configuration.
- If an interface-specific configuration exists for that member number, the stack member uses the interface-specific configuration associated with that member number.

If you replace a failed member with an identical model, the replacement member automatically uses the same interface-specific configuration as the failed device. You do not need to reconfigure the interface settings. The replacement device (referred to as the provisioned device) must have the same stack member number as the failed device.

You back up and restore the stack configuration in the same way as you would for a standalone device configuration.

Offline Configuration to Provision a Stack Member

You can use the offline configuration feature to *provision* (to supply a configuration to) a new switch before it joins the switch stack. You can configure the stack member number, the switch type, and the interfaces associated with a switch that is not currently part of the stack. The configuration that you create on the switch stack is called the *provisioned configuration*. The switch that is added to the switch stack and that receives this configuration is called the *provisioned switch*.

You manually create the provisioned configuration through the **switch** *stack-member-number* **provision** *type* global configuration command. You must change the *stack-member-number* on the provisioned switch before you add it to the stack, and it must match the stack member number that you created for the new switch on the switch stack. The switch type in the provisioned configuration must match the switch type of the newly added switch. The provisioned configuration is automatically created when a switch is added to a switch stack and when no provisioned configuration exists.

When you configure the interfaces associated with a provisioned switch, the switch stack accepts the configuration, and the information appears in the running configuration. However, as the switch is not active, any configuration on the interface is not operational and the interface associated with the provisioned switch does not appear in the display of the specific feature. For example, VLAN configuration information associated with a provisioned switch does not appear in the **show vlan** user EXEC command output on the switch stack.

The switch stack retains the provisioned configuration in the running configuration whether or not the provisioned switch is part of the stack. You can save the provisioned configuration to the startup configuration file by entering the **copy running-config startup-config** privileged EXEC command. The startup configuration file ensures that the switch stack can reload and can use the saved information whether or not the provisioned switch is part of the switch stack.

Effects of Adding a Provisioned Switch to a Switch Stack

When you add a provisioned Switch to the switch stack, the stack applies either the provisioned configuration or the default configuration. This table lists the events that occur when the switch stack compares the provisioned configuration with the provisioned switch.

Table 108: Results of Comparing the Provisioned Configuration with the Provisioned Switch

Scenario		Result
The stack member numbers and the Switch types match.	1. If the stack member number of the provisioned switch matches the stack member number in the provisioned configuration on the stack, and 2. If the Switch type of the provisioned switch matches the Switch type in the provisioned configuration on the stack.	The switch stack applies the provisioned configuration to the provisioned switch and adds it to the stack.

Scenario		Result
The stack member numbers match but the Switch types do not match.	1. If the stack member number of the provisioned switch matches the stack member number in the provisioned configuration on the stack, but 2. The Switch type of the provisioned switch does not match the Switch type in the provisioned configuration on the stack.	The switch stack applies the default configuration to the provisioned switch and adds it to the stack. The provisioned configuration is changed to reflect the new information.
The stack member number is not found in the provisioned configuration.		The switch stack applies the default configuration to the provisioned switch and adds it to the stack. The provisioned configuration is changed to reflect the new information.
The stack member number of the provisioned switch is not found in the provisioned configuration.		The switch stack applies the default configuration to the provisioned switch and adds it to the stack.

If you add a provisioned switch that is a different type than specified in the provisioned configuration to a powered-down switch stack and then apply power, the switch stack rejects the (now incorrect) **switch** *stack-member-number* **provision** *type* global configuration command in the startup configuration file. However, during stack initialization, the nondefault interface configuration information in the startup configuration file for the provisioned interfaces (potentially of the wrong type) is executed. Depending on the differences between the actual Switch type and the previously provisioned switch type, some commands are rejected, and some commands are accepted.

Note If the switch stack does not contain a provisioned configuration for a new Switch, the Switch joins the stack with the default interface configuration. The switch stack then adds to its running configuration with a **switch** *stack-member-number* **provision** *type* global configuration command that matches the new Switch. For configuration information, see the *Provisioning a New Member for a Switch Stack* section.

Effects of Replacing a Provisioned Switch in a Switch Stack

When a provisioned switch in a switch stack fails, it is removed from the stack, and is replaced with another Switch, the stack applies either the provisioned configuration or the default configuration to it. The events that occur when the switch stack compares the provisioned configuration with the provisioned switch are the same as those when you add a provisioned switch to a stack.

Effects of Removing a Provisioned Switch from a Switch Stack

If you remove a provisioned switch from the switch stack, the configuration associated with the removed stack member remains in the running configuration as provisioned information. To completely remove the configuration, use the **no switch** *stack-member-number* **provision** global configuration command.

Stack Protocol Version

Each software image includes a *stack protocol version*. The stack protocol version has a *major* version number and a *minor* version number (for example 1.4, where 1 is the major version number and 4 is the minor version number). Both version numbers determine the level of compatibility among the stack members. You can display the stack protocol version by using the **show platform stack manager all** privileged EXEC command.

The switches with the same Cisco IOS software version have the same stack protocol version. Such switches are fully compatible, and all features function properly across the switch stack. A device with the same Cisco IOS software version as the active switch can immediately join the switch stack.

If an incompatibility exists, the fully functional stack members generate a system message that describes the cause of the incompatibility on the specific stack members. The active switch sends the message to all stack members.

For more information, see the *Major Version Number Incompatibility Among Switches* procedure and the *Minor Version Number Incompatibility Among Switches* procedure.

Major Stack Protocol Version Number Incompatibility Among Stack-Capable Switches

Switch with different major Cisco IOS software versions usually have different stack protocol versions. Switch with different major version numbers are incompatible and cannot exist in the same switch stack.

Minor Stack Protocol Version Number Incompatibility Among Stack-Capable Switches

Switches with the same major version number but with a different minor version number are considered partially compatible. When connected to a switch stack, a partially compatible switch enters version-mismatch (VM) mode and cannot join the stack as a fully functioning member. The software detects the mismatched software and tries to upgrade (or downgrade) the switch in VM mode with the switch stack image or with a tar file image from the switch stack flash memory. The software uses the automatic upgrade (auto-upgrade) and the automatic advise (auto-advise) features.

The port LEDs on switches in version-mismatch mode will also remain off. Pressing the Mode button does not change the LED mode.

Auto-Upgrade

The purpose of the auto-upgrade feature is to allow a switch to be upgraded to a compatible software image, so that the switch can join the switch stack.

When a new switch attempts to join a switch stack, each stack member performs compatibility checks with itself and the new switch. Each stack member sends the results of the compatibility checks to the active stack, which uses the results to determine whether the switch can join the switch stack. If the software on the new switch is incompatible with the switch stack, the new switch enters version-mismatch (VM) mode.

If the auto-upgrade feature is enabled on the existing switch stack, the active stack automatically upgrades the new switch with the same software image running on a compatible stack member. Auto-upgrade starts a few minutes after the mismatched software is detected before starting.

By default, auto-upgrade is enabled (the **boot auto-copy-sw** global configuration command is enabled). You can disable auto-upgrade by using the **no boot auto-copy-sw** global configuration command on the active stack. You can check the status of auto-upgrade by using the **show boot** privileged EXEC command and by checking the *Auto upgrade* line in the display.

Auto-upgrade includes an auto-copy process and an auto-extract process.

- Auto-copy automatically copies the software image running on any stack member to the new switch to automatically upgrade it. Auto-copy occurs if auto-upgrade is enabled, if there is enough flash memory in the new switch, and if the software image running on the switch stack is suitable for the new switch.

> ✎
>
> **Note**   A switch in VM mode might not run all released software. For example, new switch hardware is not recognized in earlier versions of software.

- Automatic extraction (auto-extract) occurs when the auto-upgrade process cannot find the appropriate software in the stack to copy to the new switch. In that case, the auto-extract process searches all switches in the stack for the tar file needed to upgrade the switch stack or the new switch. The tar file can be in any flash file system in the switch stack or in the new switch. If a tar file suitable for the new switch is found on a stack member, the process extracts the file and automatically upgrades the new switch.

The auto-upgrade (auto-copy and auto-extract) processes start a few minutes after the mismatched software is detected.

When the auto-upgrade process is complete, the new switch reloads and joins the stack as a fully functioning member. If you have both stack cables connected during the reload, network downtime does not occur because the switch stack operates on two rings.

Auto-Advise

Automatic advise (auto-advise) occurs when the auto-upgrade process cannot find appropriate stack member software to copy to the new switch. This process tells you the command (**archive copy-sw** or **archive download-sw** privileged EXEC command) and the image name (tar filename) needed to manually upgrade the switch stack or the new switch. The recommended image can be the running switch stack image or a tar file in any flash file system in the switch stack (including the new switch). If an appropriate image is not found in the stack flash file systems, the auto-advise process tells you to install new software on the switch stack. Auto-advise cannot be disabled, and there is no command to check its status.

Examples of Auto-Advise Messages

When you add a switch that has a different minor version number to the switch stack, the software displays messages in sequence (assuming that there are no other system messages generated by the switch).

This example shows that the switch stack detected a new switch that is running a different minor version number than the switch stack. Auto-copy starts, finds suitable software to copy from a stack member to the switch in VM mode, upgrades the switch in VM mode, and then reloads it:

```
*Mar 11 20:31:19.247:%STACKMGR-6-STACK_LINK_CHANGE:Stack Port 2 Switch 2 has changed to
state UP
*Mar 11 20:31:23.232:%STACKMGR-6-SWITCH_ADDED_VM:Switch 1 has been ADDED to the
stack(VERSION_MISMATCH)
*Mar 11 20:31:23.291:%STACKMGR-6-SWITCH_ADDED_VM:Switch 1 has been ADDED to the
stack(VERSION_MISMATCH) (Stack_1-3)
*Mar 11 20:33:23.248:%IMAGEMGR-6-AUTO_COPY_SW_INITIATED:Auto-copy-software process initiated
 for switch number(s) 1
```

```
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Searching for stack member to act
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:as software donor...
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Found donor (system #2) for
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:member(s) 1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:System software to be uploaded:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:System Type: 0x00000000
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:archiving c2960x-universalk9-mz.150-2.EX1
(directory)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:archiving c2960x-universalk9-mz.150-2.EX1.bin
 (4945851 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:archiving
c2960x-universalk9-mz.150-2.EX1/info(450 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:archiving info (104 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:examining image...
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting info (104 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting
c2960x-universalk9-mz.150-2.EX1/info(450 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting info (104 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Stacking Version Number:1.4
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:System Type: 0x00000000
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Ios Image File Size: 0x004BA200
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Total Image File Size:0x00818A00
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Minimum Dram required:0x08000000
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Image Suffix:universalk9-mz.150-2.EX1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Image Directory:c2960x-universalk9-mz.150-2.EX1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Image Name:c2960x-universalk9-mz.150-2.EX1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Image 1:flash1:c2960x-universalk9-mz.150-2.EX1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Old image will be deleted after download.
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Extracting images from archive into flash on
 switch 1...
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:c2960x-universalk9-mz.150-2.EX1 (directory)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting
c2960x-universalk9-mz.150-2.EX1/c2960x-universalk9-mz.150-2.EX1 (4945851 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting c2960x-universalk9-mz.150-2.EX1/info
 (450 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting info (104 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Installing
(renaming):`flash1:update/c2960x-universalk9-mz.150-2.EX1' ->
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: `flash1:c2960x-universalk9-mz.150-2.EX1'
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:New software image installed in
flash1:c2960x-universalk9-mz.150-2.EX1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Removing old
image:flash1:c2960x-universalk9-mz.150-2.EX1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:All software images installed.
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Requested system reload in progress...
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Software successfully copied to
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:system(s) 1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Done copying software
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Reloading system(s) 1
```

This example shows that the switch stack detected a new switch that is running a different minor version number than the switch stack. Auto-copy starts but cannot find software in the switch stack to copy to the VM-mode switch to make it compatible with the switch stack. The auto-advise process starts and recommends that you download a tar file from the network to the switch in VM mode:

```
*Mar 1 00:01:11.319:%STACKMGR-6-STACK_LINK_CHANGE:Stack Port 2 Switch 2 has changed to state
 UP
*Mar 1 00:01:15.547:%STACKMGR-6-SWITCH_ADDED_VM:Switch 1 has been ADDED to the stack
(VERSION_MISMATCH)
stack_2#
*Mar 1 00:03:15.554:%IMAGEMGR-6-AUTO_COPY_SW_INITIATED:Auto-copy-software process initiated
 for switch number(s) 1
*Mar 1 00:03:15.554:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 1 00:03:15.554:%IMAGEMGR-6-AUTO_COPY_SW:Searching for stack member to act
*Mar 1 00:03:15.554:%IMAGEMGR-6-AUTO_COPY_SW:as software donor...
*Mar 1 00:03:15.554:%IMAGEMGR-6-AUTO_COPY_SW:Software was not copied
*Mar 1 00:03:15.562:%IMAGEMGR-6-AUTO_ADVISE_SW_INITIATED:Auto-advise-software process
initiated for switch number(s) 1
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:Systems with incompatible software
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:have been added to the stack. The
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:storage devices on all of the stack
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:members have been scanned, and it has
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:been determined that the stack can be
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:repaired by issuing the following
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:command(s):
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW: archive download-sw /force-reload /overwrite
 /dest 1 flash1:c2960x-universalk9-mz.150-2.EX1.tar
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:
```

Note Auto-advise and auto-copy identify which images are running by examining the info file and by searching the directory structure on the switch stack. If you download your image by using the **copy tftp:** boot loader command instead of the **archive download-sw** privileged EXEC command, the proper directory structure is not created. For more information about the info file, see the *Catalyst 2960-XR Switch Managing Cisco IOS Image Files Configuration Guide*.

SDM Template Mismatch in Switch Stacks

All stack members use the Switch Database Management (SDM) template configured on the active switch. When a new switch is added to a stack, the SDM configuration that is stored on the active switch overrides the template configured on an individual switch.

You can use the **show switch** privileged EXEC command to see if any stack members are in SDM-mismatch mode.

Version-mismatch (VM) mode has priority over SDM-mismatch mode. If a VM-mode condition and an SDM-mismatch mode exist, the switch stack first attempts to resolve the VM-mode condition.

For more information about SDM templates, see the *Catalyst 2960-XR Switch System Management Configuration Guide*.

Switch Stack Management Connectivity

You manage the switch stack and the stack member interfaces through the active switch. You can use the CLI, SNMP, and supported network management applications such as CiscoWorks. You cannot manage stack members on an individual device basis.

Connectivity to Specific Stack Members

If you want to configure a specific stack member port, you must include the stack member number in the CLI command interface notation.

To debug a specific stack member, you can access it from the active stack by using the **session** *stack-member-number* privileged EXEC command. The stack member number is appended to the system prompt. For example, Switch-2# is the prompt in privileged EXEC mode for stack member 2, and the system prompt for the active stack is Switch. Only the **show** and **debug** commands are available in a CLI session to a specific stack member.

Connectivity to the Switch Stack Through an IP Address

The switch stack is managed through a single IP address. The IP address is a system-level setting and is not specific to the active stack or to any other stack member. You can still manage the stack through the same IP address even if you remove the active stack or any other stack member from the stack, provided there is IP connectivity.

![Note icon]

Note   Stack members retain their IP addresses when you remove them from a switch stack. To avoid a conflict by having two devices with the same IP address in your network, change the IP addresses of any active stack that you remove from the switch stack.

For related information about switch stack configurations, see the *Switch Stack Configuration Files section*.

Connectivity to the Switch Stack Through Console Ports or Ethernet Management Ports

You can connect to the active switch by using one of these methods:

- You can connect a terminal or a PC to the active switch through the console port of one or more stack members.

- You can connect a PC to the active switch through the Ethernet management ports of one or more stack members. For more information about connecting to the switch stack through Ethernet management ports, see the *Using the Ethernet Management Port section*.

You can connect to the active switch by connecting a terminal or a PC to the active switch through the console port of one or more stack members.

Be careful when using multiple CLI sessions to the active switch. Commands that you enter in one session are not displayed in the other sessions. Therefore, it is possible that you might not be able to identify the session from which you entered a command.

We recommend using only one CLI session when managing the switch stack.

How to Configure a Switch Stack

Enabling the Persistent MAC Address Feature

✎

Note   When you enter the command to configure this feature, a warning message appears with the consequences of your configuration. You should use this feature cautiously. Using the old active switch MAC address elsewhere in the same domain could result in lost traffic.

Follow these steps to enable persistent MAC address:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **stack-mac persistent timer** [**0** | *time-value*]
4. **end**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Device> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Device# **configure terminal**	Enters global configuration mode.
Step 3	**stack-mac persistent timer** [**0**	*time-value*] **Example:** Device(config)# **stack-mac persistent timer 7**	Enables a time delay after a stack-active switch change before the stack MAC address changes to that of the new ac. If the previous active switch rejoins the stack during this period, the stack uses that MAC address as the stack MAC address. You can configure the time period as 0 to 60 minutes. • Enter the command with no value to set the default delay of approximately 4 minutes. We recommend that you always enter a value.

	Command or Action	Purpose
		If the command is entered without a value, the time delay appears in the running-config file with an explicit timer value of 4 minutes.
		• Enter **0** to continue using the MAC address of the current active switch indefinitely.
		The stack MAC address of the previous active switch is used until you enter the **no stack-mac persistent timer** command, which immediately changes the stack MAC address to that of the current active switch.
		• Enter a *time-value* from 1 to 60 minutes to configure the time period before the stack MAC address changes to the new active switch.
		The stack MAC address of the previous active switch is used until the configured time period expires or until you enter the **no stack-mac persistent timer** command.
		Note If you enter the **no stack-mac persistent timer** command after a new active switch takes over, before the time expires, the switch stack moves to the current active switch MAC address.
Step 4	**end** **Example:** Device(config)# **end**	Returns to privileged EXEC mode.
Step 5	**copy running-config startup-config** **Example:** Device# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

What to do next

Use the **no stack-mac persistent timer** global configuration command to disable the persistent MAC address feature.

Assigning a Stack Member Number

This optional task is available only from the active stack.

Follow these steps to assign a member number to a stack member:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **switch** *current-stack-member-number* **renumber** *new-stack-member-number*
4. **end**
5. **reload slot** *stack-member-number*
6. **show switch**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**switch** *current-stack-member-number* **renumber** *new-stack-member-number* **Example:** Switch(config)# **switch 3 renumber 4**	Specifies the current stack member number and the new stack member number for the stack member. The range is 1 to 8. You can display the current stack member number by using the **show switch** user EXEC command.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**reload slot** *stack-member-number* **Example:** Switch# **reload slot 4**	Resets the stack member.
Step 6	**show switch** **Example:** **show**Switch	Verify the stack member number.
Step 7	**copy running-config startup-config** **Example:**	(Optional) Saves your entries in the configuration file.

Command or Action	Purpose
Switch# **copy running-config startup-config**	

Setting the Stack Member Priority Value

This optional task is available only from the active stack.

Follow these steps to assign a priority value to a stack member:

SUMMARY STEPS

1. **enable**
2. **switch** *stack-member-number* **priority** *new-priority-number*
3. **show switch** *stack-member-number*
4. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch **enable**	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**switch** *stack-member-number* **priority** *new-priority-number* **Example:** Switch# **switch 3 priority 2**	Specifies the stack member number and the new priority for the stack member. The stack member number range is 1 to 8. The priority value range is 1 to 15. You can display the current priority value by using the **show switch** user EXEC command. The new priority value takes effect immediately but does not affect the current active stack. The new priority value helps determine which stack member is elected as the new active stack when the current active stack or switch stack resets.
Step 3	**show switch** *stack-member-number* **Example:** Switch# **show switch**	Verify the stack member priority value.
Step 4	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Provisioning a New Member for a Switch Stack

This optional task is available only from the active switch.

SUMMARY STEPS

1. **show switch**
2. **configure terminal**
3. **switch** *stack-member-number* **provision** *type*
4. **end**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**show switch** **Example:** Switch# **show switch**	Displays summary information about the switch stack.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**switch** *stack-member-number* **provision** *type* **Example:** Switch(config)# **switch 3 provision WS-xxxx**	Specifies the stack member number for the preconfigured switch. By default, no switches are provisioned. For *stack-member-number*, the range is 1 to 8. Specify a stack member number that is not already used in the switch stack. See Step 1. For *type*, enter the model number of a supported switch that is listed in the command-line help strings.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Removing Provisioned Switch Information

Before you begin, you must remove the provisioned switch from the stack. This optional task is available only from the active stack.

SUMMARY STEPS

1. **configure terminal**
2. **no switch** *stack-member-number* **provision**
3. **end**
4. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**no switch** *stack-member-number* **provision** **Example:** Switch(config)# **no switch 3 provision**	Removes the provisioning information for the specified member.
Step 3	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 4	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Example

If you are removing a provisioned switch in a stack with this configuration:

- The stack has four members
- Stack member 1 is the active stack
- Stack member 3 is a provisioned switch

and want to remove the provisioned information and to avoid receiving an error message, you can remove power from stack member 3, disconnect the stack cables between the stack member 3 and switches to which it is connected, reconnect the cables between the remaining stack members, and enter the **no switch** *stack-member-number* **provision** global configuration command.

Troubleshooting the Switch Stack

Accessing the CLI of a Specific Member

This optional task is for debugging purposes, and is available only from the active switch.

You can access all or specific members by using the **remote command** {**all** | *stack-member-number*} privileged EXEC command. The stack member number range is 1 to 8.

You can access specific members by using the **session** *stack-member-number* privileged EXEC command. The member number is appended to the system prompt. For example, the prompt for member 2 is `Switch-2#`, and system prompt for the active switch is `Switch#`. Enter **exit** to return to the CLI session on the active switch. Only the **show** and **debug** commands are available on a specific member.

Temporarily Disabling a Stack Port

If a stack port is flapping and causing instability in the stack ring, to disable the port, enter the **switch** *stack-member-number* **stack port** *port-number* **disable** privileged EXEC command. To reenable the port, enter the **switch** *stack-member-number* **stack port** *port-number* **enable** command.

> **Note** Be careful when using the **switch** *stack-member-number* **stack port** *port-number* **disable** command. When you disable the stack port, the stack operates at half bandwidth.

A stack is in the full-ring state when all members are connected through the stack ports and are in the ready state.

The stack is in the partial-ring state when the following occurs:

- All members are connected through their stack ports but some are not in the ready state.

- Some members are not connected through the stack ports.

SUMMARY STEPS

1. **switch** *stack-member-number* **stack port** *port-number* **disable**
2. **switch** *stack-member-number* **stack port** *port-number* **enable**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**switch** *stack-member-number* **stack port** *port-number* **disable** **Example:** `Switch# switch 2 stack port 1 disable`	Disables the specified stack port.

	Command or Action	Purpose
Step 2	**switch** *stack-member-number* **stack port** *port-number* **enable** **Example:** `Switch# `**`switch 2 stack port 1 enable`**	Reenables the stack port.

When you disable a stack port and the stack is in the full-ring state, you can disable only one stack port. This message appears:

```
Enabling/disabling a stack port may cause undesired stack changes. Continue?[confirm]
```

When you disable a stack port and the stack is in the partial-ring state, you cannot disable the port. This message appears:

```
Disabling stack port not allowed with current stack configuration.
```

Reenabling a Stack Port While Another Member Starts

Stack Port 1 on Switch 1 is connected to Port 2 on Switch 4. If Port 1 is flapping, you can disable Port 1 with the **switch 1 stack port 1 disable** privileged EXEC command. While Port 1 on Switch 1 is disabled and Switch 1 is still powered on, follow these steps to reenable a stack port:

Step 1 Disconnect the stack cable between Port 1 on Switch 1 and Port 2 on Switch 4.

Step 2 Remove Switch 4 from the stack.

Step 3 Add a switch to replace Switch 4 and assign it switch-number 4.

Step 4 Reconnect the cable between Port 1 on Switch 1 and Port 2 on Switch 4 (the replacement switch).

Step 5 Reenable the link between the switches. Enter the **switch 1 stack port 1 enable** privileged EXEC command to enable Port 1 on Switch 1.

Step 6 Power on Switch 4.

⚠️

Caution Powering on Switch 4 before enabling the Port 1 on Switch 1 might cause one of the switches to reload.

If Switch 4 is powered on first, you might need to enter the **switch 1 stack port 1 enable** and the **switch 4 stack port 2 enable** privileged EXEC commands to bring up the link.

Monitoring the Switch Stack

Table 109: Commands for Displaying Stack Information

Command	Description
**show controller ethernet-controller stack port {1	2}**	Displays stack port counters (or per-interface and per-stack port send and receive statistics read from the hardware).
show controller ethernet-controller fastethernet0	Displays information about the Ethernet management port, including the port status and the per-interface send and receive statistics read from the hardware.
show platform stack compatibility	Displays information about HULC feature compatibility.
show platform stack manager all	Displays all stack manager information, such as the stack protocol version.
show platform stack passive-links	Displays information about stack passive links.
show switch	Displays summary information about the stack, including the status of provisioned switches and switches in version-mismatch mode.
show switch *stack-member-number*	Displays information about a specific member.
show switch detail	Displays detailed information about the stack.
show switch neighbors	Displays the stack neighbors.
show switch stack-ports	Displays port information for the stack.

Configuration Examples for Switch Stacks

Switch Stack Configuration Scenarios

Most of these switch stack configuration scenarios assume that at least two switch are connected through their stack ports.

Table 110: Configuration Scenarios

Scenario		Result
Active switch election specifically determined by existing active switches	Connect two powered-on switch stacks through the stack ports.	Only one of the two active switches becomes the new active switch.

Scenario		Result
Active switch election specifically determined by the stack member priority value	1. Connect two switches through their stack ports. 2. Use the **switch** *stack-member-number* **priority** *new-priority-number* global configuration command to set one stack member with a higher member priority value. 3. Restart both stack members at the same time.	The stack member with the higher priority value is elected active switch.
Active switch election specifically determined by the configuration file	Assuming that both stack members have the same priority value: 1. Make sure that one stack member has a default configuration and that the other stack member has a saved (nondefault) configuration file. 2. Restart both stack members at the same time.	The stack member with the saved configuration file is elected active switch.
Active switch election specifically determined by the MAC address	Assuming that both stack members have the same priority value, configuration file, and feature set, restart both stack members at the same time.	The stack member with the lower MAC address is elected active switch.
Stack member number conflict	Assuming that one stack member has a higher priority value than the other stack member: 1. Ensure that both stack members have the same stack member number. If necessary, use the **switch** *current-stack-member-number* **renumber** *new-stack-member-number* global configuration command. 2. Restart both stack members at the same time.	The stack member with the higher priority value retains its stack member number. The other stack member has a new stack member number.
Add a stack member	1. Power off the new switch. 2. Through their stack ports, connect the new switch to a powered-on switch stack. 3. Power on the new switch.	The active switch is retained. The new switch is added to the switch stack.

Scenario		Result
Active switch failure	Remove (or power off) the active switch.	One of the remaining stack members becomes the new active switch. All other stack members in the stack remain as stack members and do not reboot.
Add more than eight stack members	1. Through their stack ports, connect nine switch. 2. Power on all switch.	Two switch become active switches. One active switch has eight stack members. The other active switch remains as a standalone switch. Use the Mode button and port LEDs on the switch to identify which switch are active switches and which switch belong to each active switch.

Enabling the Persistent MAC Address Feature: Example

This example shows how to configure the persistent MAC address feature for a 7-minute time delay and to verify the configuration:

```
Switch(config)# stack-mac persistent timer 7
  WARNING: The stack continues to use the base MAC of the old Master
  WARNING: as the stack MAC after a master switchover until the MAC
  WARNING: persistency timer expires. During this time the Network
  WARNING: Administrators must make sure that the old stack-mac does
  WARNING: not appear elsewhere in this network domain. If it does,
  WARNING: user traffic may be blackholed.
Switch(config)# end
Switch# show switch
Switch/Stack Mac Address : 0016.4727.a900
Mac persistency wait time: 7 mins
                                     H/W    Current
  Switch#  Role   Mac Address     Priority Version  State
  -----------------------------------------------------------
  *1       Master 0016.4727.a900    1      P2B      Ready
```

Provisioning a New Member for a Switch Stack: Example

This example shows how to provision a switch with a stack member number of 2 for the switch stack. The **show running-config** command output shows the interfaces associated with the provisioned switch:

```
Switch(config)# switch 2 provision switch_PID
  Switch(config)# end
  Switch# show running-config | include switch 2
  switch 2 provision switch_PID
```

Additional References for Switch Stacks

Related Documents

Related Topic	Document Title
Cabling and powering on a switch stack.	*Catalyst 2960-XR Switch Hardware Installation Guide*

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and software images, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

PART XI

VLAN

Configuring VTP

Prerequisites for VTP

The following are prerequisites for VTP:

- Before you create VLANs, you must decide whether to use the VLAN Trunking Protocol (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 when running the IP Lite image.

- The switch supports 255 VLANs when running the LAN base feature set.

- However, the number of routed ports, SVIs, and other 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.

Information About VTP

VTP

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.

VTP functionality is supported across the stack, and all switches in the stack maintain the same VLAN and VTP configuration inherited from the active switch. When a switch learns of a new VLAN through VTP messages or when a new VLAN is configured by the user, the new VLAN information is communicated to all switches in the stack.

When a switch joins the stack or when stacks merge, the new switches get VTP information from the active switch.

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 4094). Extended range VLANs (VLANs 1006 to 4094) 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 or switch stacks 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.

VTP Modes

Table 111: 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.
	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 functions 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 Mode	Description
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 private VLANs and when they are configured, you should not change the VTP mode from transparent to client or server mode. VTP version 3 also supports private VLANs in client and server modes. When private VLANs are configured, do not change the VTP mode from transparent to 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.
	In a switch stack, the running configuration and the saved configuration are the same for all switches in a stack.
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 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.

Because trunk ports send and receive VTP advertisements, you must ensure that at least one trunk port is configured on the switch stack and that this trunk port is connected to the trunk port of another switch. Otherwise, the switch cannot receive any VTP advertisements.

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 (including 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.

- 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 4094) database propagation—VTP versions 1 and 2 propagate only VLANs 1 to 1005.

> **Note** VTP pruning still applies only to VLANs 1 to 1005, and VLANs 1002 to 1005 are still reserved and cannot be modified.

- Private VLAN support.

- 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.

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.

With VTP versions 1 and 2, when you enable pruning on the VTP server, it is enabled for the entire VTP domain. In VTP version 3, you must manually enable pruning on each switch in the 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).

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 and Switch Stacks

VTP configuration is the same in all members of a switch stack. When the switch stack is in VTP server or client mode, all switches in the stack carry the same VTP configuration. When VTP mode is transparent, the stack is not taking part in VTP.

- When a switch joins the stack, it inherits the VTP and VLAN properties of the active stack.

- All VTP updates are carried across the stack.

- When VTP mode is changed in a switch in the stack, the other switches in the stack also change VTP mode, and the switch VLAN database remains consistent.

VTP version 3 functions the same on a standalone switch or a stack except when the switch stack is the primary server for the VTP database. In this case, the MAC address of the active stack is used as the primary server ID. If the active stack reloads or is powered off, a new active stack is elected.

- If you do not configure the persistent MAC address feature (by entering the **stack-mac persistent timer** [**0** | *time-value*] global configuration command, when the new active stack is elected, it sends a takeover message with the new active stack MAC address as the primary server.

- If persistent MAC address is configured, the new active stack waits for the configured **stack-mac persistent timer** value. If the previous active stack does not rejoin the stack during this time, then the new active stack issues the takeover message.

VTP Configuration Guidelines

Configuration Requirements

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.

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. For more information about the command, see the command reference for this release.

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.

VTP versions 1 and 2 do not support private VLANs. VTP version 3 does support private VLANs. If you configure private VLANs, the switch must be in VTP transparent mode. When private VLANs are configured on the switch, do not change the VTP mode from transparent to client or server mode.

VTP Settings

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, even if the switch resets.

When you save VTP information in the switch 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 do not match the VLAN database, the domain name and VTP mode and configuration for VLAN IDs 1 to 1005 use the VLAN database information.

Domain Names for Configuring VTP

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 the 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 for the VTP Domain

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.

VTP Version

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.

- Cisco recommends 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 4094). You must configure these VLANs manually on each device. VTP version 3 supports extended-range VLANs and support for extended range VLAN database propagation.

- 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 1 and version 2 do not propagate configuration information for extended range VLANs (VLANs 1006 to 4094). You must manually configure these VLANs on each device.

- 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.

⚠
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.

Default VTP Configuration

The following table shows the default VTP configuration.

Table 112: Default VTP Configuration

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

How to Configure VTP

Configuring VTP Mode

You can configure VTP mode as one of these:

- VTP server mode—In VTP server mode, you can change the VLAN configuration and have it propagated throughout the network.

- VTP client mode—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.

- VTP transparent mode—In 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 switch. However, a VTP transparent switch running VTP version 2 does forward received VTP advertisements on its trunk links.

- VTP off mode—VTP off mode is the same as VTP transparent mode except that VTP advertisements are not forwarded.

When you configure a domain name, it cannot be removed; you can only reassign a switch to a different domain.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vtp domain** *domain-name*
4. **vtp mode** {**client** | **server** | **transparent** | **off**} {**vlan** | **mst** | **unknown**}
5. **vtp password** *password*

6. **end**
7. **show vtp status**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**vtp domain** *domain-name* **Example:** Switch(config)# **vtp domain eng_group**	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. You should configure the VTP domain before configuring other VTP parameters.

Step 4	**vtp mode** {**client** | **server** | **transparent** | **off**} {**vlan** | **mst** | **unknown**}
	
	Example:
	
	Switch(config)# **vtp mode server**
		Configures the switch for VTP mode (client, server, transparent, or off).
		
		  • **vlan**—The VLAN database is the default if none are configured.
		
		  • **mst**—The multiple spanning tree (MST) database.
		
		  • **unknown**—An unknown database type.
		
		Note To return a switch in another mode to VTP server mode, use the **no vtp mode** global configuration command.

Step 5	**vtp password** *password* **Example:** Switch(config)# **vtp password mypassword**	(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.

	Command or Action	**Purpose**
		Note To return the switch to a no-password state, use the **no vtp password** global configuration command.
Step 6	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 7	**show vtp status** **Example:** Switch# **show vtp status**	Verifies your entries in the *VTP Operating Mode* and the *VTP Domain Name* fields of the display.
Step 8	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves the configuration in the startup configuration file. 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

You can configure a VTP version 3 password on the switch.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vtp password** *password* [**hidden** | **secret**]
4. **end**
5. **show vtp password**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:**	Enters global configuration mode.

	Command or Action	Purpose
	Switch# **configure terminal**	
Step 3	**vtp password** *password* [**hidden** | **secret**] Example: Switch(config)# **vtp password mypassword hidden**	(Optional) Sets the password for the VTP domain. The password can be 8 to 64 characters. • (Optional) **hidden**—Saves the secret key generated from the password string in the nvram: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. **Note** To clear the password, enter the **no vtp password** global configuration command.
Step 4	**end** Example: Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show vtp password** Example: Switch# **show vtp password**	Verifies your entries. The output appears like this: VTP password: 89914640C8D90868B6A0D8103847A733
Step 6	**copy running-config startup-config** Example: Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring a VTP Version 3 Primary Server

When you configure a VTP server as a VTP primary server, the takeover operation starts.

SUMMARY STEPS

1. **vtp primary** [**vlan** \| **mst**] [**force**]

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**vtp primary** [**vlan** | **mst**] [**force**] **Example:** Switch# **vtp primary vlan force**	Changes the operational state of a switch from a secondary server (the default) to a primary server and advertises the configuration to the domain. If the switch password is configured as **hidden**, you are prompted to reenter the password. • (Optional) **vlan**—Selects the VLAN database as the takeover feature. This is the default. • (Optional) **mst**—Selects the multiple spanning tree (MST) database as the takeover feature. • (Optional) **force**—Overwrites the configuration of any conflicting servers. If you do not enter **force**, you are prompted for confirmation before the takeover.

Enabling the VTP Version

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** 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 TrCRF and TrBRF Token Ring environments, you must enable VTP version 2 or VTP version 3 for Token Ring VLAN switching to function properly. For Token Ring and Token Ring-Net media, disable VTP version 2.

• ⚠
> **Caution** In VTP version 3, both the primary and secondary servers can exist on an instance in the domain.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vtp version** {**1** \| **2** \| **3**}
4. **end**

5. show vtp status

6. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**vtp version {1	2	3}** **Example:** Switch(config)# **vtp version 2**	Enables the VTP version on the switch. The default is VTP version 1. **Note** To return to the default VTP version 1, use the **no vtp version** global configuration command.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show vtp status** **Example:** Switch# **show vtp status**	Verifies that the configured VTP version is enabled.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Enabling VTP Pruning

Before you begin

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 actions:

• 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.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vtp pruning**
4. **end**
5. **show vtp status**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**vtp pruning** **Example:** Switch(config)# **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. **Note** To disable VTP pruning, use the **no vtp pruning** global configuration command.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show vtp status** **Example:**	Verifies your entries in the *VTP Pruning Mode* field of the display.

Command or Action	Purpose
Switch# **show vtp status**	

Configuring VTP on a Per-Port Basis

With VTP version 3, you can enable or disable VTP on a per-port basis. You can enable VTP only on ports that are in trunk mode. Incoming and outgoing VTP traffic are blocked, not forwarded.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **vtp**
5. **end**
6. **show running-config interface** *interface-id*
7. **show vtp status**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	Identifies an interface, and enters interface configuration mode.
Step 4	**vtp** **Example:** Switch(config)# **vtp**	Enables VTP on the specified port. **Note** To disable VTP on the interface, use the **no vtp** interface configuration command.

	Command or Action	**Purpose**
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**show running-config interface** *interface-id* **Example:** Switch# **show running-config interface gigabitethernet1/0/1**	Verifies the change to the port.
Step 7	**show vtp status** **Example:** Switch# **show vtp status**	Verifies the configuration.

Adding a VTP Client Switch to a VTP Domain

Follow these steps to verify and reset the VTP configuration revision number on a switch *before* adding it 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.

You can use the **vtp mode transparent** global configuration command to disable VTP on the switch and then to change its VLAN information without affecting the other switches in the VTP domain.

SUMMARY STEPS

1. **enable**
2. **show vtp status**
3. **configure terminal**
4. **vtp domain** *domain-name*
5. **end**
6. **show vtp status**
7. **configure terminal**
8. **vtp domain** *domain-name*
9. **end**
10. **show vtp status**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**show vtp status** **Example:** Switch# **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 substeps: • Write down the domain name. • Write down the configuration revision number. • Continue with the next steps to reset the switch configuration revision number.
Step 3	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 4	**vtp domain** *domain-name* **Example:** Switch(config)# **vtp domain domain123**	Changes the domain name from the original one displayed in Step 1 to a new name.
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode. The VLAN information on the switch is updated and the configuration revision number is reset to 0.
Step 6	**show vtp status** **Example:** Switch# **show vtp status**	Verifies that the configuration revision number has been reset to 0.
Step 7	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.

	Command or Action	**Purpose**
Step 8	**vtp domain** *domain-name* **Example:** Switch(config)# **vtp domain domain012**	Enters the original domain name on the switch
Step 9	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode. The VLAN information on the switch is updated.
Step 10	**show vtp status** **Example:** Switch# **show vtp status**	(Optional) Verifies that the domain name is the same as in Step 1 and that the configuration revision number is 0.

Monitoring VTP

This section describes commands used to display and monitor the VTP configuration.

You monitor VTP by displaying VTP configuration information: the domain name, the current VTP revision, and the number of VLANs. You can also display statistics about the advertisements sent and received by the switch.

Table 113: VTP Monitoring Commands

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 VTP

Example: Configuring the Switch as a VTP Server

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
```

Example: Configuring a Hidden Password

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
```

Example: Configuring a VTP Version 3 Primary Server

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
```

Example: Configuring VTP on a Per-Port Basis

This example shows how to configure VTP on a per-port basis:

```
Switch(config)# interface gigabitethernet 1/0/1
```

```
Switch(config-if)# vtp
Switch(config-if)# end
```

Where to Go Next

After configuring VTP, you can configure the following:

- VLANs

- VLAN trunking

- Private VLANs

- VLAN Membership Policy Server (VMPS)

- Tunneling

- Voice VLANs

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	*Catalyst 2960-XR Switch VLAN Management Command Reference*

Standards and RFCs

Standard/RFC	Title
—	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for VTP

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Configuring VLANs

Prerequisites for VLANs

The following are prerequisites and considerations for configuring VLANs:

- The switch supports 1005 VLANs when running the IP Lite image.

- The switch supports up to 1000 VLANs running the LAN Base image.

- The switch supports 256 SVIs when running the IP Lite image.

Restrictions for VLANs

The following are the restrictions for configuring VLANs:

- The switch supports homogeneous stacking, but does not support mixed stacking.

Information About VLANs

Logical Networks

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. In a switch stack, VLANs can be formed with ports across the stack. 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.

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 routedor 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.

Supported VLANs

The switch supports VLANs in VTP client, server, and transparent modes. VLANs are identified by a number from 1 to 4094. 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 4094. . VTP version 3 supports the entire VLAN range (VLANs 1 to 4094). Extended range VLANs (VLANs 1006 to 4094) 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.

The switch or switch stack supports a total of 1005 (normal range and extended range) VLANs. However, the number of routed ports, SVIs, and other configured features affects the use of the switch hardware.

The switch or switch stack supports a total of 255 VLANs when running the LAN base feature set. However, 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.

The switch supports IEEE 802.1Q trunking methods for sending VLAN traffic over Ethernet ports.

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.

When a port belongs to a VLAN, the switch learns and manages the addresses associated with the port on a per-VLAN basis.

Table 114: 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.	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 or the switch stack connected to a trunk port of a second switch or switch stack.
Trunk (IEEE 802.1Q) : • IEEE 802.1Q—Industry-standard trunking encapsulation.	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.	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 (VLAN ID 1 to 4094) and is dynamically assigned by a VLAN Member Policy Server (VMPS). 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.	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 or a switch stack must be connected to a trunk port of a second switch or switch stack.
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.	VTP is not required; it has no effect on a voice VLAN.

VLAN Configuration Files

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. If the VTP mode is transparent, they are also saved in the switch running configuration file.

In a switch stack, the whole stack uses the same vlan.dat file and running configuration. On some switches, the vlan.dat file is stored in flash memory on the active switch.

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.

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 VLAN IDs 1 to 1005 use the VLAN database information.

- In VTP versions 1 and 2, if VTP mode is server, the domain name and VLAN configuration for VLAN IDs 1 to 1005 use the VLAN database information. VTP version 3 also supports VLANs 1006 to 4094.

- From image 15.0(02)SE6, on vtp transparent and off modes, vlans get created from startup-config even if they are not applied to the interface.

Note   Ensure that you delete the vlan.dat file along with the configuration files before you reset the switch configuration using **write erase** command. This ensures that the switch reboots correctly on a reset.

Normal-Range VLAN Overview

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.)

In VTP versions 1 and 2, the switch must be in VTP transparent mode when you create extended-range VLANs (VLANs with IDs from 1006 to 4094), but these VLANs are not saved in the VLAN database. VTP version 3 supports extended-range VLANs in VTP server and transparent mode.

Configurations for VLAN IDs 1 to 1005 are written to the file vlan.dat (VLAN database), and you can display them by entering the **show vlan** privileged EXEC command. The vlan.dat file is stored in flash memory. On a switch, the vlan.dat file is stored in flash memory on the active stack. Stack members have a vlan.dat file that is consistent with the active stack.

Token Ring VLANs

Although the switch does not support Token Ring connections, a remote device such as a Catalyst 5000 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

Note   For more information on configuring Token Ring VLANs, see the *Catalyst 5000 Series Software Configuration Guide*.

Normal-Range VLANs Configuration Process

You configure VLANs in the **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 or enter multiple commands to configure the VLAN. For more information about commands available in this mode, see the **vlan** global configuration command description in the command reference for this release. 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.

VLAN Configuration Saving Process

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. In a switch stack, the whole stack uses the same vlan.dat file and running configuration. 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 VLAN IDs 1 to 1005 use the VLAN database information.

- In VTP versions 1 and 2, if VTP mode is server, the domain name and VLAN configuration for VLAN IDs 1 to 1005 use the VLAN database information. VTP version 3 also supports VLANs 1006 to 4094.

Normal-Range VLAN Configuration Guidelines

Normal-range VLANs are VLANs with IDs from 1 to 1005.

VTP 1 and 2 only support normal-range VLANs.

Follow these guidelines when creating and modifying normal-range VLANs in your network:

- 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 configurations for VLANs 1 to 1005 are always saved in the VLAN database. If the VTP mode is transparent, VTP and VLAN configurations are also saved in the switch running configuration file.

- 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.)

- With VTP versions 1 and 2, the switch supports VLAN IDs 1006 through 4094 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 4094) database propagation in VTP server mode. If extended VLANs are configured, you cannot convert from VTP version 3 to version 1 or 2.

• 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.

• When a switch in a stack learns a new VLAN or deletes or modifies an existing VLAN (either through VTP over network ports or through the CLI), the VLAN information is communicated to all stack members.

• When a switch joins a stack or when stacks merge, VTP information (the vlan.dat file) on the new switches will be consistent with the active switch.

Related Topics

Extended-Range VLAN Configuration Guidelines

Extended-range VLANs are VLANs with IDs from 1006 to 4094.

VTP 3 only supports extended-range VLANs.

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.

• For VTP version 1 or 2, you can set the VTP mode to transparent in global configuration mode. 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.

- 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.

Note   Switchesrunning the LAN Base feature set support only static routing on SVIs.

- 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 (4094) 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.

- Although the switch orswitch stack 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.

- In a switch stack, the whole stack uses the same running configuration and saved configuration, and extended-range VLAN information is shared across the stack.

Related Topics

Creating an Extended-Range VLAN
Monitoring VLANs

Default Ethernet VLAN Configuration

The following table displays the default configuration for Ethernet VLANs.

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 115: Ethernet VLAN Defaults and Range

Parameter	Default	Range
VLAN ID	1	1 to 4094. **Note** Extended-range VLANs (VLAN IDs 1006 to 4094) 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
IEEE 802.10 SAID	1500	576-18190
Private VLANs	none configured	2 to 1001, 1006 to 4094

Default VLAN Configuration

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.

How to Configure VLANs

How to Configure Normal-Range VLANs

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)

> • 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 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, follow the procedures in this section.

Creating or Modifying an Ethernet VLAN

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.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vlan** *vlan-id*
4. **name** *vlan-name*
5. **mtu** *mtu-size*
6. **remote-span**
7. **end**
8. **show vlan** {**name** *vlan-name* | **id** *vlan-id*}
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	**Purpose**
Step 2	**configure terminal**	Enters global configuration mode.
	Example:	
	Switch# **configure terminal**	
Step 3	**vlan** *vlan-id*	Enters a VLAN ID, and enters VLAN configuration mode. Enter a new VLAN ID to create a VLAN, or enter an existing VLAN ID to modify that VLAN.
	Example:	
	Switch(config)# **vlan 20**	**Note** The available VLAN ID range for this command is 1 to 4094.
Step 4	**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* value with leading zeros to the word VLAN. For example, VLAN0004 is a default VLAN name for VLAN 4.
	Example:	
	Switch(config-vlan)# **name test20**	
Step 5	**mtu** *mtu-size*	(Optional) Changes the MTU size (or other VLAN characteristic).
	Example:	
	Switch(config-vlan)# **mtu 256**	
Step 6	**remote-span**	(Optional) Configures the VLAN as the RSPAN VLAN for a remote SPAN session.
	Example:	
	Switch(config-vlan)# **remote-span**	**Note** To return the VLAN name to the default settings, use the **no name**, **no mtu**, or **no remote-span** commands.
Step 7	**end**	Returns to privileged EXEC mode.
	Example:	
	Switch(config)# **end**	
Step 8	**show vlan** {**name** *vlan-name* | **id** *vlan-id*}	Verifies your entries.
	Example:	
	Switch# **show vlan name test20 id 20**	
Step 9	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.
	Example:	
	Switch# **copy running-config startup-config**	

Deleting a VLAN

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 or a switch stack.

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.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no vlan** *vlan-id*
4. **end**
5. **show vlan brief**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**no vlan** *vlan-id* **Example:** Switch(config)# **no vlan 4**	Removes the VLAN by entering the VLAN ID.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

	Command or Action	**Purpose**
Step 5	**show vlan brief** **Example:** Switch# **show vlan brief**	Verifies the VLAN removal.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Related Topics

Assigning Static-Access Ports to 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.

If you assign an interface to a VLAN that does not exist, the new VLAN is created.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode access**
4. **switchport access vlan** *vlan-id*
5. **end**
6. **show running-config interface** *interface-id*
7. **show interfaces** *interface-id* **switchport**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:**	Enters the interface to be added to the VLAN.

	Command or Action	Purpose
	`Switch(config)# interface gigabitethernet2/0/1`	
Step 3	**switchport mode access** **Example:** `Switch(config-if)# switchport mode access`	Defines the VLAN membership mode for the port (Layer 2 access port).
Step 4	**switchport access vlan** *vlan-id* **Example:** `Switch(config-if)# switchport access vlan 2`	Assigns the port to a VLAN. Valid VLAN IDs are 1 to 4094. **Note** To return an interface to its default configuration, use the **default interface interface-id** interface configuration command.
Step 5	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 6	**show running-config interface** *interface-id* **Example:** `Switch# show running-config interface gigabitethernet2/0/1`	Verifies the VLAN membership mode of the interface.
Step 7	**show interfaces** *interface-id* **switchport** **Example:** `Switch# show interfaces gigabitethernet2/0/1 switchport`	Verifies your entries in the *Administrative Mode* and the *Access Mode VLAN* fields of the display.

How to Configure Extended-Range VLANs

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. Extended-range VLANs created in VTP version 3 are stored in the VLAN database.

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.

Creating an Extended-Range VLAN

You create an extended-range VLAN in global configuration mode by entering the **vlan** global configuration command with a VLAN ID from 1006 to 4094. The extended-range VLAN has the default Ethernet VLAN characteristics and the MTU size, and RSPAN configuration are the only parameters you can change. See the description of the **vlan** global configuration command in the command reference for the default settings of all parameters. In VTP version 1 or 2, if you enter an extended-range VLAN ID when the switch is not in VTP transparent mode, an error message is generated when you exit VLAN configuration mode, and the extended-range VLAN is not created.

In VTP version 1 and 2, extended-range VLANs are not saved in the VLAN database; they are saved in the switch running configuration file. You can save the extended-range VLAN configuration in the switch startup configuration file by using the **copy running-config startup-config** privileged EXEC command. VTP version 3 saves extended-range VLANs in the VLAN database.

Note   Before you create an extended-range VLAN, you can verify that the VLAN ID is not used internally by entering the **show vlan internal usage** privileged EXEC command. If the VLAN ID is used internally and you want to free it up, go to the *Creating an Extended-Range VLAN with an Internal VLAN ID* before creating the extended-range VLAN.

SUMMARY STEPS

1. **configure terminal**
2. **vtp mode transparent**
3. **vlan** *vlan-id*
4. **mtu** *mtu size*
5. **remote-span**
6. **end**
7. **show vlan id** *vlan-id*
8. **copy running-config startup config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**vtp mode transparent** **Example:** Switch(config)# **vtp mode transparent**	Configures the switch for VTP transparent mode, disabling VTP. **Note** This step is not required for VTP version 3.
Step 3	**vlan** *vlan-id* **Example:**	Enters an extended-range VLAN ID and enters VLAN configuration mode. The range is 1006 to 4094.

	Command or Action	Purpose
	`Switch(config)# vlan 2000` `Switch(config-vlan)#`	**Note** To delete an extended-range VLAN, use the **no vlan** *vlan-id* global configuration command.
Step 4	**mtu** *mtu size* **Example:** `Switch(config-vlan)# mtu 1024`	Modifies the VLAN by changing the MTU size.
Step 5	**remote-span** **Example:** `Switch(config-vlan)# remote-span`	(Optional) Configures the VLAN as the RSPAN VLAN.
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode.
Step 7	**show vlan id** *vlan-id* **Example:** `Switch# show vlan id 2000`	Verifies that the VLAN has been created.
Step 8	**copy running-config startup config** **Example:** `Switch# copy running-config startup-config`	Saves your entries in the switch startup 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. **Note** This step is not required for VTP version 3 because VLANs are saved in the VLAN database. The procedure for assigning static-access ports to an extended-range VLAN is the same as for normal-range VLANs.

Creating an Extended-Range VLAN with an Internal VLAN ID

If you enter an extended-range VLAN ID that is already assigned to an internal VLAN, an error message is generated, and the extended-range VLAN is rejected. To manually free an internal VLAN ID, you must temporarily shut down the routed port that is using the internal VLAN ID.

![Note icon]

Note	Switches running the LAN Base imagesupport only static routing on SVIs.

SUMMARY STEPS

1. **enable**
2. **show vlan internal usage**
3. **configure terminal**
4. **interface** *interface-id*
5. **shutdown**
6. **exit**
7. **vtp mode transparent**
8. **vlan** *vlan-id*
9. **exit**
10. **interface** *interface-id*
11. **no shutdown**
12. **end**
13. **copy running-config startup config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**show vlan internal usage** **Example:** Switch# **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.
Step 3	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 4	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/3**	Specifies the interface ID for the routed port that is using the VLAN ID, and enters interface configuration mode.

	Command or Action	**Purpose**
Step 5	**shutdown** **Example:** Switch(config-if)# **shutdown**	Shuts down the port to free the internal VLAN ID.
Step 6	**exit** **Example:** Switch(config-if)# **exit**	Returns to global configuration mode.
Step 7	**vtp mode transparent** **Example:** Switch(config)# **vtp mode transparent**	Sets the VTP mode to transparent for creating extended-range VLANs. **Note** This step is not required for VTP version 3.
Step 8	**vlan** *vlan-id* **Example:** Switch(config-vlan)# **vlan 2000**	Enters the new extended-range VLAN ID, and enters VLAN configuration mode.
Step 9	**exit** **Example:** Switch(config-vlan)# **exit**	Exits from VLAN configuration mode, and returns to global configuration mode.
Step 10	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/3**	Specifies the interface ID for the routed port that you shut down in Step 4, and enters interface configuration mode.
Step 11	**no shutdown** **Example:** Switch(config)# **no shutdown**	Reenables the routed port. It will be assigned a new internal VLAN ID.
Step 12	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

	Command or Action	**Purpose**
Step 13	**copy running-config startup config** **Example:** Switch# `copy running-config startup-config`	Saves your entries in the switch startup 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. **Note** This step is not required for VTP version 3 because VLANs are saved in the VLAN database.

Related Topics

> Supported VLANs
> Monitoring VLANs

Monitoring VLANs

Table 116: Privileged EXEC show Commands

Command	**Purpose**
show interfaces [**vlan** *vlan-id*]	Displays characteristics for all interfaces or for the specified VLAN configured on the switch.

Configuration Examples

Example: Creating a VLAN Name

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
```

Example: Configuring a Port as Access Port

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 gigabitethernet 1/0/1
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 2
Switch(config-if)# end
```

Example: Creating an Extended-Range VLAN

This example shows how to create a new extended-range VLAN with all default characteristics, enter VLAN configuration mode, and save the new VLAN in the switch startup configuration file:

```
Switch(config)# vtp mode transparent
Switch(config)# vlan 2000
Switch(config-vlan)# end
Switch# copy running-config startup config
```

Where to Go Next

After configuring VLANs, you can configure the following:

- VLAN Trunking Protocol (VTP)

- VLAN trunks

- Private VLANs

- VLAN Membership Policy Server (VMPS)

- Tunneling

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	*Catalyst 2960-XR Switch VLAN Management Command Reference*

Standards and RFCs

Standard/RFC	Title
—	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for VLAN

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Configuring VLAN Trunks

Prerequisites for VLAN Trunks

The IEEE 802.1Q trunks impose these limitations 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 disabling spanning tree.

Restrictions for VLAN Trunks

The following are restrictions for VLAN trunks:

- 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 that 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 Per VLAN Spanning Tree (PVST) mode and no more than 40 trunk ports in Multiple Spanning Tree (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.

- Dynamic Trunking Protocol (DTP) is not supported on tunnel ports.

- The switch does not support Layer 3 trunks; you cannot configure subinterfaces or use the **encapsulation** keyword on Layer 3 interfaces. The switch does support Layer 2 trunks and Layer 3 VLAN interfaces, which provide equivalent capabilities.

Information About 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.

The following trunking encapsulations are available on all Ethernet interfaces:

- IEEE 802.1Q— Industry-standard trunking encapsulation.

Trunking Modes

Ethernet trunk interfaces support different trunking modes. 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 (PPP). 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.

Layer 2 Interface Modes

Table 117: 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 switchport 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.
switchport mode dot1q-tunnel	Configures the interface as a tunnel (nontrunking) port to be connected in an asymmetric link with an IEEE 802.1Q trunk port. The IEEE 802.1Q tunneling is used to maintain customer VLAN integrity across a service provider network.

Mode	Function
switchport mode private-vlan	Configures the private VLAN mode. **Note** The **switchport mode private-vlan** command option is not supported.

Allowed VLANs on a Trunk

By default, a trunk port sends traffic to and receives traffic from all VLANs. All VLAN IDs, 1 to 4094, 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 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 is true 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.

Load Sharing on 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.

Network Load Sharing Using STP 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.

Network 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.

Feature Interactions

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 that 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 Per VLAN Spanning Tree (PVST) mode and no more than 40 trunk ports in Multiple Spanning Tree (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.

Default Layer 2 Ethernet Interface VLAN Configuration

The following table shows the default Layer 2 Ethernet interface VLAN configuration.

Table 118: Default Layer 2 Ethernet Interface VLAN Configuration

Feature	Default Setting
Interface mode	**switchport mode dynamic auto**
Trunk encapsulation	**switchport trunk encapsulation negotiate**
Allowed VLAN range	VLANs 1 to 4094
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

How to Configure VLAN Trunks

To avoid trunking misconfigurations, 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.

Configuring an Ethernet Interface as a Trunk Port

Configuring 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.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport mode** {**dynamic** {**auto** | **desirable**} | **trunk**}
5. **switchport access vlan** *vlan-id*
6. **switchport trunk native vlan** *vlan-id*
7. **end**
8. **show interfaces** *interface-id* **switchport**
9. **show interfaces** *interface-id* **trunk**
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/2**	Specifies the port to be configured for trunking, and enters interface configuration mode.
Step 4	**switchport mode** {**dynamic** {**auto** | **desirable**} | **trunk**} **Example:** Switch(config-if)# **switchport mode dynamic desirable**	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.
Step 5	**switchport access vlan** *vlan-id* **Example:** Switch(config-if)# **switchport access vlan 200**	(Optional) Specifies the default VLAN, which is used if the interface stops trunking.
Step 6	**switchport trunk native vlan** *vlan-id* **Example:** Switch(config-if)# **switchport trunk native vlan 200**	Specifies the native VLAN for IEEE 802.1Q trunks.
Step 7	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 8	**show interfaces** *interface-id* **switchport** **Example:** Switch# **show interfaces gigabitethernet1/0/2 switchport**	Displays the switch port configuration of the interface in the *Administrative Mode* and the *Administrative Trunking Encapsulation* fields of the display.

	Command or Action	Purpose
Step 9	**show interfaces** *interface-id* **trunk** **Example:** Switch# **show interfaces gigabitethernet1/0/2 trunk**	Displays the trunk configuration of the interface.
Step 10	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file. **Note** To return an interface to its default configuration, use the **default interface** *interface-id* interface configuration command. To reset all trunking characteristics of a trunking interface to the defaults, use the **no switchport trunk** interface configuration command. To disable trunking, use the **switchport mode access** interface configuration command to configure the port as a static-access port.

Defining the Allowed VLANs on a Trunk

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.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode trunk**
4. **switchport trunk allowed vlan** {**add** | **all** | **except** | **none** | **remove**} *vlan-list*
5. **end**
6. **show interfaces** *interface-id* **switchport**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	Specifies the port to be configured, and enters interface configuration mode.
Step 3	**switchport mode trunk** **Example:** Switch(config-if)# **switchport mode trunk**	Configures the interface as a VLAN trunk port.
Step 4	**switchport trunk allowed vlan** {**add**	**all**	**except**	**none**	**remove**} *vlan-list* **Example:** Switch(config-if)# **switchport trunk allowed vlan remove 2**	(Optional) Configures the list of VLANs allowed on the trunk. The *vlan-list* parameter is either a single VLAN number from 1 to 4094 or a range of VLANs described by two VLAN numbers, the lower one first, separated by a hyphen. Do not enter any spaces between comma-separated VLAN parameters or in hyphen-specified ranges. All VLANs are allowed by default.
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**show interfaces** *interface-id* **switchport** **Example:** Switch# **show interfaces gigabitethernet1/0/1 switchport**	Verifies your entries in the *Trunking VLANs Enabled* field of the display.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file. **Note** To return to the default allowed VLAN list of all VLANs, use the **no switchport trunk allowed vlan** interface configuration command.

Changing the Pruning-Eligible List

The pruning-eligible list applies only to trunk ports. Each trunk port has its own eligibility list. VTP pruning must be enabled for this procedure to take effect.

SUMMARY STEPS

    1. enable

2. **configure terminal**
3. **interface** *interface-id*
4. **switchport trunk pruning vlan** {**add** | **except** | **none** | **remove**} *vlan-list* [,*vlan* [,*vlan* [,,,]]
5. **end**
6. **show interfaces** *interface-id* **switchport**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet2/0/1**	Selects the trunk port for which VLANs should be pruned, and enters interface configuration mode.
Step 4	**switchport trunk pruning vlan** {**add**	**except**	**none**	**remove**} *vlan-list* [,*vlan* [,*vlan* [,,,]]	Configures the list of VLANs allowed to be pruned from the trunk. For explanations about using the **add**, **except**, **none**, and **remove** keywords, see the command reference for this release. Separate non-consecutive VLAN IDs with a comma and no spaces; use a hyphen to designate a range of IDs. Valid IDs are 2 to 1001. Extended-range VLANs (VLAN IDs 1006 to 4094) cannot be pruned. VLANs that are pruning-ineligible receive flooded traffic. The default list of VLANs allowed to be pruned contains VLANs 2 to 1001. **Note** To return to the default pruning-eligible list of all VLANs, use the **no switchport trunk pruning vlan** interface configuration command.
Step 5	**end** **Example:**	Returns to privileged EXEC mode.

	Command or Action	**Purpose**
	Switch(config)# **end**	
Step 6	**show interfaces** *interface-id* **switchport**	Verifies your entries in the *Pruning VLANs Enabled* field of the display.
	Example:	
	Switch# **show interfaces gigabitethernet2/0/1 switchport**	
Step 7	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.
	Example:	
	Switch# **copy running-config startup-config**	

Configuring the 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.

The native VLAN can be assigned any VLAN ID.

If a packet has a VLAN ID that is the same as the outgoing port native VLAN ID, the packet is sent untagged; otherwise, the switch sends the packet with a tag.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport trunk native vlan** *vlan-id*
5. **end**
6. **show interfaces** *interface-id* **switchport**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable**	Enables privileged EXEC mode.
	Example:	• Enter your password if prompted.
	Switch> **enable**	
Step 2	**configure terminal**	Enters global configuration mode.
	Example:	

	Command or Action	Purpose
	Switch# **configure terminal**	
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/2**	Defines the interface that is configured as the IEEE 802.1Q trunk, and enters interface configuration mode.
Step 4	**switchport trunk native vlan** *vlan-id* **Example:** Switch(config-if)# **switchport trunk native vlan 12**	Configures the VLAN that is sending and receiving untagged traffic on the trunk port. For *vlan-id*, the range is 1 to 4094. **Note** To return to the default native VLAN, VLAN 1, use the **no switchport trunk native vlan** interface configuration command.
Step 5	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 6	**show interfaces** *interface-id* **switchport** **Example:** Switch# **show interfaces gigabitethernet1/0/2 switchport**	Verifies your entries in the *Trunking Native Mode VLAN* field.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Trunk Ports for Load Sharing

Configuring Load Sharing Using STP Port Priorities

If your switch is a member of a switch stack, you must use the **spanning-tree** [**vlan** *vlan-id*] **cost** *cost* interface configuration command instead of the **spanning-tree** [**vlan** *vlan-id*] **port-priority** *priority* interface configuration command to select an interface to put in the forwarding state. Assign lower cost values to interfaces that you want selected first and higher cost values that you want selected last.

These steps describe how to configure a network with load sharing using STP port priorities.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vtp domain** *domain-name*
4. **vtp mode server**
5. **end**
6. **show vtp status**
7. **show vlan**
8. **configure terminal**
9. **interface** *interface-id*
10. **switchport mode trunk**
11. **end**
12. **show interfaces** *interface-id* **switchport**
13. Repeat the above steps on Switch A for a second port in the switch or switch stack.
14. Repeat the above steps on Switch B to configure the trunk ports that connect to the trunk ports configured on Switch A.
15. **show vlan**
16. **configure terminal**
17. **interface** *interface-id*
18. **spanning-tree vlan** *vlan-range* **port-priority** *priority-value*
19. **exit**
20. **interface** *interface-id*
21. **spanning-tree vlan** *vlan-range* **port-priority** *priority-value*
22. **end**
23. **show running-config**
24. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable**	Enables privileged EXEC mode.
	Example:	• Enter your password if prompted.
	Switch> **enable**	
Step 2	**configure terminal**	Enters global configuration mode on Switch A.
	Example:	
	Switch# **configure terminal**	
Step 3	**vtp domain** *domain-name*	Configures a VTP administrative domain.
	Example:	The domain name can be 1 to 32 characters.

	Command or Action	Purpose
	Switch(config)# **vtp domain workdomain**	
Step 4	**vtp mode server** **Example:** Switch(config)# **vtp mode server**	Configures Switch A as the VTP server.
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**show vtp status** **Example:** Switch# **show vtp status**	Verifies the VTP configuration on both Switch A and Switch B. In the display, check the *VTP Operating Mode* and the *VTP Domain Name* fields.
Step 7	**show vlan** **Example:** Switch# **show vlan**	Verifies that the VLANs exist in the database on Switch A.
Step 8	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 9	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	Defines the interface to be configured as a trunk, and enters interface configuration mode.
Step 10	**switchport mode trunk** **Example:** Switch(config-if)# **switchport mode trunk**	Configures the port as a trunk port.
Step 11	**end** **Example:**	Returns to privileged EXEC mode.

	Command or Action	**Purpose**
	`Switch(config-if)# end`	

Step 12	**show interfaces** *interface-id* **switchport**
	Example:
	
	`Switch# show interfaces gigabitethernet1/0/1`
	`switchport`
		Verifies the VLAN configuration.

Step 13	Repeat the above steps on Switch A for a second port in the switch or switch stack.	
Step 14	Repeat the above steps on Switch B to configure the trunk ports that connect to the trunk ports configured on Switch A.	
Step 15	**show vlan** **Example:** `Switch# show vlan`	When the trunk links come up, VTP passes the VTP and VLAN information to Switch B. This command verifies that Switch B has learned the VLAN configuration.
Step 16	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode on Switch A.
Step 17	**interface** *interface-id* **Example:** `Switch(config)# interface gigabitethernet1/0/1`	Defines the interface to set the STP port priority, and enters interface configuration mode.

Step 18	**spanning-tree vlan** *vlan-range* **port-priority** *priority-value*
	Example:
	
	`Switch(config-if)# spanning-tree vlan 8-10`
	`port-priority 16`
		Assigns the port priority for the VLAN range specified. Enter a port priority value from 0 to 240. Port priority values increment by 16.

Step 19	**exit** **Example:** `Switch(config-if)# exit`	Returns to global configuration mode.
Step 20	**interface** *interface-id* **Example:**	Defines the interface to set the STP port priority, and enters interface configuration mode.

	Command or Action	**Purpose**
	Switch(config)# **interface gigabitethernet1/0/2**	
Step 21	**spanning-tree vlan** *vlan-range* **port-priority** *priority-value* **Example:** Switch(config-if)# **spanning-tree vlan 3-6 port-priority 16**	Assigns the port priority for the VLAN range specified. Enter a port priority value from 0 to 240. Port priority values increment by 16.
Step 22	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 23	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 24	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Load Sharing Using STP Path Cost

These steps describe how to configure a network with load sharing using STP path costs.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport mode trunk**
5. **exit**
6. Repeat Steps 2 through 4 on a second interface in Switch A or in Switch A stack.
7. **end**
8. **show running-config**
9. **show vlan**
10. **configure terminal**
11. **interface** *interface-id*
12. **spanning-tree vlan** *vlan-range* **cost** *cost-value*

13. **end**
14. Repeat Steps 9 through 13 on the other configured trunk interface on Switch A, and set the spanning-tree path cost to 30 for VLANs 8, 9, and 10.
15. **exit**
16. **show running-config**
17. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode on Switch A.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	Defines the interface to be configured as a trunk, and enters interface configuration mode.
Step 4	**switchport mode trunk** **Example:** Switch(config-if)# **switchport mode trunk**	Configures the port as a trunk port.
Step 5	**exit** **Example:** Switch(config-if)# **exit**	Returns to global configuration mode.
Step 6	Repeat Steps 2 through 4 on a second interface in Switch A or in Switch A stack.	
Step 7	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 8	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries. In the display, make sure that the interfaces are configured as trunk ports.
Step 9	**show vlan** **Example:** Switch# **show vlan**	When the trunk links come up, Switch A receives the VTP information from the other switches. This command verifies that Switch A has learned the VLAN configuration.
Step 10	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 11	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	Defines the interface on which to set the STP cost, and enters interface configuration mode.
Step 12	**spanning-tree vlan** *vlan-range* **cost** *cost-value* **Example:** Switch(config-if)# **spanning-tree vlan 2-4 cost 30**	Sets the spanning-tree path cost to 30 for VLANs 2 through 4.
Step 13	**end** **Example:** Switch(config-if)# **end**	Returns to global configuration mode.
Step 14	Repeat Steps 9 through 13 on the other configured trunk interface on Switch A, and set the spanning-tree path cost to 30 for VLANs 8, 9, and 10.	
Step 15	**exit** **Example:** Switch(config)# **exit**	Returns to privileged EXEC mode.
Step 16	**show running-config** **Example:**	Verifies your entries. In the display, verify that the path costs are set correctly for both trunk interfaces.

	Command or Action	Purpose
	Switch# **show running-config**	
Step 17	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuration Examples for VLAN Trunking

Example: Configuring an IEEE 802.1Q Trunk

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# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# switchport mode dynamic desirable
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# end
```

Example: Removing a VLAN

This example shows how to remove VLAN 2 from the allowed VLAN list on a port:

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# switchport trunk allowed vlan remove 2
Switch(config-if)# end
```

Where to Go Next

After configuring VLAN trunks, you can configure the following:

- VTP

- VLANs

- Private VLANs

- VLAN Membership Policy Server (VMPS)

- Tunneling

- Voice VLANs

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	*Catalyst 2960-XR Switch VLAN Management Command Reference*

Standards and RFCs

Standard/RFC	Title
—	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for VLAN Trunks

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Configuring Private VLANs

Prerequisites for Private VLANs

The following are prerequisites for configuring private VLANs:

- When you configure private VLANs on switches running VTP, the switch must be in VTP transparent mode.

- When configuring private VLANs on the switch, always use the default Switch Database Management (SDM) template to balance system resources between unicast routes and Layer 2 entries. If another SDM template is configured, use the **sdm prefer default** global configuration command to set the default template.

Secondary and Primary VLAN Configuration

Follow these guidelines when configuring private VLANs:

- Private VLANs are supported in transparent mode for VTP 1, 2 and 3. If the switch is running VTP version 1 or 2, you must set VTP to transparent mode. After you configure a private VLAN, you should not change the VTP mode to client or server. VTP version 3 supports private VLANs in all modes.

- With VTP version 1 or 2, after you have configured private VLANs, use the **copy running-config startup config** privileged EXEC command to save the VTP transparent mode configuration and private-VLAN configuration in the switch startup configuration file. Otherwise, if the switch resets, it defaults to VTP server mode, which does not support private VLANs. VTP version 3 does support private VLANs.

- VTP version 1 and 2 do not propagate private-VLAN configuration. You must configure private VLANs on each device where you want private-VLAN ports unless the devices are running VTP version 3, as VTP3 propagate private vlans.

- You cannot configure VLAN 1 or VLANs 1002 to 1005 as primary or secondary VLANs. Extended VLANs (VLAN IDs 1006 to 4094) can belong to private VLANs.

- A primary VLAN can have one isolated VLAN and multiple community VLANs associated with it. An isolated or community VLAN can have only one primary VLAN associated with it.

- Although a private VLAN contains more than one VLAN, only one Spanning Tree Protocol (STP) instance runs for the entire private VLAN. When a secondary VLAN is associated with the primary VLAN, the STP parameters of the primary VLAN are propagated to the secondary VLAN.

- When copying a PVLAN configuration from a tftp server and applying it on a running-config, the PVLAN association will not be formed. You will need to check and ensure that the primary VLAN is associated to all the secondary VLANs.

  You can also use **configure replace flash:config_file force** instead of **copy flash:config_file running-config**.

- You can enable DHCP snooping on private VLANs. When you enable DHCP snooping on the primary VLAN, it is propagated to the secondary VLANs. If you configure DHCP on a secondary VLAN, the configuration does not take effect if the primary VLAN is already configured.

- When you enable IP source guard on private-VLAN ports, you must enable DHCP snooping on the primary VLAN.

- We recommend that you prune the private VLANs from the trunks on devices that carry no traffic in the private VLANs.

- You can apply different quality of service (QoS) configurations to primary, isolated, and community VLANs.

- Note the following considerations for sticky ARP:

  - Sticky ARP entries are those learned on SVIs and Layer 3 interfaces. These entries do not age out.

  - The **ip sticky-arp** global configuration command is supported only on SVIs belonging to private VLANs.

  - The **ip sticky-arp** interface configuration command is only supported on:

    - Layer 3 interfaces

    - SVIs belonging to normal VLANs

    - SVIs belonging to private VLANs

  For more information about using the **ip sticky-arp** *global* configuration and the **ip sticky-arp interface** configuration commands, see the command reference for this release.

- 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.

- PVLANs are bidirectional. They can be applied at both the ingress and egress sides.

When a frame inLayer-2 is forwarded within a private VLAN, the 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. Similarly, when the frame is routed from an external port to a Private VLAN, the private-VLAN is applied at the egress side.

Bridging

- For upstream traffic from secondary VLAN to primary VLAN, the MAP of the secondary VLAN is applied on the ingress side and the MAP of the primary VLAN is applied on the egress side.

- For downstream traffic from primary VLAN to secondary VLAN, the MAP of the primary VLAN is applied in the ingress direction and the MAP of the secondary VLAN is applied in the egress direction.

Routing

If we have two private VLAN domains - PV1 (sec1, prim1) and PV2 (sec2, prim2). For frames routed from PV1 to PV2:

- The MAP of sec1 and L3 ACL of prim1 is applied in the ingress port .

- The MAP of sec1 and L3 ACL of prim2 is applied in the egress port.

- For packets going upstream or downstream from isolated host port to promiscuous port, the isolated VLAN's VACL is applied in the ingress direction and primary VLAN'S VACL is applied in the egress direction. This allows user to configure different VACL for different secondary VLAN in a same primary VLAN domain.

To filter out specific IP traffic for a private VLAN, you should apply the VLAN map to both the primary and secondary VLANs.

- You can apply router ACLs only on the primary-VLAN SVIs. The ACL is applied to both primary and secondary VLAN Layer 3 traffic.

- Although private VLANs provide host isolation at Layer 2, hosts can communicate with each other at Layer 3.

- Private VLANs support these Switched Port Analyzer (SPAN) features:

  - You can configure a private-VLAN port as a SPAN source port.

  - You can use VLAN-based SPAN (VSPAN) on primary, isolated, and community VLANs or use SPAN on only one VLAN to separately monitor egress or ingress traffic.

Private VLAN Port Configuration

Follow these guidelines when configuring private VLAN ports:

- Use only the private VLAN configuration commands to assign ports to primary, isolated, or community VLANs. Layer 2 access ports assigned to the VLANs that you configure as primary, isolated, or community VLANs are inactive while the VLAN is part of the private VLAN configuration. Layer 2 trunk interfaces remain in the STP forwarding state.

- Do not configure ports that belong to a PAgP or LACP EtherChannel as private VLAN ports. While a port is part of the private VLAN configuration, any EtherChannel configuration for it is inactive.

- Enable Port Fast and BPDU guard on isolated and community host ports to prevent STP loops due to misconfigurations and to speed up STP convergence. When enabled, STP applies the BPDU guard feature to all Port Fast-configured Layer 2 LAN ports. Do not enable Port Fast and BPDU guard on promiscuous ports.

- If you delete a VLAN used in the private VLAN configuration, the private VLAN ports associated with the VLAN become inactive.

- Private VLAN ports can be on different network devices if the devices are trunk-connected and the primary and secondary VLANs have not been removed from the trunk.

Restrictions for Private VLANs

The following are restrictions for configuring private VLANs:

- Private VLANs are only supported on switches running the IP Lite image.

Limitations with Other Features

When configuring private VLANs, remember these limitations with other features:

> **Note** In some cases, the configuration is accepted with no error messages, but the commands have no effect.

- Do not configure fallback bridging on switches with private VLANs.

- When IGMP snooping is enabled on the switch (the default), the switch or switch stack supports no more than 20 private VLAN domains.

- Do not configure a remote SPAN (RSPAN) VLAN as a private VLAN primary or secondary VLAN.

- Do not configure private VLAN ports on interfaces configured for these other features:

  - Dynamic-access port VLAN membership

  - Dynamic Trunking Protocol (DTP)

  - Port Aggregation Protocol (PAgP)

  - Link Aggregation Control Protocol (LACP)

  - Multicast VLAN Registration (MVR)

  - Voice VLAN

  - Web Cache Communication Protocol (WCCP)

- You can configure IEEE 802.1x port-based authentication on a private VLAN port, but do not configure 802.1x with port security, voice VLAN, or per-user ACL on private VLAN ports.

- A private VLAN host or promiscuous port cannot be a SPAN destination port. If you configure a SPAN destination port as a private VLAN port, the port becomes inactive.

- If you configure a static MAC address on a promiscuous port in the primary VLAN, you must add the same static address to all associated secondary VLANs. If you configure a static MAC address on a host port in a secondary VLAN, you must add the same static MAC address to the associated primary VLAN. When you delete a static MAC address from a private VLAN port, you must remove all instances of the configured MAC address from the private VLAN.

Note 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 secondary VLAN is replicated in the primary VLAN. When the original dynamic MAC address is deleted or aged out, the replicated addresses are removed from the MAC address table.

- Configure Layer 3 VLAN interfaces (SVIs) only for primary VLANs.

Information About Private VLANs

Private VLAN Domains

The private VLAN feature addresses two problems that service providers face when using VLANs:

- To enable IP routing, each VLAN is assigned a subnet address space or a block of addresses, which can result in wasting the unused IP addresses, and cause IP address management problems.

Figure 103: Private VLAN Domain

Using private VLANs addresses the scalability problem and provides IP address management benefits for service providers and Layer 2 security for customers. Private VLANs partition a regular VLAN domain into subdomains. A subdomain is represented by a pair of VLANs: a *primary* VLAN and a *secondary* VLAN. A private VLAN can have multiple VLAN pairs, one pair for each subdomain. All VLAN pairs in a private VLAN share the same primary VLAN. The secondary VLAN ID differentiates one subdomain from another.
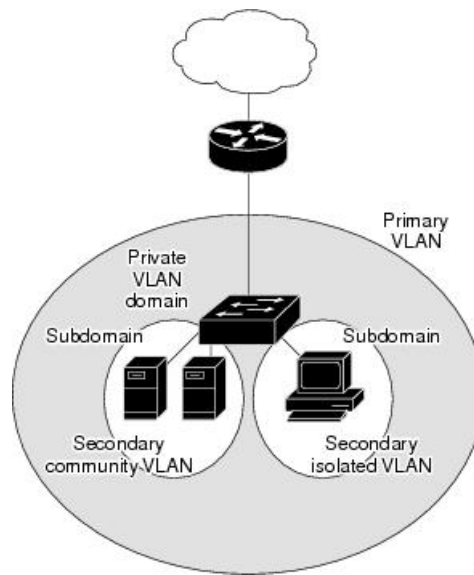
Secondary VLANs

There are two types of secondary VLANs:

• Isolated VLANs—Ports within an isolated VLAN cannot communicate with each other at the Layer 2 level.

• Community VLANs—Ports within a community VLAN can communicate with each other but cannot communicate with ports in other communities at the Layer 2 level.

Private VLANs Ports

Private VLANs provide Layer 2 isolation between ports within the same private VLAN. Private VLAN ports are access ports that are one of these types:

• Promiscuous—A promiscuous port belongs to the primary VLAN and can communicate with all interfaces, including the community and isolated host ports that belong to the secondary VLANs associated with the primary VLAN.

• Isolated—An isolated port is a host port that belongs to an isolated secondary VLAN. It has complete Layer 2 separation from other ports within the same private VLAN, except for the promiscuous ports. Private VLANs block all traffic to isolated ports except traffic from promiscuous ports. Traffic received from an isolated port is forwarded only to promiscuous ports.

• Community—A community port is a host port that belongs to a community secondary VLAN. Community ports communicate with other ports in the same community VLAN and with promiscuous ports. These interfaces are isolated at Layer 2 from all other interfaces in other communities and from isolated ports within their private VLAN.

✎

Note   Trunk ports carry traffic from regular VLANs and also from primary, isolated, and community VLANs.

Primary and secondary VLANs have these characteristics:

- Primary VLAN—A private VLAN has only one primary VLAN. Every port in a private VLAN is a member of the primary VLAN. The primary VLAN carries unidirectional traffic downstream from the promiscuous ports to the (isolated and community) host ports and to other promiscuous ports.

- Isolated VLAN —A private VLAN has only one isolated VLAN. An isolated VLAN is a secondary VLAN that carries unidirectional traffic upstream from the hosts toward the promiscuous ports and the gateway.

- Community VLAN—A community VLAN is a secondary VLAN that carries upstream traffic from the community ports to the promiscuous port gateways and to other host ports in the same community. You can configure multiple community VLANs in a private VLAN.

A promiscuous port can serve only one primary VLAN, one isolated VLAN, and multiple community VLANs. Layer 3 gateways are typically connected to the switch through a promiscuous port. With a promiscuous port, you can connect a wide range of devices as access points to a private VLAN. For example, you can use a promiscuous port to monitor or back up all the private VLAN servers from an administration workstation.

Private VLANs in Networks

In a switched environment, you can assign an individual private VLAN and associated IP subnet to each individual or common group of end stations. The end stations need to communicate only with a default gateway to communicate outside the private VLAN.

You can use private VLANs to control access to end stations in these ways:

- Configure selected interfaces connected to end stations as isolated ports to prevent any communication at Layer 2. For example, if the end stations are servers, this configuration prevents Layer 2 communication between the servers.

- Configure interfaces connected to default gateways and selected end stations (for example, backup servers) as promiscuous ports to allow all end stations access to a default gateway.

You can extend private VLANs across multiple devices by trunking the primary, isolated, and community VLANs to other devices that support private VLANs. To maintain the security of your private VLAN configuration and to avoid other use of the VLANs configured as private VLANs, configure private VLANs on all intermediate devices, including devices that have no private VLAN ports.

IP Addressing Scheme with Private VLANs

Assigning a separate VLAN to each customer creates an inefficient IP addressing scheme:

- Assigning a block of addresses to a customer VLAN can result in unused IP addresses.

- If the number of devices in the VLAN increases, the number of assigned address might not be large enough to accommodate them.

These problems are reduced by using private VLANs, where all members in the private VLAN share a common address space, which is allocated to the primary VLAN. Hosts are connected to secondary VLANs, and the DHCP server assigns them IP addresses from the block of addresses allocated to the primary VLAN. Subsequent IP addresses can be assigned to customer devices in different secondary VLANs, but in the same primary VLAN. When new devices are added, the DHCP server assigns them the next available address from a large pool of subnet addresses.

Private VLANs Across Multiple Switches

Figure 104: Private VLANs Across Switches

As with regular VLANs, private VLANs can span multiple switches. A trunk port carries the primary VLAN and secondary VLANs to a neighboring switch. The trunk port treats the private VLAN as any other VLAN. A feature of private VLANs across multiple switches is that traffic from an isolated port in Switch A does not reach an isolated port on Switch B.
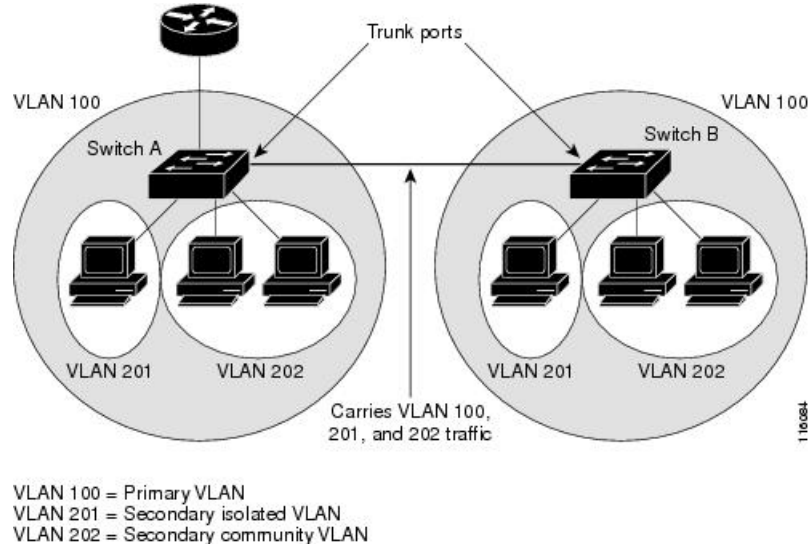
Private VLANs are supported in transparent mode for VTP 1, 2 and 3. Private vlan is also supported on server mode for VTP 3. If we have a server client setup using VTP 3, private vlans configured on the server should be reflected on the client.

Private VLAN Interaction with Other Features

Private VLANs and Unicast, Broadcast, and Multicast Traffic

In regular VLANs, devices in the same VLAN can communicate with each other at the Layer 2 level, but devices connected to interfaces in different VLANs must communicate at the Layer 3 level. In private VLANs, the promiscuous ports are members of the primary VLAN, while the host ports belong to secondary VLANs. Because the secondary VLAN is associated to the primary VLAN, members of the these VLANs can communicate with each other at the Layer 2 level.

In a regular VLAN, broadcasts are forwarded to all ports in that VLAN. Private VLAN broadcast forwarding depends on the port sending the broadcast:

- An isolated port sends a broadcast only to the promiscuous ports or trunk ports.

- A community port sends a broadcast to all promiscuous ports, trunk ports, and ports in the same community VLAN.

- A promiscuous port sends a broadcast to all ports in the private VLAN (other promiscuous ports, trunk ports, isolated ports, and community ports).

Multicast traffic is routed or bridged across private VLAN boundaries and within a single community VLAN. Multicast traffic is not forwarded between ports in the same isolated VLAN or between ports in different secondary VLANs.

Private VLAN multicast forwarding supports the following:

- Sender can be outside the VLAN and the Receivers can be inside the VLAN domain.

- Sender can be inside the VLAN and the Receivers can be outside the VLAN domain.

- Sender and Receiver can both be in the same community vlan.

Private VLANs and SVIs

In a Layer 3 switch, a switch virtual interface (SVI) represents the Layer 3 interface of a VLAN. Layer 3 devices communicate with a private VLAN only through the primary VLAN and not through secondary VLANs. Configure Layer 3 VLAN interfaces (SVIs) only for primary VLANs. You cannot configure Layer 3 VLAN interfaces for secondary VLANs. SVIs for secondary VLANs are inactive while the VLAN is configured as a secondary VLAN.

- If you try to configure a VLAN with an active SVI as a secondary VLAN, the configuration is not allowed until you disable the SVI.

- If you try to create an SVI on a VLAN that is configured as a secondary VLAN and the secondary VLAN is already mapped at Layer 3, the SVI is not created, and an error is returned. If the SVI is not mapped at Layer 3, the SVI is created, but it is automatically shut down.

When the primary VLAN is associated with and mapped to the secondary VLAN, any configuration on the primary VLAN is propagated to the secondary VLAN SVIs. For example, if you assign an IP subnet to the primary VLAN SVI, this subnet is the IP subnet address of the entire private VLAN.

Private VLANs and Switch Stacks

Private VLANs can operate within the switch stack, and private-VLAN ports can reside on different stack members. However, the following changes to the stack can impact private-VLAN operation:

- If a stack contains only one private-VLAN promiscuous port and the stack member that contains that port is removed from the stack, host ports in that private VLAN lose connectivity outside the private VLAN.

- If a stack's active switch that contains the only private-VLAN promiscuous port in the stack fails or leaves the stack and a new active switch is elected, host ports in a private VLAN that had its promiscuous port on the old active switch lose connectivity outside of the private VLAN.

- If two stacks merge, private VLANs on the winning stack are not affected, but private-VLAN configuration on the losing switch is lost when that switch reboots.

Private VLAN Configuration Tasks

To configure a private VLAN, perform these steps:

1. Set VTP mode to transparent.

2. Create the primary and secondary VLANs and associate them.

✎

| **Note** | If the VLAN is not created already, the private VLAN configuration process creates it. |

3. Configure interfaces to be isolated or community host ports, and assign VLAN membership to the host port.

4. Configure interfaces as promiscuous ports, and map the promiscuous ports to the primary-secondary VLAN pair.

5. If inter-VLAN routing will be used, configure the primary SVI, and map the secondary VLANs to the primary.

6. Verify the private VLAN configuration.

Default Private VLAN Configuration

No private VLANs are configured.

How to Configure Private VLANs

Configuring and Associating VLANs in a Private VLAN

The **private-vlan** commands do not take effect until you exit VLAN configuration mode.

To configure and associate VLANs in a Private VLAN, perform these steps:

SUMMARY STEPS

    1.   **enable**
    2.   **configure terminal**
    3.   **vtp mode transparent**
    4.   **vlan** *vlan-id*
    5.   **private-vlan primary**
    6.   **exit**
    7.   **vlan** *vlan-id*
    8.   **private-vlan isolated**
    9.   **exit**
  10.   **vlan** *vlan-id*
  11.   **private-vlan community**
  12.   **exit**
  13.   **vlan** *vlan-id*
  14.   **private-vlan community**
  15.   **exit**
  16.   **vlan** *vlan-id*
  17.   **private-vlan association** [**add** | **remove**] *secondary_vlan_list*

18. **end**

19. **show vlan private-vlan** [**type**] or **show interfaces status**

20. **copy running-config startup config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** Example: Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 3	**vtp mode transparent** Example: Switch(config)# **vtp mode transparent**	Sets VTP mode to transparent (disable VTP). **Note** For VTP3, you can set mode to either server or transparent mode
Step 4	**vlan** *vlan-id* Example: Switch(config)# **vlan 20**	Enters VLAN configuration mode and designates or creates a VLAN that will be the primary VLAN. The VLAN ID range is 2 to 1001 and 1006 to 4094.
Step 5	**private-vlan primary** Example: Switch(config-vlan)# **private-vlan primary**	Designates the VLAN as the primary VLAN.
Step 6	**exit** Example: Switch(config-vlan)# **exit**	Returns to global configuration mode.
Step 7	**vlan** *vlan-id* Example: Switch(config)# **vlan 501**	(Optional) Enters VLAN configuration mode and designates or creates a VLAN that will be an isolated VLAN. The VLAN ID range is 2 to 1001 and 1006 to 4094.

	Command or Action	**Purpose**
Step 8	**private-vlan isolated** **Example:** `Switch(config-vlan)# `**`private-vlan isolated`**	Designates the VLAN as an isolated VLAN.
Step 9	**exit** **Example:** `Switch(config-vlan)# `**`exit`**	Returns to global configuration mode.
Step 10	**vlan** *vlan-id* **Example:** `Switch(config)# `**`vlan 502`**	(Optional) Enters VLAN configuration mode and designates or creates a VLAN that will be a community VLAN. The VLAN ID range is 2 to 1001 and 1006 to 4094.
Step 11	**private-vlan community** **Example:** `Switch(config-vlan)# `**`private-vlan community`**	Designates the VLAN as a community VLAN.
Step 12	**exit** **Example:** `Switch(config-vlan)# `**`exit`**	Returns to global configuration mode.
Step 13	**vlan** *vlan-id* **Example:** `Switch(config)# `**`vlan 503`**	(Optional) Enters VLAN configuration mode and designates or creates a VLAN that will be a community VLAN. The VLAN ID range is 2 to 1001 and 1006 to 4094.
Step 14	**private-vlan community** **Example:** `Switch(config-vlan)# `**`private-vlan community`**	Designates the VLAN as a community VLAN.
Step 15	**exit** **Example:** `Switch(config-vlan)# `**`exit`**	Returns to global configuration mode.

	Command or Action	Purpose
Step 16	**vlan** *vlan-id* **Example:** Switch(config)# **vlan 20**	Enters VLAN configuration mode for the primary VLAN designated in Step 4.
Step 17	**private-vlan association** [**add** | **remove**] *secondary_vlan_list* **Example:** Switch(config-vlan)# **private-vlan association 501-503**	Associates the secondary VLANs with the primary VLAN. It can be a single private-VLAN ID or a hyphenated range of private-VLAN IDs. • The *secondary_vlan_list* parameter cannot contain spaces. It can contain multiple comma-separated items. Each item can be a single private-VLAN ID or a hyphenated range of private-VLAN IDs. • The *secondary_vlan_list* parameter can contain multiple community VLAN IDs but only one isolated VLAN ID. • Enter a *secondary_vlan_list*, or use the **add** keyword with a *secondary_vlan_list* to associate secondary VLANs with a primary VLAN. • Use the **remove** keyword with a *secondary_vlan_list* to clear the association between secondary VLANs and a primary VLAN. • The command does not take effect until you exit VLAN configuration mode.
Step 18	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 19	**show vlan private-vlan** [**type**] or **show interfaces status** **Example:** Switch# **show vlan private-vlan**	Verifies the configuration.
Step 20	**copy running-config startup config** **Example:** Switch# **copy running-config startup-config**	Saves your entries in the switch startup configuration file.

Configuring a Layer 2 Interface as a Private VLAN Host Port

Follow these steps to configure a Layer 2 interface as a private-VLAN host port and to associate it with primary and secondary VLANs:

Note   Isolated and community VLANs are both secondary VLANs.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport mode private-vlan host**
5. **switchport private-vlan host-association** *primary_vlan_id secondary_vlan_id*
6. **end**
7. **show interfaces** [*interface-id*] **switchport**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/22**	Enters interface configuration mode for the Layer 2 interface to be configured.
Step 4	**switchport mode private-vlan host** **Example:** Switch(config-if)# **switchport mode private-vlan host**	Configures the Layer 2 port as a private-VLAN host port.

	Command or Action	**Purpose**
Step 5	**switchport private-vlan host-association** *primary_vlan_id secondary_vlan_id* **Example:** Switch(config-if)# **switchport private-vlan host-association 20 501**	Associates the Layer 2 port with a private VLAN. **Note** This is a required step to associate the PVLAN to a Layer 2 interface.
Step 6	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 7	**show interfaces** [*interface-id*] **switchport** **Example:** Switch# **show interfaces gigabitethernet1/0/22 switchport**	Verifies the configuration.
Step 8	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring a Layer 2 Interface as a Private VLAN Promiscuous Port

Follow these steps to configure a Layer 2 interface as a private VLAN promiscuous port and map it to primary and secondary VLANs:

![Note icon]

Note   Isolated and community VLANs are both secondary VLANs.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport mode private-vlan promiscuous**
5. **switchport private-vlan mapping** *primary_vlan_id* {**add** | **remove**} *secondary_vlan_list*
6. **end**
7. **show interfaces** [*interface-id*] **switchport**
8. **copy running-config startup config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/2**	Enters interface configuration mode for the Layer 2 interface to be configured.
Step 4	**switchport mode private-vlan promiscuous** **Example:** Switch(config-if)# **switchport mode private-vlan promiscuous**	Configures the Layer 2 port as a private VLAN promiscuous port.
Step 5	**switchport private-vlan mapping** *primary_vlan_id* {**add** | **remove**} *secondary_vlan_list* **Example:** Switch(config-if)# **switchport private-vlan mapping 20 add 501-503**	Maps the private VLAN promiscuous port to a primary VLAN and to selected secondary VLANs. • The *secondary_vlan_list* parameter cannot contain spaces. It can contain multiple comma-separated items. Each item can be a single private VLAN ID or a hyphenated range of private VLAN IDs. • Enter a *secondary_vlan_list*, or use the **add** keyword with a *secondary_vlan_list* to map the secondary VLANs to the private VLAN promiscuous port. • Use the **remove** keyword with a *secondary_vlan_list* to clear the mapping between secondary VLANs and the private VLAN promiscuous port.
Step 6	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

	Command or Action	**Purpose**
Step 7	**show interfaces** [*interface-id*] **switchport** **Example:** Switch# **show interfaces gigabitethernet1/0/2 switchport**	Verifies the configuration.
Step 8	**copy running-config startup config** **Example:** Switch# **copy running-config startup-config**	Saves your entries in the switch startup configuration file.

Mapping Secondary VLANs to a Primary VLAN Layer 3 VLAN Interface

If the private VLAN will be used for inter-VLAN routing, you configure an SVI for the primary VLAN and map secondary VLANs to the SVI.

✎

Note   Isolated and community VLANs are both secondary VLANs.

Follow these steps to map secondary VLANs to the SVI of a primary VLAN to allow Layer 3 switching of private VLAN traffic:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface vlan** *primary_vlan_id*
4. **private-vlan mapping** [**add** | **remove**] *secondary_vlan_list*
5. **end**
6. **show interface private-vlan mapping**
7. **copy running-config startup config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:**	Enters global configuration mode.

	Command or Action	Purpose
	Switch# **configure terminal**	
Step 3	**interface vlan** *primary_vlan_id* **Example:** Switch(config)# **interface vlan 20**	Enters interface configuration mode for the primary VLAN, and configures the VLAN as an SVI. The VLAN ID range is 2 to 1001 and 1006 to 4094.
Step 4	**private-vlan mapping** [**add** | **remove**] *secondary_vlan_list* **Example:** Switch(config-if)# **private-vlan mapping 501-503**	Maps the secondary VLANs to the Layer 3 VLAN interface of a primary VLAN to allow Layer 3 switching of private VLAN ingress traffic. **Note** The **private-vlan mapping** interface configuration command only affects private VLAN traffic that is Layer 3 switched. • The *secondary_vlan_list* parameter cannot contain spaces. It can contain multiple comma-separated items. Each item can be a single private-VLAN ID or a hyphenated range of private-VLAN IDs. • Enter a *secondary_vlan_list*, or use the **add** keyword with a *secondary_vlan_list* to map the secondary VLANs to a primary VLAN. • Use the **remove** keyword with a *secondary_vlan_list* to clear the mapping between secondary VLANs and a primary VLAN.
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**show interface private-vlan mapping** **Example:** Switch# **show interfaces private-vlan mapping**	Verifies the configuration.
Step 7	**copy running-config startup config** **Example:** Switch# **copy running-config startup-config**	Saves your entries in the switch startup configuration file.

Monitoring Private VLANs

The following table displays the commands used to monitor private VLANs.

Table 119: Private VLAN Monitoring Commands

Command	Purpose
show interfaces status	Displays the status of interfaces, including the VLANs to which they belongs.
show vlan private-vlan [**type**]	Displays the private VLAN information for the Switch.
show interface switchport	Displays private VLAN configuration on interfaces.
show interface private-vlan mapping	Displays information about the private VLAN mapping for VLAN SVIs.

Configuration Examples for Private VLANs

Example: Configuring a Primary VLAN, Isolated VLAN, and a Community of VLANs

This example shows how to configure VLAN 20 as a primary VLAN, VLAN 501 as an isolated VLAN, and VLANs 502 and 503 as community VLANs, to associate them in a private VLAN, and to verify the configuration:

```
Switch# configure terminal
Switch(config)# vlan 20
Switch(config-vlan)# private-vlan primary
Switch(config-vlan)# exit
Switch(config)# vlan 501
Switch(config-vlan)# private-vlan isolated
Switch(config-vlan)# exit
Switch(config)# vlan 502
Switch(config-vlan)# private-vlan community
Switch(config-vlan)# exit
Switch(config)# vlan 503
Switch(config-vlan)# private-vlan community
Switch(config-vlan)# exit
Switch(config)# vlan 20
Switch(config-vlan)# private-vlan association 501-503
Switch(config-vlan)# end
Switch(config)# show vlan private vlan

Primary Secondary Type Ports
------- --------- ---------------- ----------------------------------------
20 501 isolated
20 502 community
20 503 community
```

```
                    20 504 non-operational
```

Example: Configuring an Interface as a Host Port

This example shows how to configure an interface as a private VLAN host port, associate it with a private VLAN pair, and verify the configuration:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/22
Switch(config-if)# switchport mode private-vlan host
Switch(config-if)# switchport private-vlan host-association 20 501
Switch(config-if)# end
Switch# show interfaces gigabitethernet1/0/22 switchport
Name: Gi1/0/22
Switchport: Enabled
Administrative Mode: private-vlan host
Operational Mode: private-vlan host
Administrative Trunking Encapsulation: negotiate
Operational Trunking Encapsulation: native
Negotiation of Trunking: Off
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: 20 501
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk private VLANs: none
Operational private-vlan:
20 501

<output truncated>
```

Example: Configuring an Interface as a Private VLAN Promiscuous Port

This example shows how to configure an interface as a private VLAN promiscuous port and map it to a private VLAN. The interface is a member of primary VLAN 20 and secondary VLANs 501 to 503 are mapped to it.

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# switchport mode private-vlan promiscous
Switch(config-if)# switchport private-vlan mapping 20 add 501-503
Switch(config-if)# end
```

Use the **show vlan private-vlan** or the **show interface status** privileged EXEC command to display primary and secondary VLANs and private-VLAN ports on the Switch.

Example: Mapping Secondary VLANs to a Primary VLAN Interface

This example shows how to map the interfaces fo VLANs 501 and 502 to primary VLAN 10, which permits routing of secondary VLAN ingress traffic from private VLANs 501 and 502:

```
Switch# configure terminal
Switch(config)# interface vlan 20
Switch(config-if)# private-vlan mapping 501-503
Switch(config-if)# end
Switch# show interfaces private-vlan mapping
Interface Secondary VLAN Type
--------- -------------- -----------------
vlan20    501            isolated
vlan20    502            community
vlan20    503            community
```

Example: Monitoring Private VLANs

This example shows output from the **show vlan private-vlan** command:

```
Switch# show vlan private-vlan
Primary Secondary Type              Ports
------- --------- ----------------- ------------------------------------------
20      501       isolated          Gi1/0/22, Gi1/0/2
20      502       community         Gi1/0/2
20      503       community         Gi1/0/2
```

Where to Go Next

You can configure the following:

- VTP

- VLANs

- VLAN trunking

- VLAN Membership Policy Server (VMPS)

- Tunneling

- Voice VLANs

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	*Catalyst 2960-XR Switch VLAN Management Command Reference*

Standards and RFCs

Standard/RFC	Title
—	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Private VLANs

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Configuring VMPS

Prerequisites for VMPS

- 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.

Restrictions for VMPS

- 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.

Information About VMPS

Dynamic VLAN Assignments

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 VLAN Membership Policy Server (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 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 an 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.

Related Topics

Configuring Dynamic-Access Ports on VMPS Clients, on page 1213

Example: VMPS Configuration, on page 1219

Dynamic-Access Port VLAN Membership

A dynamic-access port can belong to only one VLAN with an ID from 1 to 4094. 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.

Related Topics

Configuring Dynamic-Access Ports on VMPS Clients, on page 1213

Example: VMPS Configuration, on page 1219

Default VMPS Client Configuration

The following table shows the default VMPS and dynamic-access port configuration on client switches.

Table 120: Default VMPS Client and Dynamic-Access Port Configuration

Feature	Default Setting
VMPS domain server	None
VMPS reconfirm interval	60 minutes
VMPS server retry count	3
Dynamic-access ports	None configured

How to Configure VMPS

Entering the IP Address of the VMPS

Note	If the VMPS is being defined for a cluster of switches, enter the address on the command switch.

Before you begin

You must first enter the IP address of the server to configure the switch as a client.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vmps server** *ipaddress* **primary**
4. **vmps server** *ipaddress*
5. **end**
6. **show vmps**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**vmps server** *ipaddress* **primary** **Example:** Switch(config)# **vmps server 10.1.2.3 primary**	Enters the IP address of the switch acting as the primary VMPS server.
Step 4	**vmps server** *ipaddress* **Example:**	(Optional) Enters the IP address of the switch acting as a secondary VMPS server. You can enter up to three secondary server addresses.

	Command or Action	Purpose
	Switch(config)# **vmps server 10.3.4.5**	
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**show vmps** **Example:** Switch# **show vmps**	Verifies your entries in the *VMPS Domain Server* field of the display.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Dynamic-Access Ports on VMPS Clients

⚠️

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.

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.

Before you begin

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.

✎

Note   To return an interface to its default configuration, use the **default interface** *interface-id* interface configuration command. To return an interface to its default switchport mode (dynamic auto), use the **no switchport mode** interface configuration command. To reset the access mode to the default VLAN for the switch, use the **no switchport access vlan** interface configuration command.

SUMMARY STEPS

   1. **enable**
   2. **configure terminal**
   3. **interface** *interface-id*

4. **switchport mode access**
5. **switchport access vlan dynamic**
6. **end**
7. **show interfaces** *interface-id* **switchport**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet 0/1**	Specifies the switch port that is connected to the end station, and enters interface configuration mode.
Step 4	**switchport mode access** **Example:** Switch(config-if)# **switchport mode access**	Sets the port to access mode.
Step 5	**switchport access vlan dynamic** **Example:** Switch(config-if)# **switchport access vlan dynamic**	Configures the port as eligible for dynamic VLAN membership. The dynamic-access port must be connected to an end station.
Step 6	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 7	**show interfaces** *interface-id* **switchport** **Example:** Switch# **show interfaces gigabitethernet 0/1**	Verifies your entries in the *Operational Mode* field of the display.

	Command or Action	Purpose
	switchport	
Step 8	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.
	Example:	
	Switch# **copy running-config startup-config**	

Reconfirming VLAN Memberships

This task confirms the dynamic-access port VLAN membership assignments that the switch has received from the VMPS.

SUMMARY STEPS

1. **enable**
2. **vmps reconfirm**
3. **show vmps**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode.
	Example:	• Enter your password if prompted.
	Switch> **enable**	
Step 2	**vmps reconfirm**	Reconfirms dynamic-access port VLAN membership.
	Example:	
	Switch# **vmps reconfirm**	
Step 3	**show vmps**	Verifies the dynamic VLAN reconfirmation status.
	Example:	
	Switch# **show vmps**	

Changing the 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.

> **Note** 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 also must first use the **rcommand** privileged EXEC command to log in to the member switch.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vmps reconfirm** *minutes*
4. **end**
5. **show vmps**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**vmps reconfirm** *minutes* **Example:** Switch(config)# **vmps reconfirm 90**	Sets the number of minutes between reconfirmations of the dynamic VLAN membership. The range is 1 to 120. The default is 60 minutes. **Note** To return the switch to its default setting, use the **no vmps reconfirm** global configuration command.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	**show vmps** **Example:** Switch# **show vmps**	Verifies the dynamic VLAN reconfirmation status in the *Reconfirm Interval* field of the display.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Changing the Retry Count

Follow these steps to change the number of times that the switch attempts to contact the VMPS before querying the next server.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vmps retry** *count*
4. **end**
5. **show vmps**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**vmps retry** *count* **Example:** Switch(config)# **vmps retry 5**	Changes the retry count. The retry range is 1 to 10; the default is 3. **Note** To return the switch to its default setting, use the **no vmps retry** global configuration command.

	Command or Action	**Purpose**
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show vmps** **Example:** Switch# **show vmps**	Verifies your entry in the *Server Retry Count* field of the display.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Troubleshooting Dynamic-Access Port VLAN Membership

Problem The VMPS shuts down a dynamic-access port under these conditions:

- **Problem** 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.

- **Problem** More than 20 active hosts reside on a dynamic-access port.

Solution To reenable a disabled dynamic-access port, enter the **shutdown** interface configuration command followed by the **no shutdown** interface configuration command.

Monitoring the VMPS

You can display information about the VMPS by using the **show vmps** privileged EXEC command. The switch displays this information about the VMPS:

- VMPS VQP Version—The version of VQP used to communicate with the VMPS. The switch queries the VMPS that is using VQP Version 1.

- Reconfirm Interval—The number of minutes the switch waits before reconfirming the VLAN-to-MAC-address assignments.

- Server Retry Count—The number of times VQP resends a query to the VMPS. If no response is received after this many tries, the switch starts to query the secondary VMPS.

- VMPS domain server—The IP address of the configured VLAN membership policy servers. The switch sends queries to the one marked *current*. The one marked *primary* is the primary server.

• VMPS Action—The result of the most recent reconfirmation attempt. A reconfirmation attempt can occur automatically when the reconfirmation interval expires, or you can force it by entering the **vmps reconfirm** privileged EXEC command or its Network Assistant or SNMP equivalent.

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
```

Configuration Example for VMPS

Example: VMPS Configuration

Figure 105: Dynamic Port VLAN Membership Configuration

This network has a VMPS server switch and VMPS client switches with dynamic-access ports with this configuration:

• 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.

The illustration shows a network topology with a Catalyst 6500 series switch A (Primary VMPS Server 1, 172.20.26.150) connected to a Router and TFTP server (172.20.22.7). End station 1 connects via Dynamic-access port to Client switch B (172.20.26.151) with a Trunk port. Other switches include Catalyst 6500 series Secondary VMPS Server 2 (Switch C, 172.20.26.152), Switch D (172.20.26.153), Switch E (172.20.26.154), Switch F (172.20.26.155), Switch G (172.20.26.156), Switch H (172.20.26.157), Client switch I (172.20.26.158) with End station 2 via Dynamic-access port and Trunk port, and Catalyst 6500 series Secondary VMPS Server 3 (Switch J, 172.20.26.159). An Ethernet segment (Trunk link) connects them.

Related Topics

Configuring Dynamic-Access Ports on VMPS Clients, on page 1213
Dynamic VLAN Assignments, on page 1210
Dynamic-Access Port VLAN Membership, on page 1211

Where to Go Next

You can configure the following:

- VTP
- VLANs
- VLAN Trunking

- Private VLANs

- Tunneling

- Voice VLANs

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	*Catalyst 2960-XR Switch VLAN Management Command Reference*

Standards and RFCs

Standard/RFC	Title
—	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for VMPS

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling

Prerequisites for Configuring Tunneling

The following sections list prerequisites and considerations for configuring IEEE 802.1Q and Layer 2 protocol tunneling.

IEEE 802.1Q Tunneling

Although IEEE 802.1Q tunneling works well for Layer 2 packet switching, there are incompatibilities between some Layer 2 features and Layer 3 switching.

- A tunnel port cannot be a routed port.

- IP routing is not supported on a VLAN that includes IEEE 802.1Q tunnel ports. Packets received from a tunnel port are forwarded based only on Layer 2 information. If routing is enabled on a switch virtual interface (SVI) that includes tunnel ports, untagged IP packets received from the tunnel port are recognized and routed by the switch. Customers can access the Internet through its native VLAN. If this access is not needed, you should not configure SVIs on VLANs that include tunnel ports.

- Fallback bridging is not supported on tunnel ports. Because all IEEE 802.1Q-tagged packets received from a tunnel port are treated as non-IP packets, if fallback bridging is enabled on VLANs that have tunnel ports configured, IP packets would be improperly bridged across VLANs. Therefore, you must not enable fallback bridging on VLANs with tunnel ports.

- Tunnel ports do not support IP access control lists (ACLs).

- Layer 3 quality of service (QoS) ACLs and other QoS features related to Layer 3 information are not supported on tunnel ports. MAC-based QoS is supported on tunnel ports.

- EtherChannel port groups are compatible with tunnel ports as long as the IEEE 802.1Q configuration is consistent within an EtherChannel port group.

- Port Aggregation Protocol (PAgP), Link Aggregation Control Protocol (LACP), and UniDirectional Link Detection (UDLD) are supported on IEEE 802.1Q tunnel ports.

- Dynamic Trunking Protocol (DTP) is not compatible with IEEE 802.1Q tunneling because you must manually configure asymmetric links with tunnel ports and trunk ports.

- VLAN Trunking Protocol (VTP) does not work between devices that are connected by an asymmetrical link or devices that communicate through a tunnel.

- Loopback detection is supported on IEEE 802.1Q tunnel ports.

- When a port is configured as an IEEE 802.1Q tunnel port, spanning-tree bridge protocol data unit (BPDU) filtering is automatically enabled on the interface. Cisco Discovery Protocol (CDP) and the Layer Link Discovery Protocol (LLDP) are automatically disabled on the interface.

Layer 2 Protocol Tunneling

- The switch supports tunneling of CDP, STP, including multiple STP (MSTP), and VTP. Protocol tunneling is disabled by default but can be enabled for the individual protocols on IEEE 802.1Q tunnel ports or access ports.

- The switch does not support Layer 2 protocol tunneling on ports with switchport mode dynamic auto or dynamic desirable.

- DTP is not compatible with layer 2 protocol tunneling.

- The edge switches on the outbound side of the service-provider network restore the proper Layer 2 protocol and MAC address information and forward the packets to all tunnel and access ports in the same metro VLAN.

- For interoperability with third-party vendor switches, the switch supports a Layer 2 protocol-tunnel bypass feature. Bypass mode transparently forwards control PDUs to vendor switches that have different ways of controlling protocol tunneling. When Layer 2 protocol tunneling is enabled on ingress ports on a switch, egress trunk ports forward the tunneled packets with a special encapsulation. If you also enable Layer 2 protocol tunneling on the egress trunk port, this behavior is bypassed, and the switch forwards control PDUs without any processing or modification.

- The switch supports PAgP, LACP, and UDLD tunneling for emulated point-to-point network topologies. Protocol tunneling is disabled by default but can be enabled for the individual protocols on IEEE 802.1Q tunnel ports or on access ports.

- If you enable PAgP or LACP tunneling, we recommend that you also enable UDLD on the interface for faster link-failure detection.

- Loopback detection is not supported on Layer 2 protocol tunneling of PAgP, LACP, or UDLD packets.

- EtherChannel port groups are compatible with tunnel ports when the IEEE 802.1Q configuration is consistent within an EtherChannel port group.

- If an encapsulated PDU (with the proprietary destination MAC address) is received from a tunnel port or an access port with Layer 2 tunneling enabled, the tunnel port is shut down to prevent loops. The port also shuts down when a configured shutdown threshold for the protocol is reached. You can manually reenable the port (by entering a **shutdown** and a **no shutdown** command sequence). If errdisable recovery is enabled, the operation is retried after a specified time interval.

- Only decapsulated PDUs are forwarded to the customer network. The spanning-tree instance running on the service-provider network does not forward BPDUs to tunnel ports. CDP packets are not forwarded from tunnel ports.

- When protocol tunneling is enabled on an interface, you can set a per-protocol, per-port, shutdown threshold for the PDUs generated by the customer network. If the limit is exceeded, the port shuts down. You can also limit BPDU rate by using QoS ACLs and policy maps on a tunnel port.

- When protocol tunneling is enabled on an interface, you can set a per-protocol, per-port, drop threshold for the PDUs generated by the customer network. If the limit is exceeded, the port drops PDUs until the rate at which it receives them is below the drop threshold.

- Because tunneled PDUs (especially STP BPDUs) must be delivered to all remote sites so that the customer virtual network operates properly, you can give PDUs higher priority within the service-provider network than data packets received from the same tunnel port. By default, the PDUs use the same CoS value as data packets.

Layer 2 Tunneling for EtherChannels

To configure Layer 2 point-to-point tunneling to facilitate the automatic creation of EtherChannels, you need to configure both the SP (service-provider) edge switch and the customer switch.

Information about Tunneling

IEEE 802.1Q and Layer 2 Protocol Overview

Virtual private networks (VPNs) provide enterprise-scale connectivity on a shared infrastructure, often Ethernet-based, with the same security, prioritization, reliability, and manageability requirements of private networks. Tunneling is a feature designed for service providers who carry traffic of multiple customers across their networks and are required to maintain the VLAN and Layer 2 protocol configurations of each customer without impacting the traffic of other customers.

For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

IEEE 802.1Q Tunneling

Business customers of service providers often have specific requirements for VLAN IDs and the number of VLANs to be supported. The VLAN ranges required by different customers in the same service-provider network might overlap, and traffic of customers through the infrastructure might be mixed. Assigning a unique range of VLAN IDs to each customer would restrict customer configurations and could easily exceed the VLAN limit (4096) of the IEEE 802.1Q specification.

Using the IEEE 802.1Q tunneling feature, service providers can use a single VLAN to support customers who have multiple VLANs. Customer VLAN IDs are preserved, and traffic from different customers is segregated within the service-provider network, even when they appear to be in the same VLAN. Using IEEE 802.1Q tunneling expands VLAN space by using a VLAN-in-VLAN hierarchy and retagging the tagged packets. A port configured to support IEEE 802.1Q tunneling is called a tunnel port. When you configure tunneling, you assign a tunnel port to a VLAN ID that is dedicated to tunneling. Each customer requires a separate service-provider VLAN ID, but that VLAN ID supports all of the customer's VLANs.

Customer traffic tagged in the normal way with appropriate VLAN IDs comes from an IEEE 802.1Q trunk port on the customer device and into a tunnel port on the service-provider edge switch. The link between the customer device and the edge switch is asymmetric because one end is configured as an IEEE 802.1Q trunk port, and the other end is configured as a tunnel port. You assign the tunnel port interface to an access VLAN ID that is unique to each customer.

Figure 106: IEEE 802.1Q Tunnel Ports in a Service-Provider Network
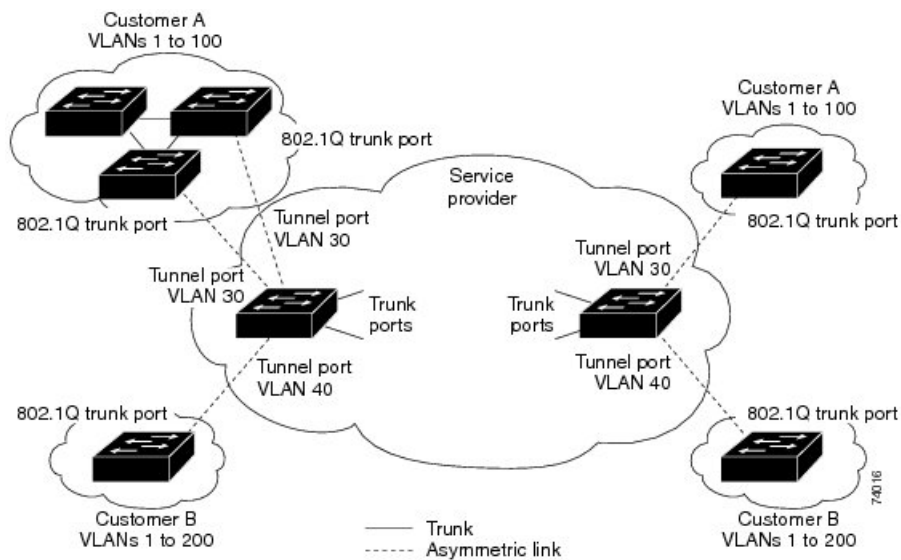
Packets coming from the customer trunk port into the tunnel port on the service-provider edge switch are normally IEEE 802.1Q-tagged with the appropriate VLAN ID. The tagged packets remain intact inside the switch and when they exit the trunk port into the service-provider network, they are encapsulated with another layer of an IEEE 802.1Q tag (called the metro tag) that contains the VLAN ID that is unique to the customer. The original customer IEEE 802.1Q tag is preserved in the encapsulated packet. Therefore, packets entering the service-provider network are double-tagged, with the outer (metro) tag containing the customer's access VLAN ID, and the inner VLAN ID being that of the incoming traffic.

When the double-tagged packet enters another trunk port in a service-provider core switch, the outer tag is stripped as the switch processes the packet. When the packet exits another trunk port on the same core switch, the same metro tag is again added to the packet.

Figure 107: Original (Normal), IEEE 802.1Q, and Double-Tagged Ethernet Packet Formats

This figure shows the tag structures of the double-tagged packets.
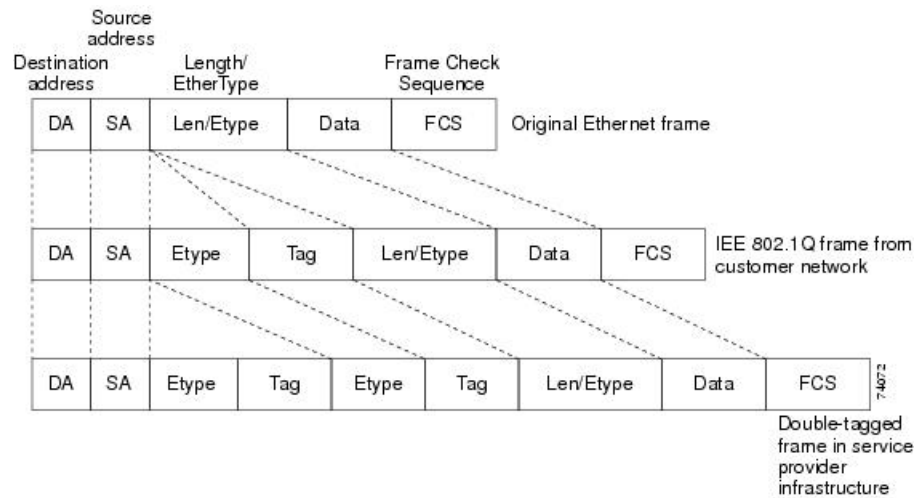
When the packet enters the trunk port of the service-provider egress switch, the outer tag is again stripped as the switch internally processes the packet. However, the metro tag is not added when the packet is sent out the tunnel port on the edge switch into the customer network. The packet is sent as a normal IEEE 802.1Q-tagged frame to preserve the original VLAN numbers in the customer network.

In the above network figure, Customer A was assigned VLAN 30, and Customer B was assigned VLAN 40. Packets entering the edge device tunnel ports with IEEE 802.1Q tags are double-tagged when they enter the service-provider network, with the outer tag containing VLAN ID 30 or 40, appropriately, and the inner tag containing the original VLAN number, for example, VLAN 100. Even if both Customers A and B have VLAN 100 in their networks, the traffic remains segregated within the service-provider network because the outer tag is different. Each customer controls its own VLAN numbering space, which is independent of the VLAN numbering space used by other customers and the VLAN numbering space used by the service-provider network.

At the outbound tunnel port, the original VLAN numbers on the customer's network are recovered. It is possible to have multiple levels of tunneling and tagging, but the device supports only one level in this release.

If traffic coming from a customer network is not tagged (native VLAN frames), these packets are bridged or routed as normal packets. All packets entering the service-provider network through a tunnel port on an edge device are treated as untagged packets, whether they are untagged or already tagged with IEEE 802.1Q headers. The packets are encapsulated with the metro tag VLAN ID (set to the access VLAN of the tunnel port) when they are sent through the service-provider network on an IEEE 802.1Q trunk port. The priority field on the metro tag is set to the interface class of service (CoS) priority configured on the tunnel port. (The default is zero if none is configured.)

On device, because 802.1Q tunneling is configured on a per-port basis, it does not matter whether the device is a standalone device or a stack member. All configuration is done on the active stack.

IEEE 802.1Q Tunneling Configuration Guidelines

When you configure IEEE 802.1Q tunneling, you should always use an asymmetrical link between the customer device and the edge switch, with the customer device port configured as an IEEE 802.1Q trunk port and the edge switch port configured as a tunnel port.

Assign tunnel ports only to VLANs that are used for tunneling.

Configuration requirements for native VLANs and for and maximum transmission units (MTUs) are explained in these next sections.

Native VLANs

When configuring IEEE 802.1Q tunneling on an edge switch, you must use IEEE 802.1Q trunk ports for sending packets into the service-provider network. However, packets going through the core of the service-provider network can be carried through IEEE 802.1Q trunks, ISL trunks, or nontrunking links. When IEEE 802.1Q trunks are used in these core switches, the native VLANs of the IEEE 802.1Q trunks must not match any native VLAN of the nontrunking (tunneling) port on the same switch because traffic on the native VLAN would not be tagged on the IEEE 802.1Q sending trunk port.

In the following network figure, VLAN 40 is configured as the native VLAN for the IEEE 802.1Q trunk port from Customer X at the ingress edge switch in the service-provider network (Switch B). Switch A of Customer X sends a tagged packet on VLAN 30 to the ingress tunnel port of Switch B in the service-provider network, which belongs to access VLAN 40. Because the access VLAN of the tunnel port (VLAN 40) is the same as the native VLAN of the edge switch trunk port (VLAN 40), the metro tag is not added to tagged packets received from the tunnel port. The packet carries only the VLAN 30 tag through the service-provider network to the trunk port of the egress-edgeswitch (Switch C) and is misdirected through the egress switch tunnel port to Customer Y.

Figure 108: Potential Problems with IEEE 802.1Q Tunneling and Native VLANs
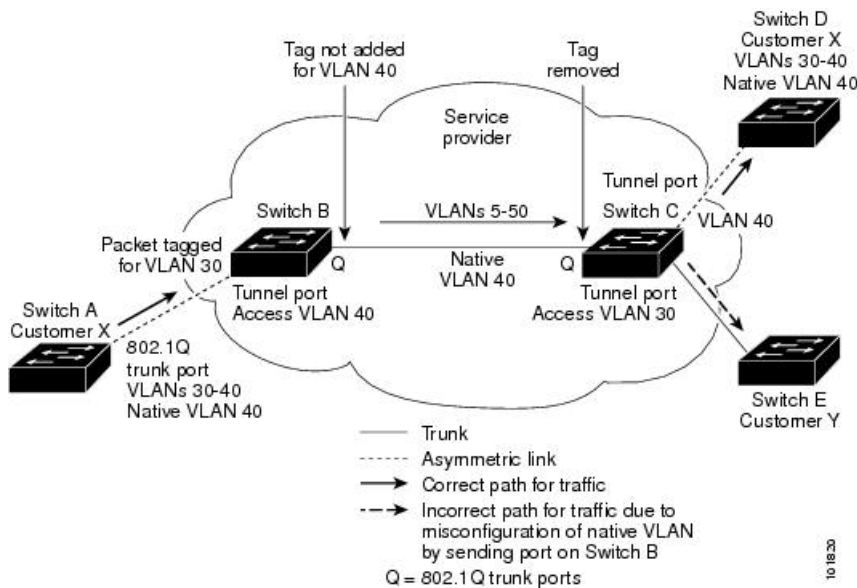
These are some ways to solve this problem:

-
- Use the **vlan dot1q tag native** global configuration command to configure the edge switches so that all packets going out an IEEE 802.1Q trunk, including the native VLAN, are tagged. If the switches is configured to tag native VLAN packets on all IEEE 802.1Q trunks, the switches accepts untagged packets, but sends only tagged packets.

- Ensure that the native VLAN ID on the edge switches trunk port is not within the customer VLAN range. For example, if the trunk port carries traffic of VLANs 100 to 200, assign the native VLAN a number outside that range.

System MTU

The default system MTU for traffic on the switch is 1500 bytes. You can configure Fast Ethernet ports on the switch members in the mixed hardware switch stack to support frames larger than 1500 bytes by using the **system mtu** global configuration command.

You can configure 10-Gigabit and Gigabit Ethernet ports to support frames larger than 1500 bytes by using the **system mtu jumbo** global configuration command.

The system MTU and system jumbo MTU values do not include the IEEE 802.1Q header. Because the IEEE 802.1Q tunneling feature increases the frame size by 4 bytes when the metro tag is added, you must configure all switches in the service-provider network to be able to process maximum frames by adding 4 bytes to the system MTU and system jumbo MTU sizes.

For example, the switch supports a maximum frame size of 1496 bytes with one of these configurations:

- The switch has a system jumbo MTU value of 1500 bytes, and the **switchport mode dot1q tunnel** interface configuration command is configured on a 10-Gigabit or Gigabit Ethernet switch port.

- The switch member has a system MTU value of 1500 bytes, and the **switchport mode dot1q tunnel** interface configuration command is configured on a Fast Ethernet port of the member.

Default IEEE 802.1Q Tunneling Configuration

By default, IEEE 802.1Q tunneling is disabled because the default switchport mode is dynamic auto. Tagging of IEEE 802.1Q native VLAN packets on all IEEE 802.1Q trunk ports is also disabled.

Layer 2 Protocol Tunneling Overview

Customers at different sites connected across a service-provider network need to use various Layer 2 protocols to scale their topologies to include all remote sites, as well as the local sites. STP must run properly, and every VLAN should build a proper spanning tree that includes the local site and all remote sites across the service-provider network. Cisco Discovery Protocol (CDP) must discover neighboring Cisco devices from local and remote sites. VLAN Trunking Protocol (VTP) must provide consistent VLAN configuration throughout all sites in the customer network.

When protocol tunneling is enabled, edge switches on the inbound side of the service-provider network encapsulate Layer 2 protocol packets with a special MAC address and send them across the service-provider network. Core switches in the network do not process these packets but forward them as normal packets. Layer 2 protocol data units (PDUs) for CDP, STP, or VTP cross the service-provider network and are delivered to customer switches on the outbound side of the service-provider network. Identical packets are received by all customer ports on the same VLANs with these results:

- Users on each of a customer's sites can properly run STP, and every VLAN can build a correct spanning tree based on parameters from all sites and not just from the local site.

- CDP discovers and shows information about the other Cisco devices connected through the service-provider network.

• VTP provides consistent VLAN configuration throughout the customer network, propagating to all switches through the service provider.

Note To provide interoperability with third-party vendors, you can use the Layer 2 protocol-tunnel bypass feature. Bypass mode transparently forwards control PDUs to vendor switches that have different ways of controlling protocol tunneling. You implement bypass mode by enabling Layer 2 protocol tunneling on the egress trunk port. When Layer 2 protocol tunneling is enabled on the trunk port, the encapsulated tunnel MAC address is removed and the protocol packets have their normal MAC address.

Layer 2 protocol tunneling can be used independently or can enhance IEEE 802.1Q tunneling. If protocol tunneling is not enabled on IEEE 802.1Q tunneling ports, remoteswitches at the receiving end of the service-provider network do not receive the PDUs and cannot properly run STP, CDP, and VTP. When protocol tunneling is enabled, Layer 2 protocols within each customer's network are totally separate from those running within the service-provider network. Customer switches on different sites that send traffic through the service-provider network with IEEE 802.1Q tunneling achieve complete knowledge of the customer's VLAN. If IEEE 802.1Q tunneling is not used, you can still enable Layer 2 protocol tunneling by connecting to the customer switch through access ports and by enabling tunneling on the service-provider access port.

For example, in the following figure (Layer 2 Protocol Tunneling), Customer X has four switches in the same VLAN, that are connected through the service-provider network. If the network does not tunnel PDUs, switches on the far ends of the network cannot properly run STP, CDP, and VTP. For example, STP for a VLAN on a switch in Customer X, Site 1, will build a spanning tree on theswitches at that site without considering convergence parameters based on Customer X's switch in Site 2. This could result in the topology shown in the Layer 2 Network Topology without Proper Convergence figure.

Figure 109: Layer 2 Protocol Tunneling
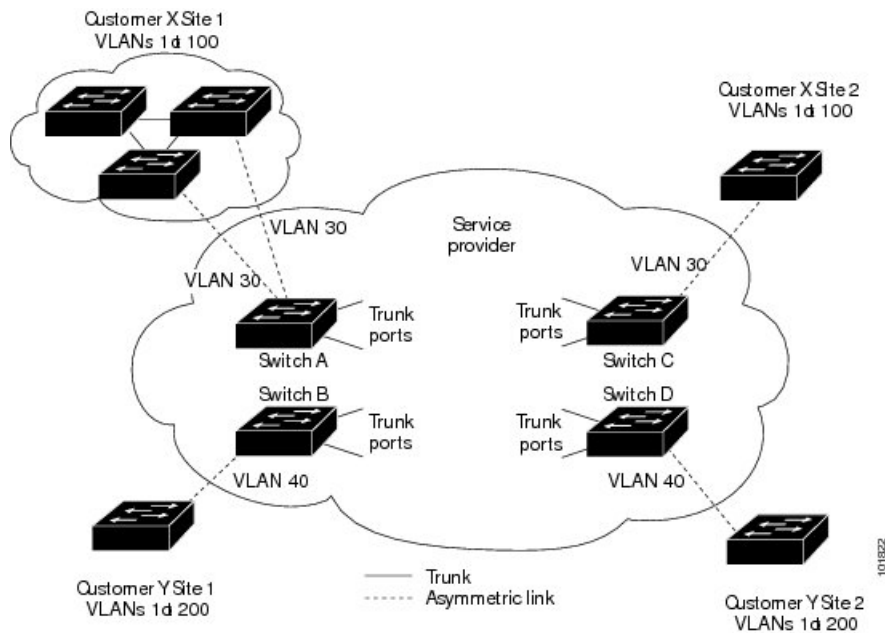
Figure 110: Layer 2 Network Topology Without Proper Convergence
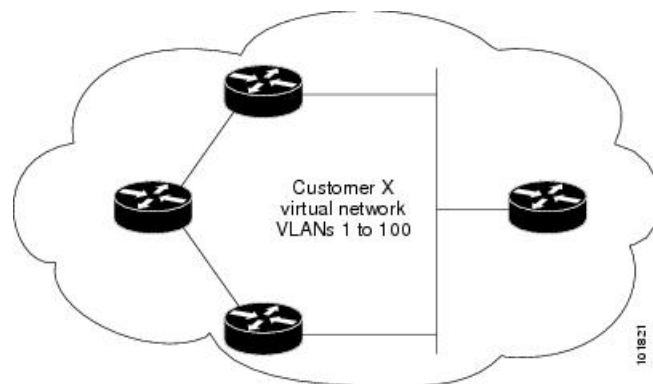
In an SP network, you can use Layer 2 protocol tunneling to enhance the creation of EtherChannels by emulating a point-to-point network topology. When you enable protocol tunneling (PAgP or LACP) on the SP switch, remote customer switches receive the PDUs and can negotiate the automatic creation of EtherChannels.

For example, in the following figure (Layer 2 Protocol Tunneling for EtherChannels), Customer A has two switches in the same VLAN that are connected through the SP network. When the network tunnels PDUs, switches on the far ends of the network can negotiate the automatic creation of EtherChannels without needing dedicated lines.

Figure 111: Layer 2 Protocol Tunneling for EtherChannels
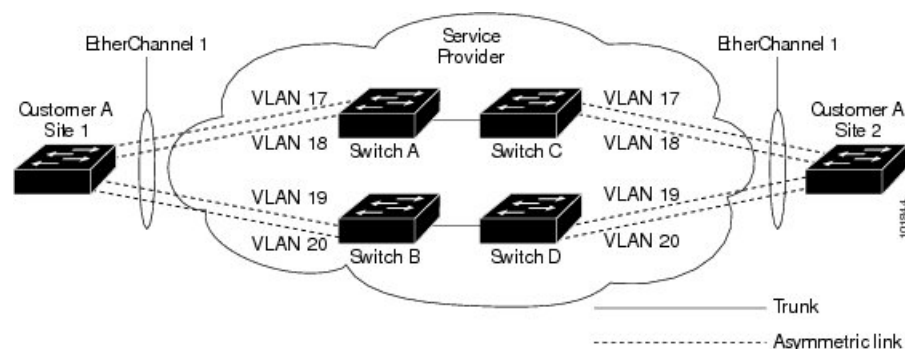
Layer 2 Protocol Tunneling on Ports

You can enable Layer 2 protocol tunneling (by protocol) on the ports that are connected to the customer in the edge switches of the service-provider network. The service-provider edge switches connected to the customer switch perform the tunneling process. Edge switch tunnel ports are connected to customer IEEE 802.1Q trunk ports. Edge switch access ports are connected to customer access ports. The edge switches connected to the customer switch perform the tunneling process.

You can enable Layer 2 protocol tunneling on ports that are configured as access ports or tunnel ports. You cannot enable Layer 2 protocol tunneling on ports configured in either **switchport mode dynamic auto** mode (the default mode) or **switchport mode dynamic desirable** mode.

The switch supports Layer 2 protocol tunneling for CDP, STP, and VTP. For emulated point-to-point network topologies, it also supports PAgP, LACP, and UDLD protocols. The switch does not support Layer 2 protocol tunneling for LLDP.

> ✎
>
> **Note** PAgP, LACP, and UDLD protocol tunneling is only intended to emulate a point-to-point topology. An erroneous configuration that sends tunneled packets to many ports could lead to a network failure.

When the Layer 2 PDUs that entered the service-provider inbound edge switch through a Layer 2 protocol-enabled port exit through the trunk port into the service-provider network, the switch overwrites the customer PDU-destination MAC address with a well-known Cisco proprietary multicast address (01-00-0c-cd-cd-d0). If IEEE 802.1Q tunneling is enabled, packets are also double-tagged; the outer tag is the customer metro tag, and the inner tag is the customer's VLAN tag. The core switches ignore the inner tags and forward the packet to all trunk ports in the same metro VLAN. The edge switches on the outbound side restore the proper Layer 2 protocol and MAC address information and forward the packets to all tunnel or access ports in the same metro VLAN. Therefore, the Layer 2 PDUs remain intact and are delivered across the service-provider infrastructure to the other side of the customer network.

See the Layer 2 Protocol Tunneling figure in , with Customer X and Customer Y in access VLANs 30 and 40, respectively. Asymmetric links connect the customers in Site 1 to edge switches in the service-provider network. The Layer 2 PDUs (for example, BPDUs) coming into Switch B from Customer Y in Site 1 are forwarded to the infrastructure as double-tagged packets with the well-known MAC address as the destination MAC address. These double-tagged packets have the metro VLAN tag of 40, as well as an inner VLAN tag (for example, VLAN 100). When the double-tagged packets enter Switch D, the outer VLAN tag 40 is removed, the well-known MAC address is replaced with the respective Layer 2 protocol MAC address, and the packet is sent to Customer Y on Site 2 as a single-tagged frame in VLAN 100.

You can also enable Layer 2 protocol tunneling on access ports on the edge switch connected to access or trunk ports on the customer switch. In this case, the encapsulation and decapsulation process is the same as described in the previous paragraph, except that the packets are not double-tagged in the service-provider network. The single tag is the customer-specific access VLAN tag.

In switch stacks, Layer 2 protocol tunneling configuration is distributed among all stack members. Each stack member that receives an ingress packet on a local port encapsulates or decapsulates the packet and forwards it to the appropriate destination port. On a single switch, ingress Layer 2 protocol-tunneled traffic is sent across all local ports in the same VLAN on which Layer 2 protocol tunneling is enabled. In a stack, packets received by a Layer 2 protocol-tunneled port are distributed to all ports in the stack that are configured for Layer 2 protocol tunneling and are in the same VLAN. All Layer 2 protocol tunneling configuration is handled by the active stack and distributed to all stack members.

Default Layer 2 Protocol Tunneling Configuration

The following table shows the default Layer 2 protocol tunneling configuration.

Table 121: Default Layer 2 Ethernet Interface VLAN Configuration

Feature	Default Setting
Layer 2 protocol tunneling	Disabled.
Shutdown threshold	None set.
Drop threshold	None set.

Feature	Default Setting
CoS Value	If a CoS value is configured on the interface, that value is used to set the BPDU CoS value for Layer 2 protocol tunneling. If no CoS value is configured at the interface level, the default value for CoS marking of L2 protocol tunneling BPDUs is 5. This does not apply to data traffic.

How to Configure Tunneling

Configuring an IEEE 802.1Q Tunneling Port

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport access vlan** *vlan-id*
5. **switchport mode dot1q-tunnel**
6. **exit**
7. **vlan dot1q tag native**
8. **end**
9. Use one of the following:

    • **show dot1q-tunnel**
    • **show running-config interface**

10. **show vlan dot1q tag native**
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.

	Command or Action	**Purpose**
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet2/0/1**	Enters interface configuration mode for the interface to be configured as a tunnel port. This should be the edge port in the service-provider network that connects to the customer switch. Valid interfaces include physical interfaces and port-channel logical interfaces (port channels 1 to 48).
Step 4	**switchport access vlan** *vlan-id* **Example:** Switch(config-if)# **switchport access vlan 2**	Specifies the default VLAN, which is used if the interface stops trunking. This VLAN ID is specific to the particular customer.
Step 5	**switchport mode dot1q-tunnel** **Example:** Switch(config-if)# **switchport mode dot1q-tunnel**	Sets the interface as an IEEE 802.1Q tunnel port. **Note** Use the **no switchport mode dot1q-tunnel** interface configuration command to return the port to the default state of dynamic desirable.
Step 6	**exit** **Example:** Switch(config-if)# **exit**	Returns to privileged EXEC mode.
Step 7	**vlan dot1q tag native** **Example:** Switch(config)# **vlan dot1q tag native**	(Optional) Sets the switch to enable tagging of native VLAN packets on all IEEE 802.1Q trunk ports. When not set, and a customer VLAN ID is the same as the native VLAN, the trunk port does not apply a metro tag, and packets could be sent to the wrong destination. **Note** Use the **no vlan dot1q tag native** global configuration command to disable tagging of native VLAN packets.
Step 8	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Step 9	Use one of the following:
	
	  • **show dot1q-tunnel**
	  • **show running-config interface**
	
	Example:
	
	Switch# **show dot1q-tunnel**
		Displays the ports configured for IEEE 802.1Q tunneling.
		
		Displays the ports that are in tunnel mode.

	Command or Action	Purpose
	or Switch# **show running-config interface**	
Step 10	**show vlan dot1q tag native** Example: Switch# **show vlan dot1q native**	Displays IEEE 802.1Q native VLAN tagging status.
Step 11	**copy running-config startup-config** Example: Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring Layer 2 Protocol Tunneling

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. Use one of the following:

    - **switchport mode access**
    - **switchport mode dot1q-tunnel**

5. **l2protocol-tunnel** [**cdp** | **lldp** | **point-to-point** | **stp** | **vtp**]
6. **l2protocol-tunnel shutdown-threshold** [*packet_second_rate_value* | **cdp** | **lldp point-to-point** | **stp** | **vtp**]
7. **l2protocol-tunnel drop-threshold** [*packet_second_rate_value* | **cdp** | **lldp** | **point-to-point** | **stp** | **vtp**]
8. **exit**
9. **errdisable recovery cause l2ptguard**
10. **l2protocol-tunnel cos** *value*
11. **end**
12. **show l2protocol**
13. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** Example:	Enables privileged EXEC mode. - Enter your password if prompted.

	Command or Action	Purpose
	Switch> **enable**	
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	Specifies the interface connected to the phone, and enters interface configuration mode.
Step 4	Use one of the following: • **switchport mode access** • **switchport mode dot1q-tunnel** **Example:** Switch# **switchport mode access** or Switch# **switchport mode dot1q-tunnel**	Configures the interface as an access port or an IEEE 802.1Q tunnel port.
Step 5	**l2protocol-tunnel** [**cdp** | **lldp** | **point-to-point** | **stp** | **vtp**] **Example:** Switch# **l2protocol-tunnel cdp**	Enables protocol tunneling for the desired protocol. If no keyword is entered, tunneling is enabled for all three Layer 2 protocols. **Note**　Use the **no l2protocol-tunnel** [**cdp** | **lldp** | **point-to-point** | **stp** | **vtp**] interface configuration command to disable protocol tunneling for one of the Layer 2 protocols or for all three.
Step 6	**l2protocol-tunnel shutdown-threshold** [*packet_second_rate_value* | **cdp** | **lldp point-to-point** | **stp** | **vtp**] **Example:** Switch# **l2protocol-tunnel shutdown-threshold 100 cdp**	(Optional) Configures the threshold for packets-per-second accepted for encapsulation. The interface is disabled if the configured threshold is exceeded. If no protocol option is specified, the threshold applies to each of the tunneled Layer 2 protocol types. The range is 1 to 4096. The default is to have no threshold configured. **Note**　If you also set a drop threshold on this interface, the **shutdown-threshold** value must be greater than or equal to the **drop-threshold** value.

	Command or Action	Purpose
		Note Use the **no l2protocol-tunnel shutdown-threshold** [*packet_second_rate_value* | **cdp** | **lldp** | **point-to-point** | **stp** | **vtp**] and the **no l2protocol-tunnel drop-threshold** [*packet_second_rate_value* | **cdp** | **lldp** | **point-to-point** |**stp** | **vtp**] commands to return the shutdown and drop thresholds to the default settings.
Step 7	**l2protocol-tunnel drop-threshold** [*packet_second_rate_value* | **cdp** | **lldp** | **point-to-point** | **stp** | **vtp**] **Example:** Switch# **l2protocol-tunnel drop-threshold 100 cdp**	(Optional) Configures the threshold for packets-per-second accepted for encapsulation. The interface drops packets if the configured threshold is exceeded. If no protocol option is specified, the threshold applies to each of the tunneled Layer 2 protocol types. The range is 1 to 4096. The default is to have no threshold configured. **Note** If you also set a shutdown threshold on this interface, the **drop-threshold** value must be less than or equal to the **shutdown-threshold** value. **Note** Use the **no l2protocol-tunnel shutdown-threshold** [**cdp** | **lldp** | **point-to-point** | **stp** | **vtp**] and the **no l2protocol-tunnel drop-threshold** [**cdp** | **stp** | **vtp**] commands to return the shutdown and drop thresholds to the default settings.
Step 8	**exit** **Example:** Switch# **exit**	Returns to global configuration mode.
Step 9	**errdisable recovery cause l2ptguard** **Example:** Switch(config)# **errdisable recovery cause l2ptguard**	(Optional) Configures the recovery mechanism from a Layer 2 maximum-rate error so that the interface is reenabled and can try again. Errdisable recovery is disabled by default; when enabled, the default time interval is 300 seconds.
Step 10	**l2protocol-tunnel cos** *value* **Example:** Switch(config)# **l2protocol-tunnel cos value 7**	(Optional) Configures the CoS value for all tunneled Layer 2 PDUs. The range is 0 to 7; the default is the default CoS value for the interface. If none is configured, the default is 5.

	Command or Action	**Purpose**
Step 11	**end**	Returns to privileged EXEC mode.
	Example:	
	Switch(config)# **end**	
Step 12	**show l2protocol**	Displays the Layer 2 tunnel ports on the switch, including the protocols configured, the thresholds, and the counters.
	Example:	
	Switch# **show l2protocol**	
Step 13	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.
	Example:	
	Switch# **copy running-config startup-config**	

Configuring the SP Edge Switch

Before you begin

For EtherChannels, you need to configure both the SP (service-provider) edge switches and the customer switches for Layer 2 protocol tunneling.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport mode dot1q-tunnel**
5. **l2protocol-tunnel point-to-point** [**pagp** | **lacp** | **udld**]
6. **l2protocol-tunnel shutdown-threshold** [**point-to-point** [**pagp** | **lacp** | **udld**]] *value*
7. **l2protocol-tunnel drop-threshold** [**point-to-point** [**pagp** | **lacp** | **udld**]] *value*
8. **no cdp enable**
9. **spanning-tree bpdu filter enable**
10. **exit**
11. **errdisable recovery cause l2ptguard**
12. **l2protocol-tunnel cos** *value*
13. **end**
14. **show l2protocol**
15. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	Specifies the interface connected to the phone, and enters interface configuration mode.
Step 4	**switchport mode dot1q-tunnel** **Example:** Switch(config-if)# **switchport mode dot1q-tunnel**	Configures the interface as an IEEE 802.1Q tunnel port.
Step 5	**l2protocol-tunnel point-to-point** [**pagp** | **lacp** | **udld**] **Example:** Switch(config-if)# **l2protocol-tunnel point-to-point pagp**	(Optional) Enables point-to-point protocol tunneling for the desired protocol. If no keyword is entered, tunneling is enabled for all three protocols. **Note** To avoid a network failure, make sure that the network is a point-to-point topology before you enable tunneling for PAgP, LACP, or UDLD packets. **Note** Use the **no l2protocol-tunnel** [**point-to-point** [**pagp** | **lacp** | **udld**]] interface configuration command to disable point-to-point protocol tunneling for one of the Layer 2 protocols or for all three.
Step 6	**l2protocol-tunnel shutdown-threshold** [**point-to-point** [**pagp** | **lacp** | **udld**]] *value* **Example:** Switch(config-if)# **l2protocol-tunnel shutdown-threshold point-to-point pagp 100**	(Optional) Configures the threshold for packets-per-second accepted for encapsulation. The interface is disabled if the configured threshold is exceeded. If no protocol option is specified, the threshold applies to each of the tunneled Layer 2 protocol types. The range is 1 to 4096. The default is to have no threshold configured.

	Command or Action	**Purpose**
		Note If you also set a drop threshold on this interface, the **shutdown-threshold** value must be greater than or equal to the **drop-threshold** value.
		Note Use the **no l2protocol-tunnel shutdown-threshold** [**point-to-point** [**pagp** | **lacp** | **udld**]] and the **no l2protocol-tunnel drop-threshold** [[**point-to-point** [**pagp** | lacp | **udld**]] commands to return the shutdown and drop thresholds to the default settings.
Step 7	**l2protocol-tunnel drop-threshold** [**point-to-point** [**pagp** | **lacp** | **udld**]] *value* **Example:** Switch(config-if)# **l2protocol-tunnel drop-threshold point-to-point pagp 500**	(Optional) Configures the threshold for packets-per-second accepted for encapsulation. The interface drops packets if the configured threshold is exceeded. If no protocol option is specified, the threshold applies to each of the tunneled Layer 2 protocol types. The range is 1 to 4096. The default is to have no threshold configured. **Note** If you also set a shutdown threshold on this interface, the **drop-threshold** value must be less than or equal to the **shutdown-threshold** value.
Step 8	**no cdp enable** **Example:** Switch(config-if)# **no cdp enable**	Disables CDP on the interface.
Step 9	**spanning-tree bpdu filter enable** **Example:** Switch(config-if)# **spanning-tree bpdu filter enable**	Enables BPDU filtering on the interface.
Step 10	**exit** **Example:** Switch(config-if)# **exit**	Returns to global configuration mode.
Step 11	**errdisable recovery cause l2ptguard** **Example:** Switch(config)# **errdisable recovery cause l2ptguard**	(Optional) Configures the recovery mechanism from a Layer 2 maximum-rate error so that the interface is reenabled and can try again. Errdisable recovery is disabled by default; when enabled, the default time interval is 300 seconds.

	Command or Action	**Purpose**
Step 12	**l2protocol-tunnel cos** *value* Example: Switch(config)# **l2protocol-tunnel cos 2**	(Optional) Configures the CoS value for all tunneled Layer 2 PDUs. The range is 0 to 7; the default is the default CoS value for the interface. If none is configured, the default is 5.
Step 13	**end** Example: Switch(config)# **end**	Returns to privileged EXEC mode.
Step 14	**show l2protocol** Example: Switch)# **show l2protocol**	Displays the Layer 2 tunnel ports on the switch, including the protocols configured, the thresholds, and the counters.
Step 15	**copy running-config startup-config** Example: Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring the Customer Switch

Before you begin

For EtherChannels, you need to configure both the SP edge switch and the customer switches for Layer 2 protocol tunneling.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport trunk encapsulation dot1q**
5. **switchport mode trunk**
6. **udld port**
7. **channel-group** *channel-group-number* **mode desirable**
8. **exit**
9. **interface port-channel port-channel** *number*
10. **shutdown**
11. **no shutdown**
12. **end**
13. **show l2protocol**

14. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	Specifies the interface connected to the phone, and enters interface configuration mode.
Step 4	**switchport trunk encapsulation dot1q** **Example:** Switch(config)# **switchport trunk encapsulation dot1q**	Sets the trunking encapsulation format to IEEE 802.1Q.
Step 5	**switchport mode trunk** **Example:** Switch(config-if)# **switchport mode trunk**	Enables trunking on the interface.
Step 6	**udld port** **Example:** Switch(config-if)# **udld port**	Enables UDLD in normal mode on the interface.
Step 7	**channel-group** *channel-group-number* **mode desirable** **Example:** Switch(config-if)# **channel-group 25 mode desirable**	Assigns the interface to a channel group, and specifies desirable for the PAgP mode.
Step 8	**exit**	Returns to global configuration mode.

	Command or Action	Purpose
	Example: Switch(config-if)# **exit**	
Step 9	**interface port-channel port-channel** *number* **Example:** Switch(config)# **interface port-channel** **port-channel 25**	Enters port-channel interface mode.
Step 10	**shutdown** **Example:** Switch(config)# **shutdown**	Shuts down the interface.
Step 11	**no shutdown** **Example:** Switch(config)# **no shutdown**	Enables the interface.
Step 12	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 13	**show l2protocol** **Example:** Switch# **show l2protocol**	Displays the Layer 2 tunnel ports on the switch, including the protocols configured, the thresholds, and the counters.
Step 14	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file. **Note** Use the **no switchport mode trunk**, the **no udld enable**, and the **no channel group** *channel-group-number* **mode desirable** interface configuration commands to return the interface to the default settings.

Configuration Examples for IEEE 802.1Q and Layer 2 Protocol Tunneling

Example: Configuring an IEEE 802.1Q Tunneling Port

The following example shows how to configure an interface as a tunnel port, enable tagging of native VLAN packets, and verify the configuration. In this configuration, the VLAN ID for the customer connected to Gigabit Ethernet interface 7 on stack member 1 is VLAN 22.

```
Switch(config)# interface gigabitethernet1/0/7
Switch(config-if)# switchport access vlan 22
% Access VLAN does not exist. Creating vlan 22
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# exit
Switch(config)# vlan dot1q tag native
Switch(config)# end
Switch# show dot1q-tunnel interface gigabitethernet1/0/7
Port
-----
Gi1/0/1Port
-----
Switch# show vlan dot1q tag native
dot1q native vlan tagging is enabled
```

Example: Configuring Layer 2 Protocol Tunneling

The following example shows how to configure Layer 2 protocol tunneling for CDP, STP, and VTP and to verify the configuration.

```
Switch(config)# interface gigabitethernet1/0/11
Switch(config-if)# l2protocol-tunnel cdp
Switch(config-if)# l2protocol-tunnel stp
Switch(config-if)# l2protocol-tunnel vtp
Switch(config-if)# l2protocol-tunnel shutdown-threshold 1500
Switch(config-if)# l2protocol-tunnel drop-threshold 1000
Switch(config-if)# exit
Switch(config)# l2protocol-tunnel cos 7
Switch(config)# end
Switch# show l2protocol

COS for Encapsulated Packets: 7
Port Protocol Shutdown Drop Encapsulation Decapsulation Drop
Threshold Threshold Counter Counter Counter
------- -------- --------- --------- ------------- ------------- -------------
Gi0/11 cdp 1500 1000 2288 2282 0
stp 1500 1000 116 13 0
vtp 1500 1000 3 67 0
pagp ---- ---- 0 0 0
lacp ---- ---- 0 0 0
udld ---- ---- 0 0 0
```

Examples: Configuring the SP Edge and Customer Switches

This example shows how to configure the SP edge switch 1 and edge switch 2. VLANs 17, 18, 19, and 20 are the access VLANs, Fast Ethernet interfaces 1 and 2 are point-to-point tunnel ports with PAgP and UDLD enabled, the drop threshold is 1000, and Fast Ethernet interface 3 is a trunk port.

SP edge switch 1 configuration:

```
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# switchport access vlan 17
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# l2protocol-tunnel point-to-point pagp
Switch(config-if)# l2protocol-tunnel point-to-point udld
Switch(config-if)# l2protocol-tunnel drop-threshold point-to-point pagp 1000
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# switchport access vlan 18
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# l2protocol-tunnel point-to-point pagp
Switch(config-if)# l2protocol-tunnel point-to-point udld
Switch(config-if)# l2protocol-tunnel drop-threshold point-to-point pagp 1000
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/3
Switch(config-if)# switchport trunk encapsulation isl
Switch(config-if)# switchport mode trunk
```

SP edge switch 2 configuration:

```
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# switchport access vlan 19
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# l2protocol-tunnel point-to-point pagp
Switch(config-if)# l2protocol-tunnel point-to-point udld
Switch(config-if)# l2protocol-tunnel drop-threshold point-to-point pagp 1000
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# switchport access vlan 20
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# l2protocol-tunnel point-to-point pagp
Switch(config-if)# l2protocol-tunnel point-to-point udld
Switch(config-if)# l2protocol-tunnel drop-threshold point-to-point pagp 1000
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/3
Switch(config-if)# switchport trunk encapsulation isl
Switch(config-if)# switchport mode trunk
```

This example shows how to configure the customer switch at Site 1. Fast Ethernet interfaces 1, 2, 3, and 4 are set for IEEE 802.1Q trunking, UDLD is enabled, EtherChannel group 1 is enabled, and the port channel is shut down and then enabled to activate the EtherChannel configuration.

```
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# udld enable
Switch(config-if)# channel-group 1 mode desirable
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/2
```

```
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# udld enable
Switch(config-if)# channel-group 1 mode desirable
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/3
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# udld enable
Switch(config-if)# channel-group 1 mode desirable
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/4
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# udld enable
Switch(config-if)# channel-group 1 mode desirable
Switch(config-if)# exit
Switch(config)# interface port-channel 1
Switch(config-if)# shutdown
Switch(config-if)# no shutdown
Switch(config-if)# exit
```

Monitoring Tunneling Status

The following table describes the commands used to monitor tunneling status.

Table 122: Commands for Monitoring Tunneling

Command	Purpose
clear l2protocol-tunnel counters	Clears the protocol counters on Layer 2 protocol tunneling ports.
show dot1q-tunnel	Displays IEEE 802.1Q tunnel ports on the switch.
show dot1q-tunnel interface *interface-id*	Verifies if a specific interface is a tunnel port.
show l2protocol-tunnel	Displays information about Layer 2 protocol tunneling ports.
show errdisable recovery	Verifies if the recovery timer from a Layer 2 protocol-tunnel error disable state is enabled.
show l2protocol-tunnel interface *interface-id*	Displays information about a specific Layer 2 protocol tunneling port.
show l2protocol-tunnel summary	Displays only Layer 2 protocol summary information.
show vlan dot1q tag native	Displays the status of native VLAN tagging on the switch.

Where to Go Next

You can configure the following:

- VTP

- VLANs

- VLAN Trunking

- Private VLANs

- VLAN Membership Policy Server (VMPS)

- Voice VLANs

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	*Catalyst 2960-XR Switch VLAN Management Command Reference*

Standards and RFCs

Standard/RFC	Title
—	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Tunneling

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

CHAPTER 57

Configuring Voice VLANs

Prerequisites for Voice VLANs

The following are the prerequisites for voice VLANs:

- 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 supported on trunk ports.

- Before you enable voice VLAN, we recommend that you enable QoS on the switch by entering the **mls qos** global configuration command and configure the port trust state to trust by entering the **mls qos trust cos** interface configuration command. If you use the auto-QoS feature, these settings are automatically configured.

- 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.)

Restrictions for Voice VLANs

The following are the restrictions for voice VLANs:

- Do not configure voice VLAN on private VLAN ports.

• You cannot configure static secure MAC addresses in the voice VLAN.

Information About Voice VLAN

Voice VLANs

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 an 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.

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.

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).

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. 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.

Voice VLAN Configuration Guidelines

- 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.

- 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). If the VLAN is not listed, create the voice VLAN.

- The Power over Ethernet (PoE) switches are capable of automatically providing power to Cisco pre-standard and IEEE 802.3af-compliant powered devices if they are not being powered by an AC power source.

- 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).

- Voice VLAN ports can also be these port types:

    - Dynamic access port.

    - IEEE 802.1x authenticated port.

    Note 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.

| Note | 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. |

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.

How to Configure Voice VLAN

Configuring Cisco IP Phone Voice 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).

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **mls qos trust cos**
4. **switchport voice vlan** {*vlan-id* | **dot1p** | **none** | **untagged** }
5. **end**
6. Use one of the following:

    - **show interfaces** *interface-id* **switchport**
    - **show running-config interface** *interface-id*

7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* Example: Switch(config)# **interface gigabitethernet1/0/1**	Specifies the interface connected to the phone, and enters interface configuration mode.
Step 3	**mls qos trust cos** Example: Switch(config-if)# **mls qos trust cos**	Configures the interface to classify incoming traffic packets by using the packet CoS value. For untagged packets, the port default CoS value is used. **Note** Before configuring the port trust state, you must first globally enable QoS by using the **mls qos** global configuration command.
Step 4	**switchport voice vlan** {*vlan-id* | **dot1p** | **none** | **untagged** } Example: Switch(config-if)# **switchport voice vlan 125**	Configures how the Cisco IP Phone carries voice traffic: • *vlan-id*—Configures the phone to forward all voice traffic through the specified VLAN. By default, the Cisco IP Phone forwards the voice traffic with an IEEE 802.1Q priority of 5. Valid VLAN IDs are 1 to 4094. • **dot1p**—Configures the switch to accept voice and data IEEE 802.1p priority frames tagged with VLAN ID 0 (the native VLAN). By default, the switch drops all voice and data traffic tagged with VLAN 0. If configured for 802.1p the Cisco IP Phone forwards the traffic with an IEEE 802.1p priority of 5. • **none**—Allows the phone to use its own configuration to send untagged voice traffic. • **untagged**—Configures the phone to send untagged voice traffic. **Note** Before configuring the switch port to detect and recognize a Cisco IP phone, confirm that the phone is powered by PoE. The configuration fails when power is provided by an AC source.
Step 5	**end** Example:	Returns to privileged EXEC mode.

	Command or Action	**Purpose**
	Switch(config-if)# **end**	
Step 6	Use one of the following: • **show interfaces** *interface-id* **switchport** • **show running-config interface** *interface-id* **Example:** Switch# **show interfaces gigabitethernet1/0/1 switchport** or Switch# **show running-config interface gigabitethernet1/0/1**	Verifies your voice VLAN entries or your QoS and voice VLAN entries.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring the 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.

Follow these steps to set the priority of data traffic received from the non-voice port on the Cisco IP Phone:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport priority extend** {**cos** *value* | **trust**}
5. **end**
6. **show interfaces** *interface-id* **switchport**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable**	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Switch> **enable**	• Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/1**	Specifies the interface connected to the Cisco IP Phone, and enters interface configuration mode.
Step 4	**switchport priority extend** {**cos** *value*	**trust**} **Example:** Switch(config-if)# **switchport priority extend 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. **Note** To return the port to its default setting, use the **no switchport priority extend** interface configuration command.
Step 5	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.
Step 6	**show interfaces** *interface-id* **switchport** **Example:** Switch# **show interfaces gigabitethernet1/0/1 switchport**	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:**	(Optional) Saves your entries in the configuration file.

Command or Action	Purpose
Switch# **copy running-config startup-config**	

Monitoring Voice VLAN

To display voice VLAN configuration for an interface, use the **show interfaces** *interface-id* **switchport** privileged EXEC command.

Configuration Examples for Voice VLANs

Example: Configuring Cisco IP Phone Voice Traffic

This example shows how to configure a port connected to a Cisco IP Phone to use the CoS value to classify incoming traffic and to accept voice and data priority traffic tagged with VLAN ID 0:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# mls qos trust cos
Switch(config-if)# switchport voice vlan dot1p
Switch(config-if)# end
```

To return the port to its default setting, use the **no switchport voice vlan** interface configuration command.

This example shows how to enable switch port voice detect on a Cisco IP Phone:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitethernet 1/0/1
Switch(config-if)# switchport voice?
detect detection enhancement keyword
vlan VLAN for voice traffic

Switch(config-if)# switchport voice detect?
cisco-phone Cisco IP Phone

Switch(config-if)# switchport voice detect cisco-phone?
full-duplex Cisco IP Phone

Switch(config-if)# switchport voice detect cisco-phone full-duplex
full-duplex full duplex keyword

Switch(config-if)# end
```

This example shows how to disable switchport voice detect on a Cisco IP Phone:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitethernet 1/0/1
```

```
Switch(config-if)# no switchport voice detect cisco-phone
Switch(config-if)# no switchport voice detect cisco-phone full-duplex
```

Example: Configuring a Port Connected to an IP Phone Not to Change Frame Priority

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# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# switchport priority extend trust
Switch(config-if)# end
```

Where to Go Next

After configuring voice VLANs, you can configure the following:

- VTP

- VLANs

- VLAN trunking

- Private VLANs

- VLAN Membership Policy Server (VMPS)

- Tunneling

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	*Catalyst 2960-XR Switch VLAN Management Command Reference*

Standards and RFCs

Standard/RFC	Title
—	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Voice VLAN

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

PART XII

System Management

Administering the System

Information About Administering the Switch

System Time and Date Management

You can manage the system time and date on your switch using automatic configuration methods (RTC and NTP), or manual configuration methods.

> **Note** For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Configuration Fundamentals Command Reference* on *Cisco.com*.

System Clock

The basis of the 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:

- RTC
- 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 Coordinated Universal Time (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.

Real Time Clock

A real-time clock (RTC) keeps track of the current time on the switch. The switch is shipped to you with RTC set to GMT time until you reconfigure clocking parameters.

The benefits of an RTC are:

- RTC is battery-powered.

- System time is retained during power outage and at system reboot.

The RTC and NTP clocks are integrated on the switch. When NTP is enabled, the RTC time is periodically synchronized to the NTP clock to maintain accuracy.

Network Time Protocol

The 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.

The figure below shows a typical network example using NTP. Switch A is the NTP primary (formerly known as NTP primary), with the **Switch** 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 Switch, Switch B and Switch F, respectively.

Figure 112: Typical NTP Network Configuration
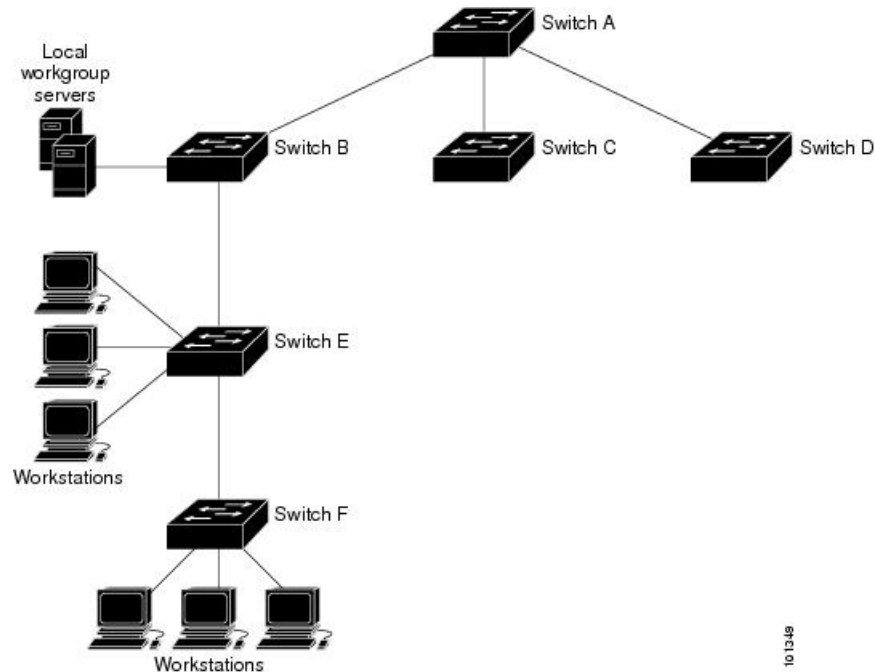
If the network is isolated from the Internet, Cisco's implementation of NTP allows a device to act as if it is synchronized through NTP, when in fact it has learned the time by using other means. Other devices then synchronize to that device through NTP.

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 Stratum

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.

NTP Associations

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.

NTP Security

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.

NTP Implementation

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 113: Typical NTP Network Configuration

The following figure shows a typical network example using NTP. Switch A is the NTP primary, with the Switch 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,

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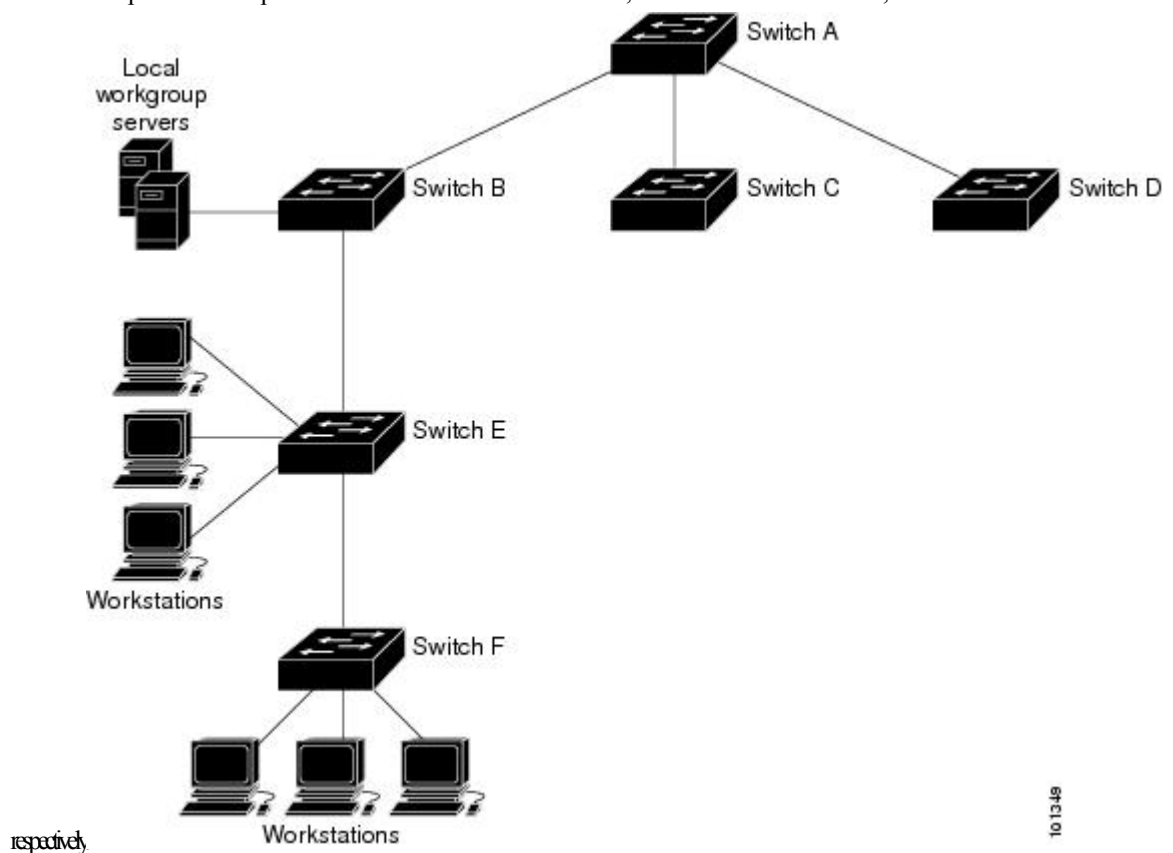
If the network is isolated from the Internet, NTP allows a device to act as if it is synchronized through NTP, when in fact it has learned the time by using other means. Other devices then synchronize to that device through NTP.

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 the *Implementing NTPv4 in IPv6* chapter of the *Cisco IOS IPv6 Configuration Guide, Release 12.4T*.

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.

For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Configuration Fundamentals Command Reference*, *Release 12.4* and the *Cisco IOS IP Command Reference*, *Volume 2 of 3: Routing Protocols*, *Release 12.4*.

Stack System Name and Prompt

If you are accessing a stack member through the active stack, you must use the **session** *stack-member-number* privileged EXEC command. The stack member number range is from 1 through 8. When you use this command, the stack member number is appended to the system prompt. For example, Switch-2# is the prompt in privileged EXEC mode for stack member 2, and the system prompt for the switch stack is Switch.

Default System Name and Prompt Configuration

The default switch system name and prompt is *Switch*.

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 Settings

Table 123: Default DNS Settings

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 is displayed 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 is also displayed on all connected terminals. It appears after the MOTD banner and before the login prompts.

Note For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.4*.

Default Banner Configuration

The MOTD and login banners are not configured.

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).

Note For complete syntax and usage information for the commands used in this section, see the command reference for this release.

MAC Address Table Creation

With multiple MAC addresses supported on all ports, you can connect any port on the switch to 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 devices 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.

MAC Addresses and Switch Stacks

The MAC address tables on all stack members are synchronized. At any given time, each stack member has the same copy of the address tables for each VLAN. When an address ages out, the address is removed from the address tables on all stack members. When a Switch joins a switch stack, that Switch receives the addresses for each VLAN learned on the other stack members. When a stack member leaves the switch stack, the remaining stack members age out or remove all addresses learned by the former stack member.

Default MAC Address Table Settings

The following table shows the default settings for the MAC address table.

Table 124: Default Settings for the MAC Address

Feature	Default Setting
Aging time	300 seconds
Dynamic addresses	Automatically learned
Static addresses	None configured

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.

For CLI procedures, see the Cisco IOS Release 12.4 documentation on *Cisco.com*.

How to Administer the Switch

Configuring the Time and Date Manually

System time remains accurate through restarts and reboot, however, you can manually configure the time and date after the system is restarted.

We recommend that you use manual configuration only when necessary. If you have an outside source to which the switch can synchronize, you do not need to manually set the system clock.

Note You must reconfigure this setting if you have manually configured the system clock before the active switchstack's active switch fails and a different stack member assumes the role of active switchstack's active switch.

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.

Follow these steps to set the system clock:

SUMMARY STEPS

1. **enable**
2. Use one of the following:

   - **clock set** *hh:mm:ss day month year*
   - **clock set** *hh:mm:ss month day year*

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	Use one of the following: • **clock set** *hh:mm:ss day month year* • **clock set** *hh:mm:ss month day year* **Example:** Switch# **clock set 13:32:00 23 March 2013**	Manually set 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

Follow these steps to manually configure the time zone:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **clock timezone** *zone hours-offset* [*minutes-offset*]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**clock timezone** *zone hours-offset* [*minutes-offset*] **Example:** Switch(config)# **clock timezone AST -3 30**	Sets the time zone. Internal time is kept in Coordinated Universal Time (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. This available where the local time zone is a percentage of an hour different from UTC.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

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:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **clock summer-time** *zone* **date** *date month year hh:mm date month year hh:mm* [*offset*]]
4. **clock summer-time** *zone* **recurring** [*week day month hh:mm week day month hh:mm* [*offset*]]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**clock summer-time** *zone* **date** *date month year hh:mm date month year hh:mm* [*offset*]] **Example:** Switch(config)# **clock summer-time PDT date 10 March 2013 2:00 3 November 2013 2:00**	Configures summer time to start and end on specified days every year.
Step 4	**clock summer-time** *zone* **recurring** [*week day month hh:mm week day month hh:mm* [*offset*]] **Example:** Switch(config)# **clock summer-time PDT recurring 10 March 2013 2:00 3 November 2013 2:00**	Configures summer time to start and end on the specified days every year. 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. 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. If the starting month is after the ending month, the system assumes that you are in the southern hemisphere. • *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 4, **first**, or **last**).

	Command or Action	Purpose
		• (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.
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Follow these steps if summer time in your area does not follow a recurring pattern (configure the exact date and time of the next summer time events):

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **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*]]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	**Purpose**
	Switch> **enable**	
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**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. • For *zone*, specify the name of the time zone (for example, PDT) to be displayed when summer time is in effect. • (Optional) For *week*, specify the week of the month (1 to 5 or last). • (Optional) For *day*, specify the day of the week (Sunday, Monday...). • (Optional) For *month*, specify the month (January, February...). • (Optional) For *hh:mm*, specify the time (24-hour format) in hours and minutes. • (Optional) For *offset*, specify the number of minutes to add during summer time. The default is 60.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring a System Name

Follow these steps to manually configure a system name:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **hostname** *name*
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**hostname** *name* **Example:** Switch(config)# **hostname remote-users**	Configures a system name. When you set the system name, it is also used as the system prompt. 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.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.

	Command or Action	**Purpose**
Step 6	**copy running-config startup-config** **Example:** Switch# `copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

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.

Follow these steps to set up your switch to use the DNS:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip domain-name** *name*
4. **ip name-server** *server-address1* [*server-address2 ... server-address6*]
5. **ip domain-lookup** [**nsap** | **source-interface** *interface*]
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> `enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# `configure terminal`	Enters global configuration mode.
Step 3	**ip domain-name** *name* **Example:** Switch(config)# `ip domain-name Cisco.com`	Defines a default domain name that the software uses to complete unqualified hostnames (names without a dotted-decimal domain name).

	Command or Action	**Purpose**
		Do not include the initial period that separates an unqualified name from the domain name.
		At boot 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).
Step 4	**ip name-server** *server-address1* [*server-address2 ... server-address6*] **Example:** Switch(config)# **ip name-server 192.168.1.100 192.168.1.200 192.168.1.300**	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.
Step 5	**ip domain-lookup** [**nsap** | **source-interface** *interface*] **Example:** Switch(config)# **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).
Step 6	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 7	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 8	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

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

Follow these steps to configure a MOTD login banner:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **banner motd** *c message c*
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**banner motd** *c message c* **Example:** Switch(config)# **banner motd #** This is a secure site. Only authorized users are allowed. For access, contact technical support. #	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.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.

	Command or Action	**Purpose**
Step 6	**copy running-config startup-config** **Example:** Switch# `copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

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.

Follow these steps to configure a login banner:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **banner login** *c message c*
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> `enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# `configure terminal`	Enters global configuration mode.

Step 3	**banner login** *c message c*
	
	Example:
	
	Switch(config)# `banner login $`
	`Access for authorized users only.`
	`Please enter your username and`
	`password.`
	`$`
		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.

	Command or Action	Purpose
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Managing the MAC Address Table

Changing the Address Aging Time

Follow these steps to configure the dynamic address table aging time:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mac address-table aging-time** [*0* | *10-1000000*] [**routed-mac** | **vlan** *vlan-id*]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:**	Enters global configuration mode.

	Command or Action	Purpose
	Switch# **configure terminal**	
Step 3	**mac address-table aging-time** [*0* | *10-1000000*] [**routed-mac** | **vlan** *vlan-id*] **Example:** Switch(config)# **mac address-table aging-time 500 vlan 2**	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 4094.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring MAC Address Change Notification Traps

Follow these steps to configure the switch to send MAC address change notification traps to an NMS host:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server host** *host-addr community-string notification-type* { **informs** \| **traps** } {**version** {**1** \| **2c** \| **3**}} {**vrf** *vrf instance name*}
4. **snmp-server enable traps mac-notification change**
5. **mac address-table notification change**
6. **mac address-table notification change** [**interval** *value*] [**history-size** *value*]
7. **interface** *interface-id*
8. **snmp trap mac-notification change** {**added** \| **removed**}
9. **end**
10. **show running-config**
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**snmp-server host** *host-addr community-string notification-type* { **informs** | **traps** } {**version** {**1** | **2c** | **3**}} {**vrf** *vrf instance name*} **Example:** Switch(config)# **snmp-server host 172.20.10.10 traps private mac-notification**	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. Though you can set this string by using the **snmp-server host** command, 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. • **vrf** *vrf instance name*—Specifies the VPN routing/forwarding instance for this host.
Step 4	**snmp-server enable traps mac-notification change** **Example:** Switch(config)# **snmp-server enable traps mac-notification change**	Enables the switch to send MAC address change notification traps to the NMS.
Step 5	**mac address-table notification change** **Example:** Switch(config)# **mac address-table notification change**	Enables the MAC address change notification feature.

	Command or Action	Purpose
Step 6	**mac address-table notification change** [**interval** *value*] [**history-size** *value*] **Example:** Switch(config)# **mac address-table notification change interval 123** Switch(config)#**mac address-table notification change history-size 100**	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.
Step 7	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/2**	Enters interface configuration mode, and specifies the Layer 2 interface on which to enable the SNMP MAC address notification trap.
Step 8	**snmp trap mac-notification change** {**added**	**removed**} **Example:** Switch(config-if)# **snmp trap mac-notification change added**	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.
Step 9	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 10	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 11	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

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.

Follow these steps to configure the switch to send MAC address-move notification traps to an NMS host:

SUMMARY STEPS

    1. **enable**

    2. **configure terminal**

    3. **snmp-server host** *host-addr* {**traps** | **informs**} {**version** {**1** | **2c** | **3**}} *community-string notification-type*

    4. **snmp-server enable traps mac-notification move**

    5. **mac address-table notification mac-move**

    6. **end**

    7. **show running-config**

    8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**snmp-server host** *host-addr* {**traps**	**informs**} {**version** {**1**	**2c**	**3**}} *community-string notification-type* **Example:** Switch(config)# **snmp-server host** **172.20.10.10 traps private mac-notification**	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. Though you can set this string by using the **snmp-server host** command, 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.
Step 4	**snmp-server enable traps mac-notification move** **Example:** Switch(config)# **snmp-server enable traps** **mac-notification move**	Enables the switch to send MAC address move notification traps to the NMS.

	Command or Action	**Purpose**
Step 5	**mac address-table notification mac-move** **Example:** Switch(config)# **mac address-table notification mac-move**	Enables the MAC address move notification feature.
Step 6	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 7	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 8	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

What to do next

To disable MAC address-move notification traps, use the **no snmp-server enable traps mac-notification move** global configuration command. To disable the MAC address-move notification feature, use the **no mac address-table notification mac-move** global configuration command.

You can verify your settings by entering the **show mac address-table notification mac-move** privileged EXEC commands.

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.

Follow these steps to configure the switch to send MAC address table threshold notification traps to an NMS host:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server host** *host-addr* {**traps** / **informs**} {**version** {**1** | **2c** | **3**}} *community-string notification-type*
4. **snmp-server enable traps mac-notification threshold**
5. **mac address-table notification threshold**
6. **mac address-table notification threshold** [**limit** *percentage*] | [**interval** *time*]

7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**snmp-server host** *host-addr* {**traps** / **informs**} {**version** {**1** | **2c** | **3**}} *community-string notification-type* **Example:** Switch(config)# **snmp-server host 172.20.10.10 traps private mac-notification**	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.
Step 4	**snmp-server enable traps mac-notification threshold** **Example:** Switch(config)# **snmp-server enable traps mac-notification threshold**	Enables MAC threshold notification traps to the NMS.
Step 5	**mac address-table notification threshold** **Example:** Switch(config)# **mac address-table notification threshold**	Enables the MAC address threshold notification feature.

	Command or Action	Purpose
Step 6	**mac address-table notification threshold** [**limit** *percentage*]	[**interval** *time*] **Example:** Switch(config)# **mac address-table notification threshold interval 123** Switch(config)# **mac address-table notification threshold limit 78**	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.
Step 7	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 8	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 9	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Adding and Removing Static Address Entries

Follow these steps to add a static address:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mac address-table static** *mac-addr* **vlan** *vlan-id* **interface** *interface-id*
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:**	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
	Switch> **enable**	
Step 2	**configure terminal**	Enters global configuration mode.
	Example:	
	Switch# **configure terminal**	
Step 3	**mac address-table static** *mac-addr* **vlan** *vlan-id* **interface** *interface-id*	Adds a static address to the MAC address table.
	Example:	• *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.
	Switch(config)# **mac address-table static c2f3.220a.12f4 vlan 4 interface gigabitethernet 1/0/1**	• *vlan-id*—Specifies the VLAN for which the packet with the specified MAC address is received. Valid VLAN IDs are 1 to 4094.
		• *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.
Step 4	**end**	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.
	Example:	
	Device(config)# **end**	
Step 5	**show running-config**	Verifies your entries.
	Example:	
	Switch# **show running-config**	
Step 6	**copy running-config startup-config**	(Optional) Saves your entries in the configuration file.
	Example:	
	Switch# **copy running-config startup-config**	

Configuring Unicast MAC Address Filtering

Follow these steps to configure the Switch to drop a source or destination unicast static address:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mac address-table static** *mac-addr* **vlan** *vlan-id* **drop**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.

Step 3	**mac address-table static** *mac-addr* **vlan** *vlan-id* **drop**
	
	Example:
	
	Switch(config)# **mac address-table static c2f3.220a.12f4 vlan 4 drop**
		Enables unicast MAC address filtering and configure 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 (48-bit). 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 4094.

Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:**	(Optional) Saves your entries in the configuration file.

Command or Action	Purpose
Switch# **copy running-config startup-config**	

Monitoring and Maintaining Administration of the Switch

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 *mac-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 *interface-name*	Displays the MAC address table information for the specified interface.
show mac address-table move update	Displays the MAC address table move update information.
show mac address-table multicast	Displays a list of multicast MAC addresses.
show mac address-table notification {change | mac-move | threshold}	Displays the MAC notification parameters and history table.
show mac address-table secure	Displays the secure MAC addresses.
show mac address-table static	Displays only static MAC address table entries.
show mac address-table vlan *vlan-id*	Displays the MAC address table information for the specified VLAN.

Configuration Examples for Switch Administration

Example: Setting the System Clock

This example shows how to manually set the system clock:

```
Switch# clock set 13:32:00 23 July 2013
```

Examples: Configuring Summer Time

This example (for daylight savings time) shows how to specify that summer time starts on March 10 at 02:00 and ends on November 3 at 02:00:

```
Switch(config)# clock summer-time PDT recurring PST date
10 March 2013 2:00 3 November 2013 2:00
```

This example shows how to set summer time start and end dates:

```
Switch(config)#clock summer-time PST date
20 March 2013 2:00 20 November 2013 2:00
```

Example: Configuring a MOTD Banner

This example shows how to configure a MOTD banner 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 192.0.2.15

Trying 192.0.2.15...

Connected to 192.0.2.15.

Escape character is '^]'.

This is a secure site. Only authorized users are allowed.

For access, contact technical support.
```

```
    User Access Verification

    Password:
```

Example: Configuring a Login Banner

This example shows how to configure a login banner 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)#
```

Example: Configuring MAC Address Change Notification Traps

This example shows how to specify 172.20.10.10 as the NMS, enable 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 gigabitethernet1/2/1
Switch(config-if)# snmp trap mac-notification change added
```

Example: Configuring MAC Threshold Notification Traps

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
```

Example: Adding the Static Address to the MAC Address Table

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 gigabitethernet1/1/1
```

Example: Configuring Unicast MAC Address Filtering

This example shows how to enable unicast MAC address filtering and how to configure 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 for Switch Administration

Related Documents

Related Topic	Document Title
Switch administration commands	*Catalyst 2960-XR Switch System Management Command Reference*
Network management configuration	*Catalyst 2960-XR Switch Network Management Configuration Guide*
Layer 2 configuration	*Catalyst 2960-XR Switch Layer 2 Configuration Guide*
VLAN configuration	*Catalyst 2960-XR Switch VLAN Management Configuration Guide*
Platform-independent command references	*Cisco IOS 15.3M&T Command References*
Platform-independent configuration information	*Cisco IOS 15.3M&T Configuration Guides*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Switch Administration

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

The image shows a city skyline photograph as part of the chapter header design.

Performing Switch Setup Configuration

Information About Performing Switch Setup Configuration

Review the sections in this module before performing your initial switch configuration tasks that include IP address assignments and DHCP autoconfiguration.

Boot Process

To start your switch, you need to follow the procedures in the getting started guide or the hardware installation guide for installing and powering on the switch and setting up the initial switch configuration (IP address, subnet mask, default gateway, secret and Telnet passwords, and so forth).

The boot loader software performs the normal boot process and includes these activities:

- Locates the bootable (base) package in the bundle or installed package set.

- Performs low-level CPU initialization. It initializes the CPU registers, which control where physical memory is mapped, its quantity, its speed, and so forth.

- Performs power-on self-test (POST) for the CPU subsystem and tests the system DRAM.

- Initializes the file systems 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 systems before the operating system is loaded. Normally, the boot loader is used only to load, decompress, and start 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 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 operation 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.

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

- Baud rate default is 9600.

- Data bits default is 8.

> **Note** If the data bits option is set to 8, set the parity option to none.

- Stop bits default is 2 (minor).

- Parity settings default is none.

Switches 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.

It 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.

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 described in the *Boot Process* section.

Default Switch Information

Table 125: Default Switch Information

Feature	Default Setting
IP address and subnet mask	No IP address or subnet mask are defined.
Default gateway	No default gateway is defined.
Enable secret password	No password is defined.
Hostname	The factory-assigned default hostname is Switch.

Feature	Default Setting
Telnet password	No password is defined.
Cluster command switch functionality	Disabled.
Cluster name	No cluster name is defined.

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 an operation 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 want to use 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.

Note   We recommend a redundant connection between a switch stack and the DHCP, DNS, and TFTP servers. This is to help ensure that these servers remain accessible in case one of the connected stack members is removed from the switch stack.

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.

This is the sequence of messages that are exchanged between the DHCP client and the DHCP server.

Figure 114: DHCP Client and Server Message Exchange
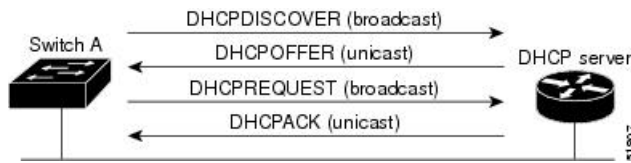
The client, Switch A, broadcasts a DHCPDISCOVER message to locate a DHCP server. The DHCP server offers configuration parameters (such as an IP address, subnet mask, gateway IP address, DNS IP address, a lease for the IP address, and so forth) to the client in a DHCPOFFER unicast message.

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 server.

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 client; 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. Simultaneous image and configuration upgrade for all switches in the network 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.

Restrictions for DHCP-based Autoconfiguration

- 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 if the configuration file is corrupted.

- 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. If the downloaded configuration is saved to the startup configuration, the feature is not triggered during subsequent system restarts.

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.)

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 Cisco IOS image file) settings.

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:

- You should 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)

- DNS server IP address (optional)

- Router IP address (default gateway address to be used by the switch) (required)

- 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.)

Purpose of the 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, *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. The preferred solution is to configure the DHCP server with all the required information.

Purpose of the 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 LAN or on a different LAN from the switch. If it is on a different LAN, the switch must be able to access it through a router.

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 the console connection. Unplug the switch power cord, then reconnect the power cord. Hold down the **MODE** button until you see the boot loader switch prompt

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, functions. 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 present; it has a value if it is listed 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

This table describes the function of the most common environment variables.

Table 126: Common 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** {*filesystem* **:** */file-url ...* Specifies the Cisco IOS image to load during the next boot cycle and the stack members on which the image is loaded. This command changes the setting of the BOOT environment variable.
MANUAL_BOOT	**set MANUAL_BOOT yes** Decides whether the switch automatically or manually boots. 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 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.

Variable	Boot Loader Command	Cisco IOS Global Configuration Command
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.
SWITCH_NUMBER	**set SWITCH_NUMBER** *stack-member-number* Changes the member number of a stack member.	**switch** *current-stack-member-number* **renumber** *new-stack-member-number* Changes the member number of a stack member.
SWITCH_PRIORITY	**set SWITCH_PRIORITY** *stack-member-number* Changes the priority value of a stack member.	**switch** *stack-member-number* **priority** *priority-number* Changes the priority value of a stack member.
BAUD	**set BAUD** *baud-rate*	**line console 0** **speed***speed-value* Configures the baud rate.
ENABLE_BREAK	**set ENABLE_BREAK yes/no**	**boot enable-break switch yes/no** This command can be issued when the flash filesystem is initialized when **ENABLE_BREAK** is set to **yes**.

Environment Variables for TFTP

When the switch is connected to a PC through the Ethernet management port, you can download or upload a configuration file to the boot loader by using TFTP. Make sure the environment variables in this table are configured.

Table 127: Environment Variables for TFTP

Variable	Description
MAC_ADDR	Specifies the MAC address of the switch. **Note** We recommend that you do not modify this variable. However, if you modify this variable after the boot loader is up or the value is different from the saved value, enter this command before using TFTP.

Variable	Description
IP_ADDR	Specifies the IP address and the subnet mask for the associated IP subnet of the switch.
DEFAULT_ROUTER	Specifies the IP address and subnet mask of the default gateway.

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:

- Reload of the software to take affect in the specified minutes or hours and minutes. The reload must take place within approximately 24 hours. You can specify the reason for the reload in a string up to 255 characters in length.

- Reload of the software 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 then 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 an existing switch in the network so that it can support the autoconfiguration of a new switch.

SUMMARY STEPS

1. **configure terminal**
2. **ip dhcp pool** *poolname*
3. **boot** *filename*
4. **network** *network-number mask prefix-length*
5. **default-router** *address*
6. **option 150** *address*
7. **exit**
8. **tftp-server flash**:*filename.text*
9. **interface** *interface-id*
10. **no switchport**
11. **ip address** *address mask*
12. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**ip dhcp pool** *poolname* **Example:** Switch(config)# **ip dhcp pool pool**	Creates a name for the DHCP server address pool, and enters DHCP pool configuration mode.
Step 3	**boot** *filename* **Example:** Switch(dhcp-config)# **boot config-boot.text**	Specifies the name of the configuration file that is used as a boot image.
Step 4	**network** *network-number mask prefix-length* **Example:** Switch(dhcp-config)# **network 10.10.10.0 255.255.255.0**	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 (/).
Step 5	**default-router** *address* **Example:**	Specifies the IP address of the default router for a DHCP client.

	Command or Action	**Purpose**
	Switch(dhcp-config)# **default-router 10.10.10.1**	
Step 6	**option 150** *address* **Example:** Switch(dhcp-config)# **option 150 10.10.10.1**	Specifies the IP address of the TFTP server.
Step 7	**exit** **Example:** Switch(dhcp-config)# **exit**	Returns to global configuration mode.
Step 8	**tftp-server flash**:*filename.text* **Example:** Switch(config)# **tftp-server flash:config-boot.text**	Specifies the configuration file on the TFTP server.
Step 9	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitethernet1/0/4**	Specifies the address of the client that will receive the configuration file.
Step 10	**no switchport** **Example:** Switch(config-if)# **no switchport**	Puts the interface into Layer 3 mode.
Step 11	**ip address** *address mask* **Example:** Switch(config-if)# **ip address 10.10.10.1 255.255.255.0**	Specifies the IP address and mask for the interface.
Step 12	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.

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

This task describes DHCP autoconfiguration to configure TFTP and DHCP settings on an existing switch to support the installation of a new switch.

Before you begin

You must first 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 (forexample, c3750e-ipservices-mz.122-44.3.SE.tarc3750x-ipservices-mz.122-53.3.SE2.tar). This image must be a tar and not a bin file.

SUMMARY STEPS

1. **configure terminal**
2. **ip dhcp pool** *poolname*
3. **boot** *filename*
4. **network** *network-number mask prefix-length*
5. **default-router** *address*
6. **option 150** *address*
7. **option 125** *hex*
8. **copy tftp flash** *filename.txt*
9. **copy tftp flash** *imagename.bin*
10. **exit**
11. **tftp-server flash:** *config.text*
12. **tftp-server flash:** *imagename.bin*
13. **tftp-server flash:** *filename.txt*
14. **interface** *interface-id*
15. **no switchport**
16. **ip address** *address mask*
17. **end**
18. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**ip dhcp pool** *poolname* **Example:** Switch(config)# **ip dhcp pool pool1**	Creates a name for the DHCP server address pool and enter DHCP pool configuration mode.

	Command or Action	**Purpose**
Step 3	**boot** *filename* **Example:** Switch(dhcp-config)# **boot config-boot.text**	Specifies the name of the file that is used as a boot image.
Step 4	**network** *network-number mask prefix-length* **Example:** Switch(dhcp-config)# **network 10.10.10.0** **255.255.255.0**	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 (/).
Step 5	**default-router** *address* **Example:** Switch(dhcp-config)# **default-router 10.10.10.1**	Specifies the IP address of the default router for a DHCP client.
Step 6	**option 150** *address* **Example:** Switch(dhcp-config)# **option 150 10.10.10.1**	Specifies the IP address of the TFTP server.
Step 7	**option 125** *hex* **Example:** Switch(dhcp-config)# **option 125 hex** **0000.0009.0a05.08661.7574.6f69.6e73.7461.6c6c.5f64.686370**	Specifies the path to the text file that describes the path to the image file.
Step 8	**copy tftp flash** *filename.txt* **Example:** Switch(config)# **copy tftp flash image.bin**	Uploads the text file to the switch.
Step 9	**copy tftp flash** *imagename.bin* **Example:** Switch(config)# **copy tftp flash image.bin**	Uploads the tar file for the new image to the switch.
Step 10	**exit** **Example:**	Returns to global configuration mode.

	Command or Action	Purpose
	Switch(dhcp-config)# **exit**	
Step 11	**tftp-server flash:** *config.text* **Example:** Switch(config)# **tftp-server flash:config-boot.text**	Specifies the Cisco IOS configuration file on the TFTP server.
Step 12	**tftp-server flash:** *imagename.bin* **Example:** Switch(config)# **tftp-server flash:image.bin**	Specifies the image name on the TFTP server.
Step 13	**tftp-server flash:** *filename.txt* **Example:** Switch(config)# **tftp-server flash:boot-config.text**	Specifies the text file that contains the name of the image file to download
Step 14	**interface** *interface-id* **Example:** Switch(config)# **interface gigabitEthernet1/0/4**	Specifies the address of the client that will receive the configuration file.
Step 15	**no switchport** **Example:** Switch(config-if)# **no switchport**	Puts the interface into Layer 3 mode.
Step 16	**ip address** *address mask* **Example:** Switch(config-if)# **ip address 10.10.10.1 255.255.255.0**	Specifies the IP address and mask for the interface.
Step 17	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.

	Command or Action	**Purpose**
Step 18	**copy running-config startup-config** Example: Switch(config-if)# **end**	(Optional) Saves your entries in the configuration file.

Configuring the Client to Download Files from DHCP Server

✎

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

SUMMARY STEPS

1. **configure terminal**
2. **boot host dhcp**
3. **boot host retry timeout** *timeout-value*
4. **banner config-save ^C** *warning-message* **^C**
5. **end**
6. **show boot**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 2	**boot host dhcp** Example: Switch(conf)# **boot host dhcp**	Enables autoconfiguration with a saved configuration.
Step 3	**boot host retry timeout** *timeout-value* Example: Switch(conf)# **boot host retry timeout 300**	(Optional) Sets the amount of time the system tries to download a configuration file. **Note** If you do not set a timeout, the system will try indefinitely to obtain an IP address from the DHCP server.
Step 4	**banner config-save ^C** *warning-message* **^C** Example:	(Optional) Creates warning messages to be displayed when you try to save the configuration file to NVRAM.

	Command or Action	Purpose
	```Switch(conf)# banner config-save ^C Caution - Saving Configuration File to NVRAM May Cause You to No longer Automatically Download Configuration Files at Reboot^C```	
Step 5	**end** **Example:** ```Switch(config-if)# end```	Returns to privileged EXEC mode.
Step 6	**show boot** **Example:** ```Switch# show boot```	Verifies the configuration.

Manually Assigning IP Information to Multiple SVIs

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

> **Note** If the switch is running the IP Lite image, you can also manually assign IP information to a port if you first put the port into Layer 3 mode by using the **no switchport** interface configuration command.

SUMMARY STEPS

1. **configure terminal**
2. **interface vlan** *vlan-id*
3. **ip address** *ip-address subnet-mask*
4. **exit**
5. **ip default-gateway** *ip-address*
6. **end**
7. **show interfaces vlan** *vlan-id*
8. **show ip redirects**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** ```Switch# configure terminal```	Enters global configuration mode.

	Command or Action	**Purpose**
Step 2	**interface vlan** *vlan-id* **Example:** Switch(config)# **interface vlan 99**	Enters interface configuration mode, and enter the VLAN to which the IP information is assigned. The range is 1 to 4094.
Step 3	**ip address** *ip-address subnet-mask* **Example:** Switch(config-vlan)# **ip address 10.10.10.2 255.255.255.0**	Enters the IP address and subnet mask.
Step 4	**exit** **Example:** Switch(config-vlan)# **exit**	Returns to global configuration mode.
Step 5	**ip default-gateway** *ip-address* **Example:** Switch(config)# **ip default-gateway 10.10.10.1**	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.
Step 6	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 7	**show interfaces vlan** *vlan-id* **Example:** Switch# **show interfaces vlan 99**	Verifies the configured IP address.
Step 8	**show ip redirects** **Example:** Switch# **show ip redirects**	Verifies the configured default gateway.

Configuring the NVRAM Buffer Size

The default NVRAM buffer size is 512 KB. In some cases, the configuration file might be too large to save to NVRAM. Typically, this occurs when you have many switches in a switch stack. You can configure the size of the NVRAM buffer to support larger configuration files. The new NVRAM buffer size is synced to all current and new member switches.

> ✎
>
> **Note**   After you configure the NVRAM buffer size, reload the switch or switch stack.
>
> When you add a switch to a stack and the NVRAM size differs, the new switch syncs with the stack and reloads automatically.

SUMMARY STEPS

1. **configure terminal**
2. **boot buffersize** *size*
3. **end**
4. **show boot**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**boot buffersize** *size* **Example:** Switch(config)# **boot buffersize 524288**	Configures the NVRAM buffersize in KB. The valid range for *size* is from 4096 to 1048576.
Step 3	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 4	**show boot** **Example:** Switch# **show boot**	Verifies the configuration.

Modifying the Switch 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 cycle.

Before you begin

Use a standalone switch for this task.

SUMMARY STEPS

1. **configure terminal**
2. **boot flash**:*/file-url*
3. **end**
4. **show boot**
5. **copy running-config startup-config**

DETAILED STEPS

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

	Command or Action	Purpose
	Switch# **copy running-config startup-config**	

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.

SUMMARY STEPS

1. **configure terminal**
2. **boot manual**
3. **end**
4. **show boot**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**boot manual** **Example:** Switch(config)# **boot manual**	Enables the switch to manually boot up during the next boot cycle.
Step 3	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 4	**show boot** **Example:** Switch# **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.

	Command or Action	Purpose
		• *filesystem*:—Uses flash: for the system board flash device. switch: **boot flash:** • For *file-url*—Specifies the path (directory) and the name of the bootable image. Filenames and directory names are case-sensitive.
Step 5	**copy running-config startup-config** Example: Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

Configuring a Scheduled Software Image Reload

This task describes how to configure your switch to reload the software image at a later time.

SUMMARY STEPS

1. **configure terminal**
2. **copy running-config startup-config**
3. **reload in** [*hh*:]*mm* [*text*]
4. **reload at** *hh*: *mm* [*month day* | *day month*] [*text*]
5. **reload cancel**
6. **show reload**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 2	**copy running-config startup-config** Example: **copy running-config startup-config**	Saves your switch configuration information to the startup configuration before you use the **reload** command.
Step 3	**reload in** [*hh*:]*mm* [*text*] Example: Switch(config)# **reload in 12** System configuration has been modified. Save?	Schedules a reload of the software to take affect 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.

	Command or Action	**Purpose**
	`[yes/no]: `**`y`**	
Step 4	**reload at** *hh*: *mm* [*month day* | *day month*] [*text*] **Example:** `Switch(config)# `**`reload at 14:00`**	Specifies the time in hours and minutes for the reload to occur. **Note** Use the **at** keyword only if the switch system clock has been set (through Network Time Protocol (NTP), the hardware calendar, or manually). The time is relative to the configured time zone on the switch. To schedule reloads across several switches to occur simultaneously, the time on each switch must be synchronized with NTP.
Step 5	**reload cancel** **Example:** `Switch(config)# `**`reload cancel`**	Cancels a previously scheduled reload.
Step 6	**show reload** **Example:** **`show reload`**	Displays information about a previously scheduled reload or identifies if a reload has been scheduled on the switch.

Monitoring Switch Setup Configuration

Example: Verifying the Switch Running Configuration

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

Current configuration: 1363 bytes
!
version 12.4
no service pad
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Stack1
!
enable secret 5 $1$ej9.$DMUvAUnZOAmvmgqBEzIxE0
!
.
<output truncated>
.
interface gigabitethernet6/0/2
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
```

Examples: Displaying Software Install

This example displays software bootup in install mode:

```
switch# boot flash:/c2960x-universalk9-mz-150-2.EX1/c2960x-universalk9-mz-150-2.EX1.bin
```

Configuration Examples for Performing Switch Setup

Example: Configuring a Switch as a DHCP Server

```
Switch# configure terminal
Switch(config)# ip dhcp pool pool1
Switch(dhcp-config)# network 10.10.10.0 255.255.255.0
Switch(dhcp-config)# boot 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 gigabitethernet1/0/4
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.10.10.1 255.255.255.0
Switch(config-if)# end
```

Example: Configuring DHCP Auto-Image Update

```
Switch# configure terminal
Switch(config)# ip dhcp pool pool1
Switch(dhcp-config)# network 10.10.10.0 255.255.255.0
Switch(dhcp-config)# boot 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:image_name
Switch(config)# tftp-server flash:boot-config.text
Switch(config)# tftp-server flash: autoinstall_dhcp
Switch(config)# interface gigabitethernet1/0/4
Switch(config-if)# no switchport
Switch(config-if)# ip address 10.10.10.1 255.255.255.0
Switch(config-if)# end
```

Example: Configuring a Switch to Download Configurations from a 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(config)# boot host dhcp
Switch(config)# boot host retry timeout 300
Switch(config)# banner config-save ^C Caution - Saving Configuration File to NVRAM May Cause
 You to No longer 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#
```

Example: Configuring NVRAM Buffer Size

```
Switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# boot buffersize 600000
Switch(config)# 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   :
Auto upgrade       : yes
Auto upgrade path  :
NVRAM/Config file
```

```
      buffer size:   600000
Timeout for Config
        Download:    300 seconds
Config Download
     via DHCP:       enabled (next boot: enabled)
Switch#
```

Additional References for Performing Switch Setup

Related Documents

Related Topic	Document Title
Switch setup commands Boot loader commands	*Catalyst 2960-XR Switch System Management Command Reference*
USB flash devices	*Catalyst 2960-XR Switch Interface and Hardware Component Configuration Guide* *Catalyst 2960-XR Switch Managing Cisco IOS Image Files Configuration Guide*
Hardware installation	*Catalyst 2960-XR Switch Hardware Installation Guide*
Platform-independent command references	*Cisco IOS 15.3M&T Command References*
Platform-independent configuration information	*Cisco IOS 15.3M&T Configuration Guides*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.

To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.

Access to most tools on the Cisco Support website requires a Cisco.com user ID and password. | http://www.cisco.com/support |

Feature History and Information For Performing Switch Setup Configuration

Command History

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Configuring SDM Templates

Information About Configuring SDM Templates

Restrictions for SDM Templates

The following are restrictions when using SDM templates:

The switch supports homogeneous stacking, but does not support mixed stacking.

SDM Templates

You can use Switch Database Management (SDM) templates to configure system resources to optimize support for specific features, depending on how your device is used in the network. You can select a template to provide maximum system usage for some functions: for example, use the default template to balance resources, and use the access template to obtain maximum ACL usage. The switch SDM templates allocate system hardware resources for different uses.

You can select SDM templates for IP Version 4 (IPv4) to optimize these features on switches running the IP Lite feature set:

Note When the switch is running the LAN Base feature set, you must use the default template.

- IPv4—The IPv4 template provides maximum support for IPv4. No IPv6 support is available on this template.

- VLAN—The VLAN template provides Layer 2 functionality and the maximum number of unicast MAC addresses.

· Default—The default template gives balance to all functions providing support for Layer 2, IPv4 and IPv6 functionality.

> **Note** Use the default template when configuring IPv4 static routing on SVIs on switches running the LAN Base feature set. You can configure up to 16 static routes.

For information about homogeneous stacks, see the *Catalyst 2960-XR Switch Stacking Configuration Guide*.

After you change the template and the system reboots, you can use the **show sdm prefer** privileged EXEC command to verify the new template configuration. If you enter the **show sdm prefer** command before you enter the **reload** privileged EXEC command, the **show sdm prefer** command shows the template currently in use and the template that becomes active after a reload.

> **Note** · The SDM templates contain only those commands that are defined as part of the templates. If a template enables another related command that is not defined in the template, then this other command will be visible when the **show running config** command is entered. For example, if the SDM template enables the **switchport voice vlan** command, then the **spanning-tree portfast edge** command may also be enabled (although it is not defined on the SDM template).
>
> If the SDM template is removed, then other such related commands are also removed and have to be reconfigured explicitly.
>
> · SDM templates do not create VLANs. You must create the VLANs before adding commands to the SDM templates.

SDM Template Resources

You can use SDM templates to configure system resources to optimize support for specific features, depending on how your device is used in the network. You can select a template to provide maximum system usage for some functions.

To allocate ternary content addressable memory (TCAM) resources for different usages, the switch SDM templates prioritize system resources to optimize support for certain features. These templates are supported on your device:

Table 128: Approximate Number of Feature Resources Allowed by Templates

Resource	Default	IPv4	VLAN
Unicast MAC addresses	16 K	16 K	32 K
Active VLANs/VLAN IDs	1K/4096	1K/4096	1K/4096
IPv4 IGMP groups	1 K	1 K	1 K
IPv4 unicast routes	5.25 K	24 K	0.5 K

Resource	Default	IPv4	VLAN
• Directly connected hosts	4 K	16 K	0.25 K
• Indirect routes	1.25 K	8 K	0.25 K
IPv4 policy-based routing ACEs	0.5 K	0.375 k	0.5 K
IPv6 multicast groups:	1 K	0	1 K
• Directly connected IPv6 addresses	4 K	0	0.25 K
• Indirect IPv6 unicast routes	1.25 K	0	0.25 K
IPv4 MAC QoS ACEs	0.5 K	0.5 K	0.5 K
IPv4 MAC security ACEs	1 K	0.875 k	1 K
IPv6 policy-based routing ACEs	0.25 K	0	0
IPv6 QoS ACEs	0.25 K	0	0.5 K
IPv6 security ACEs	0.5 K	60	0.5 K

SDM Templates and Switch Stacks

All stack members use the same SDM template that is stored on the active stack. When a new switch is added to a stack, as with the switch configuration and VLAN database files, the SDM configuration that is stored on the active stack overrides the template configured on an individual switch.

Version-mismatch (VM) mode has priority over SDM-mismatch mode. If a VM mode condition and an SDM-mismatch mode exist, the switch stack first attempts to resolve the VM-mode condition. You can use the **show switch** privileged EXEC command to see if any stack members are in SDM-mismatch mode.

How to Configure SDM Templates

Setting the SDM Template

Follow these steps to use the SDM template to maximize feature usage:

SUMMARY STEPS

    1. enable
    2. configure terminal
    3. sdm prefer { default | ipv4 | vlan }
    4. end
    5. reload

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 3	**sdm prefer** { **default** | **ipv4** | **vlan** } **Example:** Switch(config)# **sdm prefer default**	Specifies the SDM template to be used on the switch. The keywords have these meanings: • **default** —The default template provides balance for all Layer 2, IPv4 and IPv6 functionality. • **IPv4** —The IPv4 template provides maximum support for IPv4. No IPv6 support is available on this template. • **VLAN** —The VLAN template provides Layer 2 functionality and the maximum number of unicast MAC addresses. Use the **no sdm prefer** command to set the switch to the default template. The default template balances the use of system resources.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 5	**reload** **Example:** Switch# **reload**	Reloads the operating system.

Configuration Examples for SDM Templates

Examples: Displaying SDM Templates

This is an example output showing the default template information:

```
Switch# show sdm prefer default

"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 directly-connected IPv6 addresses:      4K
  number of indirect IPv6 unicast routes:           1.25K
  number of IPv4 policy based routing aces:         0.5K
  number of IPv4/MAC qos aces:                      0.5K
  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:                     0.5K


Switch#
```

This is an example output showing the IPv4 template information:

```
Switch# show sdm prefer ipv4

 "ipv4" 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 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:                 0.875k
  number of IPv6 policy based routing aces:         0
  number of IPv6 qos aces:                          0
  number of IPv6 security aces:                     60
```

This is an example output showing the VLAN template information:

```
Switch# show sdm prefer vlan

"vlan" 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:                 32K
 number of IPv4 IGMP groups + multicast routes:   1K
 number of IPv4 unicast routes:                   0.5K
   number of directly-connected IPv4 hosts:       0.25K
   number of indirect IPv4 routes:                0.25K
 number of IPv6 multicast groups:                 1K
 number of directly-connected IPv6 addresses:     0.25K
 number of indirect IPv6 unicast routes:          0.25K
 number of IPv4 policy based routing aces:        0.5K
 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:                         0.5K
 number of IPv6 security aces:                    0.5K
```

Examples: Configuring SDM Templates

This example shows how to configure the VLAN template:

```
Switch(config)# sdm prefer vlan
Switch(config)# exit
Switch# reload
  Proceed with reload? [confirm]
```

Additional References for SDM Templates

Related Documents

Related Topic	Document Title
SDM commands	*Catalyst 2960-XR Switch System Management Command Reference*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Configuring SDM Templates

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

CHAPTER 61

Configuring System Message Logging and Smart Logging

Information About Configuring System Message Logs and Smart Logs

System Messsage Logging

By default, a switch sends the output from system messages and **debug** privileged EXEC commands to a logging process. Stack members can trigger system messages. A stack member that generates a system message appends its hostname in the form of hostname-n, where n is a switch range from 1 to 8, and redirects the output to the logging process on the active switchstack's active switch. Though the active switchstack's active switch is a stack member, it does not append its hostname to system messages. 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.

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 active consoles 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. For information on possible messages, see the system message guide for this release.

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 on a standalone switch, and in the case of a switch stack, on the active switchstack's active switch. If a standalone switch or the stack master fails, the log is lost unless you had saved it to flash memory.

You can remotely monitor system messages by viewing the logs on a syslog server or by accessing the switch through Telnet, through the console port, or through the Ethernet management port. In a switch stack, all stack member consoles provide the same console output.

> **Note** The syslog format is compatible with 4.3 BSD UNIX.

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. Depending on the switch, messages appear in one of these formats:

- *seq no:timestamp: %facility-severity-MNEMONIC:description (hostname-n)*

- *seq no:timestamp: %facility-severity-MNEMONIC:description*

The part of the message preceding the percent sign depends on the setting of these global configuration commands:

- **service sequence-numbers**

- **service timestamps log datetime**

- **service timestamps log datetime** [**localtime**] [**msec**] [**show-timezone**]

- **service timestamps log uptime**

Table 129: 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.
timestamp formats: *mm/dd h h: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.
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.

Element	Description
hostname-n	Hostname of a stack member and its switch number in the stack. Though the active switchstack's active switch is a stack member, it does *not* append its hostname to system messages.

Default System Message Logging Settings

Table 130: Default System Message Logging Settings

Feature	Default Setting
System message logging to the console	Enabled.
Console severity	Debugging.
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.
Server facility	Local7
Server severity	Informational.

Syslog Message Limits

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.

The history table lists the level keywords and severity level. For SNMP usage, the severity level values increase by 1. For example, *emergencies* equal 1, not 0, and *critical* equals 3, not 2.

Smart Logging

Smart logging provides a mechanism to capture and export packet flows based on predefined or user-configured triggers. The switch supports smart logging for these events:

- DHCP snooping violations

- Dynamic ARP inspection violations

- IP source guard denied traffic

- ACL permitted or denied traffic

To use smart logging, you must first configure a NetFlow exporter that you identify when you enable smart logging. For information on configuring the NetFlow feature, see the *Catalyst 2960-XR Switch NetFlow Lite Configuration Guide*.

Smart logging processing creates a NetFlow packet for the configured event and sends the packet to the external NetFlow collector. Smart logging counters reflect the number of packets that are logged. This number is the same as the number of packets sent to the collector if no packets are dropped between the switch and the NetFlow collector. You enable smart logging globally on the switch, and you can then configure specific events to be smart logged.

Smart Logging for Port ACL Deny or Permit Actions

The switch supports port ACLs, router ACLs, and VLAN ACLs.

- Port ACLs are IP or MAC ACLs applied to a Layer 2 port. Logging is not supported on port ACLs, but smart logging is supported on IP ACLs applied to Layer 2 ports.

- Router ACLs are ACLs applied to Layer 3 ports. Router ACLs support logging but not smart logging.

- VLAN ACLs or VLAN maps are ACLs applied to VLANs. You can configure logging on VLAN maps, but not smart logging.

When you configure any permit or deny ACL, you can configure logging or smart logging as part of the access list, to take place on all traffic that the ACL permits or denies. The type of port that you attach the ACL to determines the type of logging. If you attach an ACL with smart log configured to a router or a VLAN, the ACL is attached, but smart logging does not take affect. If you configure logging on an ACL attached to a Layer 2 port, the logging keyword is ignored.

How to Configure System Message Logs and Smart Logs

Setting the Message Display Destination Device

If message logging is enabled, you can send messages to specific locations in addition to the console.

This task is optional.

SUMMARY STEPS

1. **configure terminal**

    2. **logging buffered** [*size*]

    3. **logging** *host*

    4. **logging file flash:** *filename* [*max-file-size* [*min-file-size*]] [*severity-level-number* | *type*]

    5. **end**

    6. **terminal monitor**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**logging buffered** [*size*] **Example:** Switch(config)# **logging buffered 8192**	Logs messages to an internal buffer on the switch or on a standalone switch or, in the case of a switch stack, on the active switch. The range is 4096 to 2147483647 bytes. The default buffer size is 4096 bytes. If a standalone switch or the active switch fails, the log file is lost unless you 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.
Step 3	**logging** *host* **Example:** Switch(config)# **logging 125.1.1.100**	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.
Step 4	**logging file flash:** *filename* [*max-file-size* [*min-file-size*]] [*severity-level-number*	*type*] **Example:** Switch(config)# **logging file flash:log_msg.txt 40960 4096 3**	Stores log messages in a file in flash memory on a standalone switch or, in the case of a switch stack, on the active switch. • *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.

	Command or Action	**Purpose**
		• (Optional) *severity-level-number*	*type*—Specifies either the logging severity level or the logging type. The severity range is 0 to 7.
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 6	**terminal monitor** **Example:** Switch# **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

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.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **line** [**console** | **vty**] *line-number* [*ending-line-number*]
3. **logging synchronous** [**level** [*severity-level* | **all**] | **limit** *number-of-buffers*]
4. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**line** [**console** | **vty**] *line-number* [*ending-line-number*] **Example:** Switch(config)# **line console**	Specifies the line to be configured for synchronous logging of messages. • **console** —Specifies configurations that occur through the switch console port or the Ethernet management port. • **line vty** *line-number*—Specifies 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 You can also 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.
Step 3	**logging synchronous** [**level** [*severity-level* | **all**] | **limit** *number-of-buffers*] **Example:** Switch(config)# **logging synchronous level 3 limit 1000**	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.
Step 4	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

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.

The **logging synchronous** global configuration command also affects the display of messages to the console. When this command is enabled, messages appear only after you press **Return**.

To reenable message logging after it has been disabled, use the **logging on** global configuration command.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **no logging console**
3. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**no logging console** **Example:** Switch(config)# **no logging console**	Disables message logging.
Step 3	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Enabling and Disabling Time Stamps on Log Messages

By default, log messages are not time-stamped.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. Use one of these commands:

   - **service timestamps log uptime**
   - **service timestamps log datetime**[**msec** | **localtime** | **show-timezone**]

3. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.

Step 2	Use one of these commands:
	
	  • **service timestamps log uptime**
	  • **service timestamps log datetime**[**msec** | **localtime** | **show-timezone**]
	
	Example:
	Switch(config)# **service timestamps log uptime**
	
	or
	
	Switch(config)# **service timestamps log datetime**
		Enables log time stamps.
		
		• **log uptime**—Enables time stamps on log messages, showing the time since the system was rebooted.
		
		• **log datetime**—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.

Step 3	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Enabling and Disabling Sequence Numbers in Log Messages

If there is more than one log message with the same time stamp, you can display messages with sequence numbers to view these messages. By default, sequence numbers in log messages are not displayed.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **service sequence-numbers**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 2	**service sequence-numbers** Example: Switch(config)# **service sequence-numbers**	Enables sequence numbers.
Step 3	**end** Example: Switch(config)# **end**	Returns to privileged EXEC mode.

Defining the Message Severity Level

Limit messages displayed to the selected device by specifying the severity level of the message.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **logging console** *level*
3. **logging monitor** *level*
4. **logging trap** *level*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 2	**logging console** *level* Example: Switch(config)# **logging console 3**	Limits messages logged to the console. By default, the console receives debugging messages and numerically lower levels.

		Command or Action	**Purpose**
Step 3		**logging monitor** *level* **Example:** Switch(config)# **logging monitor 3**	Limits messages logged to the terminal lines. By default, the terminal receives debugging messages and numerically lower levels.
Step 4		**logging trap** *level* **Example:** Switch(config)# **logging trap 3**	Limits messages logged to the syslog servers. By default, syslog servers receive informational messages and numerically lower levels.
Step 5		**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Limiting Syslog Messages Sent to the History Table and to SNMP

This task explains how to limit syslog messages that are sent to the history table and to SNMP.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **logging history** *level*
3. **logging history size** *number*
4. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**logging history** *level* **Example:** Switch(config)# **logging history 3**	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.

	Command or Action	**Purpose**
Step 3	**logging history size** *number*	Specifies the number of syslog messages that can be stored in the history table.
	Example:	The default is to store one message. The range is 0 to 500 messages.
	Switch(config)# **logging history size 200**	
Step 4	**end**	Returns to privileged EXEC mode.
	Example:	
	Switch(config)# **end**	

Logging Messages to a UNIX Syslog Daemon

This task 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.

Before you begin

- Log in as root.

- Before you can send system log messages to a UNIX syslog server, you must configure the syslog daemon on a UNIX server.

SUMMARY STEPS

1. Add a line to the file /etc/syslog.conf.
2. Enter these commands at the UNIX shell prompt.
3. Make sure the syslog daemon reads the new changes.

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	Add a line to the file /etc/syslog.conf.	• **local7**—Specifies the logging facility.
	Example:	• **debug**—Specifies the syslog level. The file must already exist, and the syslog daemon must have permission to write to it.
	local7.debug /usr/adm/logs/*cisco.log*	
Step 2	Enter these commands at the UNIX shell prompt.	Creates the log file. The syslog daemon sends messages at this level or at a more severe level to this file.
	Example:	

	Command or Action	Purpose
	`$ touch /var/log/`*`cisco.log`* `$ chmod 666 /var/log/`*`cisco.log`*	
Step 3	Make sure the syslog daemon reads the new changes. **Example:** `$ kill -HUP `\`cat /etc/syslog.pid\`	For more information, see the **man syslog.conf** and **man syslogd** commands on your UNIX system.

Enabling Smart Logging

SUMMARY STEPS

1. **configure terminal**
2. **logging smartlog**
3. **logging smartlog exporter** *exporter_name*
4. **logging packet capture size** *packet_size*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**logging smartlog** **Example:** Switch(config)#**logging smartlog**	Turns on the smart logging feature.
Step 3	**logging smartlog exporter** *exporter_name* **Example:** Switch(config)# **logging smartlog exporter export-file**	Identifies the smart log exporter. You must have already configured the exporter by using the NetFlow CLI. If the exporter name does not exist, you receive an error message. By default, the switch sends data to the collector every 60 seconds.

	Command or Action	**Purpose**
Step 4	**logging packet capture size** *packet_size* **Example:** Switch(config)# **logging packet capture size 128**	(Optional) Configures the size of the packet to be sent to the exporter. The range is from 64 to 1024 bytes in 4-byte increments. The default size is 64 bytes. **Note** Increasing the packet capture size reduces the number of flow records per packet.
Step 5	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Enabling Smart Logging for DHCP Snooping Violations

DHCP snooping intercepts and inspects DHCP packets entering untrusted ports and either forwards or drops the packets. You can enable DHCP snooping smart logging to send the contents of dropped packets to the NetFlow collector.

SUMMARY STEPS

1. **configure terminal**
2. **ip dhcp snooping vlan** {*vlan-id* | *vlan-range* }**smartlog**
3. **end**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**ip dhcp snooping vlan** {*vlan-id*	*vlan-range* }**smartlog** **Example:** Switch(config)#**ip dhcp snooping vlan 5-8 smartlog**	Specifies a VLAN ID or a range of VLANs on which to enable DHCP snooping smart logging.
Step 3	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Enabling Smart Logging for Dynamic ARP Inspection Violations

Dynamic ARP inspection intercepts ARP packets on untrusted ports and validates them before forwarding. The functionality is similar to DHCP snooping but for ARP packets. You can configure dynamic ARP inspection logging by using the **ip arp inspection log-buffer** global configuration command. By default, all dropped packets are logged. You can also configure the switch to apply smart logging to the same packets that are being logged, sending the packet contents packet to the Cisco NetFlow collector.

SUMMARY STEPS

1. **configure terminal**
2. **ip arp inspection smartlog**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**ip arp inspection smartlog** **Example:** Switch(config)# **ip arp inspection smartlog**	Specifies that whatever packets are currently being logged (the default is all dropped packets) are also smart-logged.
Step 3	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.

Enabling Smart Logging for IP Source Guard Violations

IP source guard is a security feature related to DHCP snooping. You can use IP source guard to filter traffic based on the IP source address or the MAC address. All IP packets with a source address other than the specified address or addresses learned through DHCP snooping are denied. You can enable IP source guard smart logging to send the contents of the denied packets to the NetFlow collector.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip verify source smartlog**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** Switch(config)#**interface GigabitEthernet1/0/1**	Specifies an interface and enters interface configuration mode.
Step 3	**ip verify source smartlog** **Example:** Switch(config-if)# **ip verify source smartlog**	Enables IP source guard smart logging for all packets that are denied by IP source guard.
Step 4	**end** **Example:** Switch(config-if)# **end**	Returns to privileged EXEC mode.

Monitoring and Maintaining System Message Logs and Smart Logs

Monitoring Configuration Archive Logs

Command	Purpose
show archive log config {**all**	**number** [*end-number*]	**user** *username* [**session** *number*] *number* [*end-number*]	**statistics**} [**provisioning**]	Displays the entire configuration log or the log for specified parameters.

Monitoring Smart Logging

Command	Purpose
show logging smartlog	Displays smart logging entries.

Command	Purpose
show ip arp inspection	Displays the IP ARP smart logging entries.
show ip verify source	Displays IP source guard smart logging entries. The output shows whether or not smart logging is enabled on the interface.

Configuration Examples for System Message Logs and Smart Logs

Example: Stacking System Message

This example shows a partial switch system message for active stack and a stack member (hostname *Switch-2*):

```
00:00:46: %LINK-3-UPDOWN: Interface Port-channel1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet1/0/1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet1/0/2, 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 GigabitEthernet1/0/1, 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)

00:00:46: %LINK-3-UPDOWN: Interface Port-channel1, changed state to up (Switch-2)
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet2/0/1, changed state to up (Switch-2)
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet2/0/2, changed state to up (Switch-2)
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down
(Switch-2)
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet2/0/1, changed
state to down 2 (Switch-2)
```

Example: Switch System Message

This example shows a partial switch system message on a switch:

```
00:00:46: %LINK-3-UPDOWN: Interface Port-channel1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet0/1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet0/2, 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 GigabitEthernet0/1, 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)
```

Example: Enabling Smart Logging

You add the smart log configuration option when you create the permit and deny conditions for an ACL.

This example shows how to enable smart logging on a numbered access list:

```
Switch(config)#  access-list 199 permit ip any any smartlog
```

This example shows how to enable smart logging on a named access list:

```
Switch(config)#  ip access-list extended test1
Switch(config-ext-nacl)# deny ip host 10.1.1.3 any smartlog
```

Examples: Displaying Service Timestamps Log

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)
(Switch-2)
```

This example shows part of a logging display with the **service timestamps log uptime** global configuration command enabled:

```
00:00:46: %LINK-3-UPDOWN: Interface Port-channel1, changed state to up (Switch-2)
```

This example shows part of a logging display with the sequence numbers enabled.

```
000019: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36) (Switch-2)
```

Additional References for System Message Logs and Smart Logs

Related Documents

Related Topic	Document Title
System message log commands	*Catalyst 2960-XR Switch System Management Command Reference*

Related Topic	Document Title
Platform-independent command references	*Cisco IOS 15.3M&T Command References*
Platform-independent configuration information	*Cisco IOS 15.3M&T Configuration Guides*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information For System Message Logs

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

C H A P T E R 62

Configuring Online Diagnostics

Information About Configuring Online Diagnostics

Online Diagnostics

With online diagnostics, you can test and verify the hardware functionality of the Switch while the Switch is connected to a live network.

The online diagnostics contain packet switching tests that check different hardware components and verify the data path and the control signals.

The online diagnostics detect problems in these areas:

- Hardware components

- Interfaces (Ethernet ports and so forth)

- Solder joints

Online diagnostics are categorized as on-demand, scheduled, or health-monitoring diagnostics. On-demand diagnostics run from the CLI; scheduled diagnostics run at user-designated intervals or at specified times when the Switch is connected to a live network; and health-monitoring runs in the background with user-defined intervals. By default, the health-monitoring test runs for every 30 seconds.

After you configure online diagnostics, you can manually start diagnostic tests or display the test results. You can also see which tests are configured for the Switch or switch stack and the diagnostic tests that have already run.

How to Configure Online Diagnostics

Starting Online Diagnostic Tests

After you configure diagnostic tests to run on the switch, use the **diagnostic start** privileged EXEC command to begin diagnostic testing.

After starting the tests, you cannot stop the testing process.

Use this privileged EXEC command to manually start online diagnostic testing.

SUMMARY STEPS

1. **diagnostic start switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all** | **basic** | **non-disruptive** }

DETAILED STEPS

	Command or Action	Purpose
Step 1	**diagnostic start switch** *number* **test** {*name*	*test-id*	*test-id-range*	**all**	**basic**	**non-disruptive** } **Example:** `Switch# diagnostic start switch 2 test basic`	Starts the diagnostic tests. The **switch** *number* keyword is supported only on stacking switches. The range is from 1 to 8. You can specify the tests by using one of these options: • *name*—Enters the name of the test. • *test-id*—Enters the ID number of the test. • *test-id-range*—Enters the range of test IDs by using integers separated by a comma and a hyphen. • **all**—Starts all of the tests. • **basic**— Starts the basic test suite. • **non-disruptive**—Starts the non-disruptive test suite.

Configuring Online Diagnostics

You must configure the failure threshold and the interval between tests before enabling diagnostic monitoring.

Scheduling Online Diagnostics

You can schedule online diagnostics to run at a designated time of day or on a daily, weekly, or monthly basis for a switch. Use the **no** form of this command to remove the scheduling.

SUMMARY STEPS

1. **configure terminal**

2. **diagnostic schedule switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all** | **basic** | **non-disruptive** |} {**daily** | **on** *mm dd yyyy hh:mm* | **weekly** *day-of-week hh:mm*}

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal** **Example:** Switch# **configure terminal**	Enters global configuration mode.
Step 2	**diagnostic schedule switch** *number* **test** {*name*	*test-id*	*test-id-range*	**all**	**basic**	**non-disruptive** |} {**daily**	**on** *mm dd yyyy hh:mm*	**weekly** *day-of-week hh:mm*} **Example:** Switch(config)# **diagnostic schedule switch 1 test 1-5 on July 3 2013 23:10**	Schedules on-demand diagnostic tests for a specific day and time. The **switch** *number* keyword is supported only on stacking switches. The range is from 1 to 8. When specifying the tests to be scheduled, use these options: • *name*—Name of the test that appears in the **show diagnostic content** command output. • *test-id*—ID number of the test that appears in the **show diagnostic content** command output. • *test-id-range*—ID numbers of the tests that appear in the **show diagnostic content** command output. • **all**—All test IDs. • **basic**—Starts the basic on-demand diagnostic tests. • **non-disruptive**—Starts the non-disruptive test suite. You can schedule the tests as follows: • Daily—Use the **daily** *hh:mm* parameter. • Specific day and time—Use the **on** *mm dd yyyy hh:mm* parameter. • Weekly—Use the **weekly** *day-of-week hh:mm* parameter.

Configuring Health-Monitoring Diagnostics

You can configure health-monitoring diagnostic testing on a Switch while it is connected to a live network. You can configure the execution interval for each health-monitoring test, enable the Switch to generate a syslog message because of a test failure, and enable a specific test.

Use the **no** form of this command to disable testing.

By default, health monitoring is disabled, but the Switch generates a syslog message when a test fails.

Follow these steps to configure and enable the health-monitoring diagnostic tests:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **diagnostic monitor interval switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all**} *hh:mm:ss milliseconds day*
4. **diagnostic monitor syslog**
5. **diagnostic monitor threshold switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all**} **failure count** *count*
6. **diagnostic monitor switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all**}
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** Example: Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** Example: Switch# **configure terminal**	Enters global configuration mode.
Step 3	**diagnostic monitor interval switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all**} *hh:mm:ss milliseconds day* Example: Switch(config)# **diagnostic monitor interval switch 2 test 1 12:30:00 750 5**	Configures the health-monitoring interval of the specified tests. The **switch** *number* keyword is supported only on stacking switches. When specifying the tests, use one of these parameters: • *name*—Name of the test that appears in the **show diagnostic content** command output. • *test-id*—ID number of the test that appears in the **show diagnostic content** command output. • *test-id-range*—ID numbers of the tests that appear in the **show diagnostic content** command output. • **all**—All of the diagnostic tests. When specifying the interval, set these parameters:

	Command or Action	**Purpose**
		• *hh:mm:ss*—Monitoring interval in hours, minutes, and seconds. The range for *hh* is 0 to 24, and the range for *mm* and *ss* is 0 to 60.
		• *milliseconds*—Monitoring interval in milliseconds (ms). The range is from 0 to 999.
		• *day*—Monitoring interval in the number of days. The range is from 0 to 20.
Step 4	**diagnostic monitor syslog** **Example:** Switch(config)# **diagnostic monitor syslog**	(Optional) Configures the switch to generate a syslog message when a health-monitoring test fails.
Step 5	**diagnostic monitor threshold switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all**} **failure count** *count* **Example:** Switch(config)# **diagnostic monitor threshold switch 2 test 1 failure count 20**	(Optional) Sets the failure threshold for the health-monitoring tests. The **switch** *number* keyword is supported only on stacking switches. The range is from 1 to 8. When specifying the tests, use one of these parameters: • *name*—Name of the test that appears in the **show diagnostic content** command output. • *test-id*—ID number of the test that appears in the **show diagnostic content** command output. • *test-id-range*—ID numbers of the tests that appear in the **show diagnostic content** command output. • **all**—All of the diagnostic tests. The range for the failure threshold *count* is 0 to 99.
Step 6	**diagnostic monitor switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all**} **Example:** Switch(config)# **diagnostic monitor switch 2 test 1**	Enables the specified health-monitoring tests. The **switch** *number* keyword is supported only on stacking switches. The range is from 1 to 9. When specifying the tests, use one of these parameters: • *name*—Name of the test that appears in the **show diagnostic content** command output. • *test-id*—ID number of the test that appears in the **show diagnostic content** command output. • *test-id-range*—ID numbers of the tests that appear in the **show diagnostic content** command output. • **all**—All of the diagnostic tests.

	Command or Action	**Purpose**
Step 7	**end** **Example:** Switch(config)# **end**	Returns to privileged EXEC mode.
Step 8	**show running-config** **Example:** Switch# **show running-config**	Verifies your entries.
Step 9	**copy running-config startup-config** **Example:** Switch# **copy running-config startup-config**	(Optional) Saves your entries in the configuration file.

What to do next

Use the **no diagnostic monitor interval test**test-id | test-id-range } global configuration command to change the interval to the default value or to zero. Use the **no diagnostic monitor syslog** command to disable generation of syslog messages when a health-monitoring test fails. Use the **diagnostic monitor threshold test**test-id | test-id-range }**failure count**command to remove the failure threshold.

Monitoring and Maintaining Online Diagnostics

Displaying Online Diagnostic Tests and Test Results

You can display the online diagnostic tests that are configured for the Switch or Switch stack and check the test results by using the privileged EXEC **show** commands in this table:

Table 131: Commands for Diagnostic Test Configuration and Results

Command	**Purpose**
show diagnostic content switch [number | **all**]	Displays the online diagnostics configured for a switch. The **switch** [number | **all**] parameter is supported only on stacking switches.
show diagnostic status	Displays the currently running diagnostic tests.
show diagnostic result switch [number | **all**] [**detail** | **test** {name | test-id | test-id-range | **all**} [**detail**]]	Displays the online diagnostics test results. The **switch** [number | **all**] parameter is supported only on stacking switches.

Command	Purpose
show diagnostic switch [*number* | **all**] [**detail**]	Displays the online diagnostics test results. The **switch** [*number* | **all**] parameter is supported only on stacking switches.
show diagnostic schedule switch [*number* | **all**]	Displays the online diagnostics test schedule. The **switch** [*number* | **all**] parameter is supported only on stacking switches.
show diagnostic post	Displays the POST results. (The output is the same as the **show post** command output.)

Configuration Examples for Online Diagnostic Tests

Starting Online Diagnostic Tests

After you configure diagnostic tests to run on the switch, use the **diagnostic start** privileged EXEC command to begin diagnostic testing.

After starting the tests, you cannot stop the testing process.

Use this privileged EXEC command to manually start online diagnostic testing.

SUMMARY STEPS

1. **diagnostic start switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all** | **basic** | **non-disruptive** }

DETAILED STEPS

	Command or Action	Purpose
Step 1	**diagnostic start switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all** | **basic** | **non-disruptive** } **Example:** `Switch# diagnostic start switch 2 test basic`	Starts the diagnostic tests. The **switch** *number* keyword is supported only on stacking switches. The range is from 1 to 8. You can specify the tests by using one of these options: • *name*—Enters the name of the test. • *test-id*—Enters the ID number of the test. • *test-id-range*—Enters the range of test IDs by using integers separated by a comma and a hyphen. • **all**—Starts all of the tests. • **basic**— Starts the basic test suite. • **non-disruptive**—Starts the non-disruptive test suite.

Example: Configure a Health Monitoring Test

This example shows how to configure a health-monitoring test:

```
Switch(config)# diagnostic monitor threshold switch 1 test 1 failure count 50
Switch(config)# diagnostic monitor interval switch 1 test TestPortAsicStackPortLoopback
```

Examples: Schedule Diagnostic Test

This example shows how to schedule diagnostic testing for a specific day and time on a specific switch:

```
Switch(config)# diagnostic schedule test DiagThermalTest on June 3 2013  22:25
```

This example shows how to schedule diagnostic testing to occur weekly at a certain time on a specific switch:

```
Switch(config)# diagnostic schedule switch 1 test 1,2,4-6 weekly saturday 10:30
```

Displaying Online Diagnostics: Examples

This example shows how to display the online diagnostic detailed information on a specific switch:

```
Switch# show diagnostic switch 1 detail

Switch 1: SerialNo :

Overall Diagnostic Result for Switch 1 : UNTESTED

Test results: (. = Pass, F = Fail, U = Untested)

_____

1) TestPortAsicStackPortLoopback ---> U

Error code ------------------> 3 (DIAG_SKIPPED)
Total run count -------------> 0
Last test testing type ------> n/a
Last test execution time ----> n/a
First test failure time -----> n/a
Last test failure time ------> n/a
Last test pass time ---------> n/a
Total failure count ---------> 0
Consecutive failure count ---> 0
_____

2) TestPortAsicLoopback ------------> U

Error code ------------------> 3 (DIAG_SKIPPED)
Total run count -------------> 0
Last test testing type ------> n/a
Last test execution time ----> n/a
First test failure time -----> n/a
Last test failure time ------> n/a
Last test pass time ---------> n/a
```

```
Total failure count ---------> 0
Consecutive failure count ---> 0
_____

3) TestPortAsicCam -----------------> U

Error code ------------------> 3 (DIAG_SKIPPED)
Total run count -------------> 0
Last test testing type ------> n/a
Last test execution time ----> n/a
First test failure time -----> n/a
Last test failure time ------> n/a
Last test pass time ---------> n/a
Total failure count ---------> 0
Consecutive failure count ---> 0
_____

4) TestPortAsicMem -----------------> U

Error code ------------------> 3 (DIAG_SKIPPED)
Total run count -------------> 0
Last test testing type ------> n/a
Last test execution time ----> n/a
First test failure time -----> n/a
Last test failure time ------> n/a
Last test pass time ---------> n/a
Total failure count ---------> 0
Consecutive failure count ---> 0
_____

5) TestInlinePwrCtlr ---------------> U

Error code ------------------> 3 (DIAG_SKIPPED)
Total run count -------------> 0
Last test testing type ------> n/a
Last test execution time ----> n/a
First test failure time -----> n/a
Last test failure time ------> n/a
Last test pass time ---------> n/a
Total failure count ---------> 0
Consecutive failure count ---> 0
_____
```

This example shows how to display the online diagnostics that are configured on a specific switch:

```
Switch# show diagnostic content switch 3

Switch 1:
Diagnostics test suite attributes:
    B/* - Basic ondemand test / NA
  P/V/* - Per port test / Per device test / NA
  D/N/* - Disruptive test / Non-disruptive test / NA
    S/* - Only applicable to standby unit / NA
    X/* - Not a health monitoring test / NA
    F/* - Fixed monitoring interval test / NA
    E/* - Always enabled monitoring test / NA
    A/I - Monitoring is active / Monitoring is inactive
    R/* - Switch will reload after test list completion / NA
    P/* - will partition stack / NA

                                                Test Interval    Thre-
```

```
ID   Test Name                            Attributes     day hh:mm:ss.ms shold
==== ================================== ============    =============== =====
  1) TestPortAsicStackPortLoopback --->  B*N****I**      not configured  n/a
  2) TestPortAsicLoopback ------------>  B*D*X**IR*      not configured  n/a
  3) TestPortAsicCam ----------------->  B*D*X**IR*      not configured  n/a
  4) TestPortAsicRingLoopback -------->  B*D*X**IR*      not configured  n/a
  5) TestMicRingLoopback ------------->  B*D*X**IR*      not configured  n/a
  6) TestPortAsicMem ----------------->  B*D*X**IR*      not configured  n/a
```

This example shows how to display the online diagnostic results for a switch:

```
Switch# show diagnostic result

Switch 1: SerialNo :
Overall diagnostic result: PASS
Test results: (. = Pass, F = Fail, U = Untested)
1) TestPortAsicStackPortLoopback ---> .
2) TestPortAsicLoopback ------------> .
3) TestPortAsicCam -----------------> .
4) TestPortAsicRingLoopback --------> .
5) TestMicRingLoopback -------------> .
6) TestPortAsicMem -----------------> .
```

This example shows how to display the online diagnostic test status:

```
Switch# show diagnostic status

<BU> - Bootup Diagnostics, <HM> - Health Monitoring Diagnostics,
<OD> - OnDemand Diagnostics, <SCH> - Scheduled Diagnostics
====== ================================== ============================== ======
Card   Description                        Current Running Test           Run by
------ ---------------------------------- --------------------           ------
1                                         N/A                            N/A
2                                         TestPortAsicStackPortLoopback  <OD>
                                          TestPortAsicLoopback           <OD>
                                          TestPortAsicCam                <OD>
                                          TestPortAsicRingLoopback       <OD>
                                          TestMicRingLoopback            <OD>
                                          TestPortAsicMem                <OD>
3                                         N/A                            N/A
4                                         N/A                            N/A
====== ================================== ============================== ======
Switch#
```

This example shows how to display the online diagnostic test schedule for a switch:

```
Switch# show diagnostic schedule switch 1

Current Time = 14:39:49 PST Tue May 5 2013
Diagnostic for Switch 1:
Schedule #1:
To be run daily 12:00
Test ID(s) to be executed: 1.
```

Additional References for Online Diagnostics

Related Documents

Related Topic	Document Title
Online diagnostics commands	*Catalyst 2960-XR Switch System Management Command Reference*
Platform-independent command references	*Cisco IOS 15.3M&T Command References*
Platform-independent configuration information	*Cisco IOS 15.3M&T Configuration Guides*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Configuring Online Diagnostics

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

C H A P T E R 63

Troubleshooting the Software Configuration

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), Device Manager, or Network Assistant to identify and solve problems.

Additional troubleshooting information, such as LED descriptions, is provided in the hardware installation guide.

Information About Troubleshooting the Software Configuration

Software Failure on a Switch

Switch software can be corrupted during an upgrade by downloading the incorrect 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.

Lost or Forgotten Password on a Switch

The default configuration for the switch allows an end user with physical access to the switch to recover from a lost password by interrupting the boot process during power-on and by entering a new password. These recovery procedures require that you have physical access to the switch.

Note On these switches, a system administrator can disable some of the functionality of this feature by allowing an end user to reset a password only by agreeing to return to the default configuration. If you are an end user trying to reset a password when password recovery has been disabled, a status message reminds you to return to the default configuration during the recovery process.

Power over Ethernet Ports

A Power over Ethernet (PoE) switch port automatically supplies power to one of these connected devices if the switch detects that there is no power on the circuit:

- a Cisco pre-standard powered device (such as a Cisco IP Phone or a Cisco Aironet Access Point)

- an IEEE 802.3af-compliant powered device

- an IEEE 802.3at-compliant powered device

A powered device can receive redundant power when it is connected to a PoE switch port and to an AC power source. The device does not receive redundant power when it is only connected to the PoE port.

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

For more information, see the "Configuring PoE" chapter in the *Catalyst 2960-XR Switch Interface and Hardware Component Configuration Guide*.

Disabled Port Caused by Power Loss

If a powered device (such as a Cisco IP Phone 7910) that is connected to a PoE Switch port and powered by an AC power source loses power from the AC power source, the device might enter an error-disabled state. To recover from an error-disabled state, enter the **shutdown** interface configuration command, and then enter the **no shutdown** interface command. You can also configure automatic recovery on the Switch to recover from the error-disabled state.

On a Switch, the **errdisable recovery cause loopback** and the **errdisable recovery interval** *seconds* global configuration commands automatically take the interface out of the error-disabled state after the specified period of time.

Monitoring PoE Port Status

- **show controllers power inline** privileged EXEC command

- **show power inline** EXEC command

- **debug ilpower** privileged EXEC command

Disabled Port Caused by False Link-Up

If a Cisco powered device is connected to a port and you configure the port by using the **power inline never** interface configuration command, a false link-up can occur, placing the port into an error-disabled state. To take the port out of the error-disabled state, enter the **shutdown** and the **no shutdown** interface configuration commands.

You should not connect a Cisco powered device to a port that has been configured with the **power inline never** command.

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. Traceroute finds the path by using the MAC address tables of the Switch 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 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.

- A Switch is reachable from another Switch when you can test connectivity by using the **ping** privileged EXEC command. All Switch 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 Switch 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.

- This feature is not supported in Token Ring VLANs.

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 Switch 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 Switch 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.

Time Domain Reflector Guidelines

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 10/100/1000 copper Ethernet ports. It is not supported on 10-Gigabit Ethernet ports and 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

When you run TDR, the Switch reports accurate information in these situations:

- The cable for the gigabit link is a solid-core cable.

- The open-ended cable is not terminated.

When you run TDR, the Switch does not report accurate information in these situations:

- The cable for the gigabit link is a twisted-pair cable or is in series with a solid-core cable.

- The link is a 10-megabit or a 100-megabit link.

- The cable is a stranded cable.

- The link partner is a Cisco IP Phone.

- The link partner is not IEEE 802.3 compliant.

Debug Commands

⚠️

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.

Onboard Failure Logging on the Switch

You can use the onboard failure logging (OBFL) feature to collect information about the Switch. The information includes uptime, temperature, and voltage information and helps Cisco technical support representatives to troubleshoot Switch problems. We recommend that you keep OBFL enabled and do not erase the data stored in the flash memory.

By default, OBFL is enabled. It collects information about the Switch and small form-factor pluggable (SFP) modules. The Switch stores this information in the flash memory:

- CLI commands—Record of the OBFL CLI commands that are entered on a standalone Switch or a switch stack member.

- Environment data—Unique device identifier (UDI) information for a standalone Switch or a switch stack member and for all the connected FRU devices: the product identification (PID), the version identification (VID), and the serial number.

- Message—Record of the hardware-related system messages generated by a standalone Switch or a switch stack member.

- Power over Ethernet (PoE)—Record of the power consumption of PoE ports on a standalone Switch or a switch stack member.

- Temperature—Temperature of a standalone Switch or a switch stack member.

- Uptime data—Time when a standalone Switch or a switch stack member starts, the reason the Switch restarts, and the length of time the Switch has been running since it last restarted.

- Voltage—System voltages of a standalone Switch or a switch stack member.

You should manually set the system clock or configure it by using Network Time Protocol (NTP).

When the Switch is running, you can retrieve the OBFL data by using the **show logging onboard** privileged EXEC commands. If the Switch fails, contact your Cisco technical support representative to find out how to retrieve the data.

When an OBFL-enabled Switch is restarted, there is a 10-minute delay before logging of new data begins.

Possible Symptoms of High CPU Utilization

Excessive CPU utilization might result in these symptoms, but the symptoms might 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

Layer 3 switches:

- Dropped packets or increased latency for packets routed in software

- BGP or OSPF routing topology changes

- HSRP flapping

How to Troubleshoot the Software Configuration

Recovering from a Software Failure

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.

Step 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.

Step 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. If you are using Windows, use a zip program that can read a tar file. Use the zip program to navigate. If you are using UNIX, follow these steps:

a) Display the contents of the tar file by using the **tar -tvf** *<image_filename.tar>* UNIX command.

Example:
```
unix-1% tar -tvf image_filename.tar
```

b) Locate the bin file, and extract it by using the **tar -xvf** *<image_filename.tar> <image_filename.bin>* UNIX command.

Example:
```
unix-1% tar -xvf image_filename.tar image_filename.bin
x c2960x-universalk9-mz-150-2.EX1/c2960x-universalk9-mz-150-2.EX1.bin, 2928176 bytes, 5720
tape blocks
```

c) Verify that the bin file was extracted by using the **ls -l** *<image_filename.bin>* UNIX command.

Example:
```
unix-1% ls -l image_filename.bin
-rw-r--r--   1 boba      2928176 Apr 21 12:01
c2960x-universalk9-mz.150-2.0.66.UCP/c2960x-universalk9-mz.150-2.0.66.UCP.bin
```

Step 3 Connect your PC with terminal-emulation software supporting the Xmodem Protocol to the switch console port.

Step 4   Set the line speed on the emulation software to 9600 baud.

Step 5   Unplug the switch power cord.

Step 6   Press the **Mode** button, and at the same time reconnect the power cord to the switch. You can release the Mode button a second or two after the LED above port 1 goes off. Several lines of information about the software appear along with instructions.

Example:
```
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
```

Step 7   Initialize the flash file system.

Example:
```
switch: flash_init
```

Step 8   If you had set the console port speed to any speed 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.

Step 9   Load any helper files.

Example:
```
switch: load_helper
```

Step 10   Start the file transfer by using the Xmodem Protocol.

Example:
```
switch: copy xmodem: flash:image_filename.bin
```

Step 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.

Step 12   Boot the newly downloaded Cisco IOS image.

Example:
```
switch: boot flash:image_filename.bin
```

Step 13   Use the **archive download-sw** privileged EXEC command to download the software image to the switch or to the switch stack.

Step 14   Use the **reload** privileged EXEC command to restart the switch and to verify that the new software image is operating properly.

Step 15   Delete the **flash**:*image_filename.bin* file from the switch.

Recovering from a Lost or Forgotten Password

The default configuration for the switch allows an end user with physical access to the switch to recover from a lost password by interrupting the boot process during power-on and by entering a new password. These recovery procedures require that you have physical access to the switch.

> ✎
>
> **Note**   On these switches, a system administrator can disable some of the functionality of this feature by allowing an end user to reset a password only by agreeing to return to the default configuration. If you are an end user trying to reset a password when password recovery has been disabled, a status message shows this during the recovery process.

You enable or disable password recovery by using the **service password-recovery** global configuration command.

The switch supports homogeneous stacking, but does not support mixed stacking.

Step 1   Connect a terminal or PC to the switch.

- Connect a terminal or a PC with terminal-emulation software to the switch console port.

  Or

- Connect a PC to the Ethernet management port.

Step 2   Set the line speed on the emulation software to 9600 baud.

Step 3   On a switch, power off the switch.

Step 4   Reconnect the power cord to the switch. Within 15 seconds, press the **Mode** button while the System LED is still flashing green. Continue pressing the **Mode** button until all the system LEDs turn on and remain solid, then release the **Mode** button.

Several lines of information about the software appear with instructions, informing you if the password recovery procedure has been disabled or not.

- If you see a message that begins with this statement:

  ```
  The system has been interrupted prior to initializing the flash file system. The following commands
   will initialize the flash file system
  ```

  proceed to the "Procedure with Password Recovery Enabled" section, and follow the steps.

- If you see a message that begins with this statement:

  ```
  The password-recovery mechanism has been triggered, but is currently disabled.
  ```

  proceed to the "Procedure with Password Recovery Disabled" section, and follow the steps.

Step 5   After recovering the password, reload the switch.

On a switch:

```
Switch> reload
Proceed with reload? [confirm] y
```

Procedure with Password Recovery Enabled

If the password-recovery operation is enabled, this message appears:

```
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
```

Step 1   Initialize the flash file system.

```
Switch: flash_init
```

Step 2   If you had set the console port speed to any number 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.

Step 3   Load any helper files.

```
Switch: load_helper
```

Step 4   Display the contents of flash memory.

```
Switch: dir: flash:
Directory of flash:
   13  drwx         192   Mar 01 2013 22:30:48
c2960x-universalk9-mz-150-2.EX1/c2960x-universalk9-mz-150-2.EX1.bin
   11  -rwx        5825   Mar 01 2013 22:31:59  config.text

16128000 bytes total (10003456 bytes free)
```

Step 5   Rename the configuration file to config.text.old

This file contains the password definition.

```
Switch: rename flash: config.text flash: config.text.old
```

Step 6   Boot up the system.

```
Switch: boot
```

You are prompted to start the setup program. Enter **N** at the prompt.

```
Continue with the configuration dialog?? [yes/no]: No
```

Step 7   At the switch prompt, enter privileged EXEC mode.

```
Switch> enable
Switch#
```

Step 8   Rename the configuration file to its original name.

```
Switch# rename flash: config.text.old flash: config.text
```

Note   Before continuing to Step 9, power on any connected stack members and wait until they have completely initialized. Failure to follow this step can result in a lost configuration depending on how your switch is set up.

Step 9   Copy the configuration file into memory

```
Switch# copy flash: config.text system: running-config
Source filename [config.text]?
Destination filename [running-config]?
```

Press **Return** in response to the confirmation prompts. The configuration file is now reloaded, and you can change the password.

Step 10   Enter global configuration mode.

```
Switch# configure terminal
```

Step 11   Change the password.

```
Switch(config)# enable secret password
```

The secret password can be from 1 to 25 alphanumeric characters, can start with a number, is case sensitive, and allows spaces but ignores leading spaces.

Step 12   Return to privileged EXEC mode.

```
Switch(config)# exit
Switch#
```

Step 13   Write the running configuration to the startup configuration file.

```
Switch# copy running-config startup-config
```

The new password is now in the startup configuration.

Note   This procedure is likely to leave your switch virtual interface in a shutdown state. You can see which interface is in this state by entering the **show running-config** privileged EXEC command. To reenable the interface, enter the **interface vlan** *vlan-id* global configuration command, and specify the VLAN ID of the shutdown interface. With the switch in interface configuration mode, enter the **no shutdown** command.

Step 14   Boot the switch with the *packages.conf* file from flash.

```
Switch: boot flash:packages.conf
```

Step 15   Reload the switch stack.

```
Switch# reload
```

Procedure with Password Recovery Disabled

If the password-recovery mechanism is disabled, this message appears:

```
The password-recovery mechanism has been triggered, but
is currently disabled.  Access to the boot loader prompt
through the password-recovery mechanism is disallowed at
this point.  However, if you agree to let the system be
reset back to the default system configuration, access
to the boot loader prompt can still be allowed.

Would you like to reset the system back to the default configuration (y/n)?
```

⚠️

Caution Returning the Switch to the default configuration results in the loss of all existing configurations. We recommend that you contact your system administrator to verify if there are backup Switch and VLAN configuration files.

• If you enter **n** (no), the normal boot process continues as if the **Mode** button had not been pressed; you cannot access the boot loader prompt, and you cannot enter a new password. You see the message:

```
Press Enter to continue.......
```

• If you enter **y** (yes), the configuration file in flash memory and the VLAN database file are deleted. When the default configuration loads, you can reset the password.

Step 1   Choose to continue with password recovery and delete the existing configuration:

```
Would you like to reset the system back to the default configuration (y/n)? Y
```

Step 2   Display the contents of flash memory:

```
Switch: dir flash:
```

The Switch file system appears.

Step 3   Boot up the system:

```
Switch: boot
```

You are prompted to start the setup program. To continue with password recovery, enter **N** at the prompt:

```
Continue with the configuration dialog? [yes/no]: N
```

Step 4   At the Switch prompt, enter privileged EXEC mode:

```
Switch> enable
```

Step 5   Enter global configuration mode:

```
Switch# configure terminal
```

Step 6   Change the password:

```
Switch(config)# enable secret password
```

The secret password can be from 1 to 25 alphanumeric characters, can start with a number, is case sensitive, and allows spaces but ignores leading spaces.

Step 7   Return to privileged EXEC mode:

```
Switch(config)# exit
Switch#
```

Note   Before continuing to Step 9, power on any connected stack members and wait until they have completely initialized.

Step 8   Write the running configuration to the startup configuration file:

```
Switch# copy running-config startup-config
```

The new password is now in the startup configuration.

Step 9   You must now reconfigure the Switch. If the system administrator has the backup Switch and VLAN configuration files available, you should use those.

Recovering from a Command Switch Failure

This section describes how to recover from a failed command switch. You can configure a redundant command switch group by using the Hot Standby Router Protocol (HSRP).

Note   HSRP is the preferred method for supplying redundancy to a cluster.

Note   This feature is introduced from Cisco IOS Release 15.2(5)E2.

If you have not configured a standby command switch, and your command switch loses power or fails in some other way, management contact with the member switches is lost, and you must install a new command switch. However, connectivity between switches that are still connected is not affected, and the member switches forward packets as usual. You can manage the members as standalone switches through the console port, or, if they have IP addresses, through the other management interfaces.

You can prepare for a command switch failure by assigning an IP address to a member switch or another switch that is command-capable, making a note of the command-switch password, and cabling your cluster to provide redundant connectivity between the member switches and the replacement command switch. These sections describe two solutions for replacing a failed command switch:

- Replacing a Failed Command Switch with a Cluster Member

- Replacing a Failed Command Switch with Another Switch

These recovery procedures require that you have physical access to the switch. For information on command-capable switches, see the release notes.

Replacing a Failed Command Switch with a Cluster Member

To replace a failed command switch with a command-capable member in the same cluster, follow these steps

Step 1   Disconnect the command switch from the member switches, and physically remove it from the cluster.

Step 2   Insert the member switch in place of the failed command switch, and duplicate its connections to the cluster members.

Step 3   Start a CLI session on the new command switch.

You can access the CLI by using the console port or, if an IP address has been assigned to the switch, by using Telnet. For details about using the console port, see *Catalyst 2960-X Switch Hardware Installation Guide*.

Step 4   At the switch prompt, enter privileged EXEC mode.

Example:

```
Switch> enable
Switch#
```

Step 5   Enter the password of the *failed command switch*.

Step 6   Enter global configuration mode.

Example:

```
Switch# configure terminal

Enter configuration commands, one per line. End with CNTL/Z.
```

Step 7   Remove the member switch from the cluster.

Example:

```
Switch(config)# no cluster commander-address
```

Step 8   Return to privileged EXEC mode.

Example:

```
Switch(config)# end
Switch#
```

Step 9   Use the setup program to configure the switch IP information. This program prompts you for IP address information and passwords. From privileged EXEC mode, enter EXEC mode, enter **setup**, and press **Return**.

Example:

```
Switch# setup
```

```
 --- System Configuration Dialog ---
Continue with configuration dialog? [yes/no]: y
At any point you may enter a question mark '?' for help.
Use ctrl-c to abort configuration dialog at any prompt.
Default settings are in square brackets '[]'.
Basic management setup configures only enough connectivity
for management of the system, extended setup will ask you
to configure each interface on the system
Would you like to enter basic management setup? [yes/no]:
```

Step 10   Enter **Y** at the first prompt.

Example:

```
The prompts in the setup program vary depending on the member switch that you selected to be the
command switch:
Continue with configuration dialog? [yes/no]: y

or

Configuring global parameters:
```

If this prompt does not appear, enter **enable**, and press **Return**. Enter **setup**, and press **Return** to start the setup program.

Step 11   Respond to the questions in the setup program.

When prompted for the hostname, it is limited to 28 characters and 31 characters on a member switch. Do not use *-n*, where *n* is a number, as the last characters in a hostname for any switch. When prompted for the Telnet (virtual terminal) password, it is 1 to 25 alphanumeric characters, is case sensitive, allows spaces, but ignores leading spaces.

Step 12   When prompted for the **enable secret** and **enable** passwords, enter the passwords of the *failed command switch* again.

Step 13   When prompted, make sure to enable the switch as the cluster command switch, and press **Return.**

Step 14   When prompted, assign a name to the cluster, and press **Return**.

The cluster name can be 1 to 31 alphanumeric characters, dashes, or underscores.

Step 15   After the initial configuration displays, verify that the addresses are correct.

Step 16   If the displayed information is correct, enter **Y**, and press **Return**.

If this information is not correct, enter **N**, press **Return**, and begin again at Step 9.

Step 17   Start your browser, and enter the IP address of the new command switch.

Step 18   From the Cluster menu, select **Add to Cluster** to display a list of candidate switches to add to the cluster.

Replacing a Failed Command Switch with Another Switch

To replace a failed command switch with a switch that is command-capable but not part of the cluster, follow these steps:

Step 1   Insert the new switch in place of the failed command switch, and duplicate its connections to the cluster members.

Step 2   You can access the CLI by using the console port or, if an IP address has been assigned to the switch, by using Telnet. For details about using the console port, see the switch hardware installation guide.

Step 3   At the switch prompt, enter privileged EXEC mode.

Example:

```
Switch> enable
Switch#
```

Step 4   Enter the password of the *failed command switch*.

Step 5   Use the setup program to configure the switch IP information. This program prompts you for IP address information and passwords. From privileged EXEC mode, enter EXEC mode, enter **setup**, and press **Return**.

Example:

```
Switch# setup

 --- System Configuration Dialog ---
Continue with configuration dialog? [yes/no]: y
At any point you may enter a question mark '?' for help.
Use ctrl-c to abort configuration dialog at any prompt.
Default settings are in square brackets '[]'.
Basic management setup configures only enough connectivity
for management of the system, extended setup will ask you
to configure each interface on the system
Would you like to enter basic management setup? [yes/no]:
```

Step 6   Enter **Y** at the first prompt.

Example:

```
The prompts in the setup program vary depending on the member switch that you selected to be the
command switch:
Continue with configuration dialog? [yes/no]: y

or

Configuring global parameters:
```

If this prompt does not appear, enter **enable**, and press **Return**. Enter **setup**, and press **Return** to start the setup program.

Step 7   Respond to the questions in the setup program.

When prompted for the hostname, it is limited to 28 characters and 31 characters on a member switch. Do not use *-n*, where *n* is a number, as the last characters in a hostname for any switch. When prompted for the Telnet (virtual terminal) password, it is 1 to 25 alphanumeric characters, is case sensitive, allows spaces, but ignores leading spaces.

Step 8   When prompted for the **enable secret** and **enable** passwords, enter the passwords of the *failed command switch* again.

Step 9   When prompted, make sure to enable the switch as the cluster command switch, and press **Return.**

Step 10   When prompted, assign a name to the cluster, and press **Return**.

The cluster name can be 1 to 31 alphanumeric characters, dashes, or underscores.

Step 11   After the initial configuration displays, verify that the addresses are correct.

Step 12   If the displayed information is correct, enter **Y**, and press **Return**.

If this information is not correct, enter **N**, press **Return**, and begin again at Step 9.

Step 13   Start your browser, and enter the IP address of the new command switch.

Step 14   From the Cluster menu, select **Add to Cluster** to display a list of candidate switches to add to the cluster.

Preventing Switch Stack Problems

To prevent switch stack problems, you should do the following:

- Make sure that the Switch that you add to or remove from the switch stack are powered off. For all powering considerations in switch stacks, see the "Switch Installation" chapter in the hardware installation guide.
- Press the **Mode** button on a stack member until the Stack mode LED is on. The last two port LEDs on the Switch should be green. Depending on the Switch model, the last two ports are either 10/100/1000 ports or small form-factor pluggable (SFP) module. If one or both of the last two port LEDs are not green, the stack is not operating at full bandwidth.
- We recommend using only one CLI session when managing the switch stack. Be careful when using multiple CLI sessions to the active stack. Commands that you enter in one session are not displayed in the other sessions. Therefore, it is possible that you might not be able to identify the session from which you entered a command.

- Manually assigning stack member numbers according to the placement of the Switch in the stack can make it easier to remotely troubleshoot the switch stack. However, you need to remember that the Switch have manually assigned numbers if you add, remove, or rearrange Switch later. Use the **switch** *current-stack-member-number* **renumber** *new-stack-member-number* global configuration command to manually assign a stack member number.

If you replace a stack member with an identical model, the new Switch functions with the exact same configuration as the replaced Switch. This is also assuming the new Switch is using the same member number as the replaced Switch.

Removing powered-on stack members causes the switch stack to divide (partition) into two or more switch stacks, each with the same configuration. If you want the switch stacks to remain separate, change the IP address or addresses of the newly created switch stacks. To recover from a partitioned switch stack, follow these steps:

1. Power off the newly created switch stacks.

2. Reconnect them to the original switch stack through their StackWise Plus ports.

3. Power on the Switch.

For the commands that you can use to monitor the switch stack and its members, see the *Displaying Switch Stack Information* section.

Preventing Autonegotiation Mismatches

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.

Troubleshooting 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. For more information about the **errdisable recovery** command, see the command reference for this release.

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.

Monitoring 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. For more information, see the **show interfaces transceiver** command in the command reference for this release.

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 Switch.

✎

Note   Though other protocol keywords are available with the **ping** command, they are not supported in this release.

Use this command to ping another device on the network from the Switch:

Command	Purpose
ping ip *host*	*address* Switch# ping 172.20.52.3	Pings a remote host through IP or by supplying the hostname or network address.

Monitoring Temperature

The Switch monitors the temperature conditions and uses the temperature information to control the fans.

Use the **show env temperature status** privileged EXEC command to display the temperature value, state, and thresholds. The temperature value is the temperature in the Switch (not the external temperature).You can configure only the yellow threshold level (in Celsius) by using the **system env temperature threshold yellow** *value* global configuration command to set the difference between the yellow and red thresholds. You cannot configure the green or red thresholds. For more information, see the command reference for this release.

Monitoring the Physical Path

You can monitor the physical path that a packet takes from a source device to a destination device by using one of these privileged EXEC commands:

Table 132: Monitoring the Physical Path

Command	Purpose
traceroute mac [**interface** *interface-id*] {*source-mac-address*} [**interface** *interface-id*] {*destination-mac-address*} [**vlan** *vlan-id*] [**detail**]	Displays the Layer 2 path taken by the packets from the specified source MAC address to the specified destination MAC address.
traceroute mac ip {*source-ip-address* | *source-hostname*} {*destination-ip-address* | *destination-hostname*} [**detail**]	Displays the Layer 2 path taken by the packets from the specified source IP address or hostname to the specified destination IP address or hostname.

Executing IP Traceroute

✎

Note   Though other protocol keywords are available with the **traceroute** privileged EXEC command, they are not supported in this release.

Command	Purpose
traceroute ip *host* `Switch# traceroute ip 192.51.100.1`	Traces the path that packets take through the network.

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.

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 or the Ethernet management 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. When you log messages to the console, very high overhead occurs. When you log messages to a virtual terminal, less overhead occurs. 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*.

Using the 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.

Configuring OBFL

> **Caution** We recommend that you do not disable OBFL and that you do not remove the data stored in the flash memory.

- To enable OBFL, use the **hw-switch switch** [*switch-number*] **logging onboard** [**message level** *level*] global configuration command. On switches, the range for *switch-number* is from 1 to 9. Use the **message level** *level* parameter to specify the severity of the hardware-related messages that the switch generates and stores in the flash memory.

- To copy the OBFL data to the local network or a specific file system, use the **copy onboard switch** *switch-number* **url** *url-destination* privileged EXEC command.

- To disable OBFL, use the **no hw-switch switch** [*switch-number*] **logging onboard** [**message level**] global configuration command.

- To clear all the OBFL data in the flash memory except for the uptime and CLI command information, use the **clear onboard switch** *switch-number* privileged EXEC command.

- In a switch stack, you can enable OBFL on a standalone switch or on all stack members by using the **hw-switch switch** [*switch-number*] **logging onboard** [**message level** *level*] global configuration command.

- You can enable or disable OBFL on a member switch from the active stack.

For more information about the commands in this section, see the command reference for this release.

Verifying Troubleshooting of the Software Configuration

Displaying OBFL Information

Table 133: Commands for Displaying OBFL Information

Command	Purpose
show logging onboard [**module**[*switch-number*]]**clilog** `Switch# show logging onboard  1 clilog`	Displays the OBFL CLI commands that were entered on a standalone switch or the specified stack members.
show logging onboard [**module**[*switch-number*]] **environment** `Switch# show logging onboard 1 environment`	Displays the UDI information for a standalone switch or the specified stack members and for all the connected FRU devices: the PID, the VID, and the serial number.
show logging onboard [**module**[*switch-number*]] **message** `Switch# show logging onboard 1 message`	Displays the hardware-related messages generated by a standalone switch or the specified stack members.
show logging onboard [**module**[*switch-number*]] **poe** `Switch# show logging onboard 1 poe`	Displays the power consumption of PoE ports on a standalone switch or the specified stack members.

Command	Purpose
show logging onboard [**module**[*switch-number*]] **temperature** Switch# show logging onboard 1 temperature	Displays the temperature of a standalone switch or or the specified stack members.
show logging onboard [**module**[*switch-number*]] **uptime** Switch# show logging onboard 1 uptime	Displays the time when a standalone switch or the specified stack members start, the reason the standalone switch or specified stack members restart, and the length of time that the standalone switch or the specified stack members have been running since they last restarted.
show logging onboard [**module**[*switch-number*]] **voltage** Switch# show logging onboard 1 voltage	Displays the system voltages of a standalone switch or the specified stack members.
show logging onboard [**module**[*switch-number*]] **continuous** Switch# show logging onboard 1 continuous	Displays the data in the continuous file.
show logging onboard [**module**[*switch-number*]] **detail** Switch# show logging onboard 1 detail	Displays both the continuous and summary data .
show logging onboard [**module**[*switch-number*]] **end***hh:mm:ss* Switch# show logging onboard 1 end 13:00:15 jul 2013	Displays end time and date on a standalone switch or the specified stack members.
show logging onboard [**module**[*switch-number*]] Switch# show logging onboard 1	Displays OBFL information about the specified switches in the system.
show logging onboard [**module**[*switch-number*]] **raw** Switch# show logging onboard 1 raw	Displays the raw information on a standalone switch or the specified stack members.
show logging onboard [**module**[*switch-number*]] **start** Switch# show logging onboard 1 start 13:00:10 jul 2013	Displays the start time and date on a standalone switch or the specified stack members.
show logging onboard [**module**[*switch-number*]] **status** Switch# show logging onboard 1 status	Displays status information on a standalone switch or the specified stack members.

Command	Purpose
show logging onboard [**module**[*switch-number*]] **summary** `Switch# show logging onboard 1 summary`	Displays both the data in the summary file

For more information, see the *Catalyst 2960-X Switch System Management Command Reference*.

Example: Verifying the 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
309 42289103 752750 56180 1.75% 1.20% 1.22% 0 RIP Timers
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 134: 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."

Scenarios for Troubleshooting the Software Configuration

Scenarios to Troubleshoot Power over Ethernet (PoE)

Table 135: Power over Ethernet Troubleshooting Scenarios

Symptom or Problem	Possible Cause and Solution
Only one port does not have PoE. Trouble is on only one switch port. PoE and non-PoE devices do not work on this port, but do on other ports.	Verify that the powered device works on another PoE port. Use the **show run**, or **show interface status** user EXEC commands to verify that the port is not shut down or error-disabled. **Note** Most switches turn off port power when the port is shut down, even though the IEEE specifications make this optional. Verify that the Ethernet cable from the powered device to the switch port is good: Connect a known good non-PoE Ethernet device to the Ethernet cable, and make sure that the powered device establishes a link and exchanges traffic with another host. Verify that the total cable length from the switch front panel to the powered device is not more than 100 meters. Disconnect the Ethernet cable from the switch port. Use a short Ethernet cable to connect a known good Ethernet device directly to this port on the switch front panel (not on a patch panel). Verify that it can establish an Ethernet link and exchange traffic with another host, or ping the port VLAN SVI. Next, connect a powered device to this port, and verify that it powers on. If a powered device does not power on when connected with a patch cord to the switch port, compare the total number of connected powered devices to the switch power budget (available PoE). Use the **show inline power** command to verify the amount of available power.

Symptom or Problem	**Possible Cause and Solution**
No PoE on all ports or a group of ports. Trouble is on all switch ports. Nonpowered Ethernet devices cannot establish an Ethernet link on any port, and PoE devices do not power on.	If there is a continuous, intermittent, or reoccurring alarm related to power, replace the power supply if possible it is a field-replaceable unit. Otherwise, replace the switch.
	If the problem is on a consecutive group of ports but not all ports, the power supply is probably not defective, and the problem could be related to PoE regulators in the switch.
	Use the **show log** privileged EXEC command to review alarms or system messages that previously reported PoE conditions or status changes.
	If there are no alarms, use the **show interface status** command to verify that the ports are not shut down or error-disabled. If ports are error-disabled, use the **shut** and **no shut** interface configuration commands to reenable the ports.
	Use the **show env power** and **show power inline** privileged EXEC commands to review the PoE status and power budget (available PoE).
	Review the running configuration to verify that **power inline never** is not configured on the ports.
	Connect a nonpowered Ethernet device directly to a switch port. Use only a short patch cord. Do not use the existing distribution cables. Enter the **shut** and **no shut** interface configuration commands, and verify that an Ethernet link is established. If this connection is good, use a short patch cord to connect a powered device to this port and verify that it powers on. If the device powers on, verify that all intermediate patch panels are correctly connected.
	Disconnect all but one of the Ethernet cables from switch ports. Using a short patch cord, connect a powered device to only one PoE port. Verify the powered device does not require more power than can be delivered by the switch port.
	Use the **show power inline** privileged EXEC command to verify that the powered device can receive power when the port is not shut down. Alternatively, watch the powered device to verify that it powers on.
	If a powered device can power on when only one powered device is connected to the switch, enter the **shut** and **no shut** interface configuration commands on the remaining ports, and then reconnect the Ethernet cables one at a time to the switch PoE ports. Use the **show interface status** and **show power inline** privileged EXEC commands to monitor inline power statistics and port status.
	If there is still no PoE at any port, a fuse might be open in the PoE section of the power supply. This normally produces an alarm. Check the log again for alarms reported earlier by system messages.

Symptom or Problem	Possible Cause and Solution
Cisco IP Phone disconnects or resets. After working normally, a Cisco phone or wireless access point intermittently reloads or disconnects from PoE.	Verify all electrical connections from the switch to the powered device. Any unreliable connection results in power interruptions and irregular powered device functioning such as erratic powered device disconnects and reloads.
	Verify that the cable length is not more than 100 meters from the switch port to the powered device.
	Notice what changes in the electrical environment at the switch location or what happens at the powered device when the disconnect occurs.
	Notice whether any error messages appear at the same time a disconnect occurs. Use the **show log** privileged EXEC command to review error messages.
	Verify that an IP phone is not losing access to the Call Manager immediately before the reload occurs. (It might be a network problem and not a PoE problem.)
	Replace the powered device with a non-PoE device, and verify that the device works correctly. If a non-PoE device has link problems or a high error rate, the problem might be an unreliable cable connection between the switch port and the powered device.
Non-Cisco powered device does not work on Cisco PoE switch. A non-Cisco powered device is connected to a Cisco PoE switch, but never powers on or powers on and then quickly powers off. Non-PoE devices work normally.	Use the **show power inline** command to verify that the switch power budget (available PoE) is not depleted before or after the powered device is connected. Verify that sufficient power is available for the powered device type before you connect it.
	Use the **show interface status** command to verify that the switch detects the connected powered device.
	Use the **show log** command to review system messages that reported an overcurrent condition on the port. Identify the symptom precisely: Does the powered device initially power on, but then disconnect? If so, the problem might be an initial surge-in (or *inrush*) current that exceeds a current-limit threshold for the port.

Configuration Examples for Troubleshooting Software

Example: Pinging an IP Host

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 136: Ping Output Display 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.

Example: Performing a Traceroute to an IP Host

This example shows how to perform a **traceroute** to an IP host:

```
Switch# traceroute ip 192.0.2.10

Type escape sequence to abort.
Tracing the route to 192.0.2.10

  1 192.0.2.1 0 msec 0 msec 4 msec
  2 192.0.2.203 12 msec 8 msec 0 msec
  3 192.0.2.100 4 msec 0 msec 0 msec
  4 192.0.2.10 0 msec 4 msec 0 msec
```

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 137: Traceroute Output Display 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.

Character	Description
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.

Example: Enabling All System Diagnostics

⚠

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.

This command disables all-system diagnostics:

```
Switch# debug all
```

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.

Additional References for Troubleshooting Software Configuration

Related Documents

Related Topic	Document Title
Troubleshooting commands	*Catalyst 2960-XR Switch System Management Command Reference*
Interface and hardware component configuration	*Catalyst 2960-XR Switch Interface and Hardware Component Configuration Guide*
Platform-independent command references	*Cisco IOS 15.3M&T Command References*
Platform-independent configuration information	*Cisco IOS 15.3M&T Configuration Guides*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/support

Feature History and Information for Troubleshooting Software Configuration

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Working with the Cisco IOS File System, Configuration Files, and Software Images

Working with the Cisco IOS File System, Configuration Files, and Software Images

Working with the Flash File System

Information About 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 bundles and configuration files. The default flash file system on the switch is named flash:.

As viewed from the active switch, or any stack member, flash: refers to the local flash device, which is the device attached to the same switch on which the file system is being viewed. In a switch stack, each of the flash devices from the various stack members can be viewed from the active switch. The names of these flash file systems include the corresponding switch member numbers. For example, flash-3:, as viewed from the active switch, refers to the same file system as does flash: on stack member 3. Use the **show file systems** privileged EXEC command to list all file systems, including the flash file systems in the switch stack.

Only one user at a time can manage the software bundles and configuration files for a switch stack .

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 for a standalone switch:

```
Switch# show file systems
```

```
File Systems:
     Size(b)       Free(b)        Type      Flags     Prefixes
*   15998976       5135872        flash      rw       flash:
          -             -        opaque      rw       bs:
          -             -        opaque      rw       vb:
      524288        520138         nvram      rw       nvram:
          -             -        network     rw       tftp:
          -             -        opaque      rw       null:
          -             -        opaque      rw       system:
          -             -        opaque      ro       xmodem:
          -             -        opaque      ro       ymodem:
```

This example shows a switch stack. In this example, the active switch is stack member 1; the file system on stack member 2 is displayed as flash-2:, the file system on stack member 3 is displayed as flash-3: and so on up to stack member 9, displayed as flash-9: for a 9-member stack. The example also shows the crashinfo directories and a USB flash drive plugged into the active switch:

```
Switch# show file systems
File Systems:
     Size(b)       Free(b)        Type      Flags     Prefixes
    145898496      5479424         disk       rw       crashinfo:crashinfo-1:
    248512512     85983232         disk       rw       crashinfo-2:stby-crashinfo:
    146014208     17301504         disk       rw       crashinfo-3:
    146014208            0         disk       rw       crashinfo-4:
    146014208      1572864         disk       rw       crashinfo-5:
    248512512     30932992         disk       rw       crashinfo-6:
    146014208      6291456         disk       rw       crashinfo-7:
    146276352     15728640         disk       rw       crashinfo-8:
    146276352     73400320         disk       rw       crashinfo-9:
*   741621760    481730560         disk       rw       flash:flash-1:
   1622147072   1360527360         disk       rw       flash-2:stby-flash:
    729546752    469762048         disk       rw       flash-3:
    729546752    469762048         disk       rw       flash-4:
    729546752    469762048         disk       rw       flash-5:
   1622147072   1340604416         disk       rw       flash-6:
    729546752    469762048         disk       rw       flash-7:
   1749549056   1487929344         disk       rw       flash-8:
   1749549056   1487929344         disk       rw       flash-9:
            0            0         disk       rw       unix:
            -            -         disk       rw       usbflash0:usbflash0-1:
            -            -         disk       rw       usbflash0-2: stby-usbflash0:
            -            -         disk       rw       usbflash0-3:
            -            -         disk       rw       usbflash0-4:
            -            -         disk       rw       usbflash0-5:
            -            -         disk       rw       usbflash0-6:
            -            -         disk       rw       usbflash0-7:
            -            -         disk       rw       usbflash0-8:
            -            -         disk       rw       usbflash0-9:
            0            0         disk       ro       webui:
            -            -        opaque      rw       system:
            -            -        opaque      rw       tmpsys:
      2097152      2055643         nvram      rw       stby-nvram:
            -            -         nvram      rw       stby-rcsf:
            -            -        opaque      rw       null:
            -            -        opaque      ro       tar:
            -            -        network     rw       tftp:
      2097152      2055643         nvram      rw       nvram:
            -            -        opaque      wo       syslog:
            -            -        network     rw       rcp:
            -            -        network     rw       http:
            -            -        network     rw       ftp:
            -            -        network     rw       scp:
```

```
          -               -     network    rw    https:
          -               -      opaque    ro    cns:
          -               -      opaque    rw    revrcsf:
```

Table 138: 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.
	disk—The file system is for a flash memory device, USB flash, and crashinfo file.
	network—The file system for network devices; for example, an FTP server or and HTTP server.
	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.

Field	Value
Prefixes	Alias for file system.
	crashinfo:—Crashinfo file.
	flash:—Flash file system.
	ftp:—FTP server.
	http:—HTTP server.
	https:—Secure HTTP server.
	nvram:—NVRAM.
	null:—Null destination for copies. You can copy a remote file to null to find its size.
	rcp:—Remote Copy Protocol (RCP) server.
	scp:—Session Control Protocol (SCP) server.
	system:—Contains the system memory, including the running configuration.
	tftp:—TFTP network server.
	usbflash0:—USB flash memory.
	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.

Setting the Default File System

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 listed in the following table.

Table 139: Commands for Displaying Information About Files

Command	Description
dir [/all] [*filesystem:filename*]	Displays a list of files on a file system.
show file systems	Displays more information about each of the files on a file system.
show file information *file-url*	Displays information about a specific file.
show file descriptors	Displays 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

Follow these steps to change directories and to display the working directory:

SUMMARY STEPS

1. **enable**
2. **dir** *filesystem:*
3. **cd** *directory_name*
4. **pwd**
5. **cd**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**enable** **Example:** Switch> **enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**dir** *filesystem:* **Example:** Switch# dir flash:	Displays the directories on the specified file system. For *filesystem:*, use flash: for the system board flash device. To access flash partitions of switch members in a stack, use flash-*n* where *n* is the stack member number. For example, flash-4.
Step 3	**cd** *directory_name* **Example:** Switch# cd new_configs	Navigates to the specified directory. The command example shows how to navigate to the directory named *new_configs*.

	Command or Action	**Purpose**
Step 4	**pwd** **Example:** Switch# pwd	Displays the working directory.
Step 5	**cd** **Example:** Switch# cd	Navigates to the default directory.

Creating Directories

Beginning in privileged EXEC mode, follow these steps to create a directory:

SUMMARY STEPS

1. **dir** *filesystem:*
2. **mkdir** *directory_name*
3. **dir** *filesystem:*

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**dir** *filesystem:* **Example:** Switch# dir flash:	Displays the directories on the specified file system. For *filesystem:*, use flash: for the system board flash device.
Step 2	**mkdir** *directory_name* **Example:** Switch# mkdir new_configs	Creates a new directory. Directory names are case sensitive and are limited to 45 characters between the slashes (/); the name cannot contain control characters, spaces, slashes, quotes, semicolons, or colons.
Step 3	**dir** *filesystem:* **Example:** Switch# dir flash:	Verifies your entry.

Removing Directories

To remove 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.

For *filesystem*, use **flash:** for the system board flash device. For *file-url*, enter the name of the directory to be deleted. All of the files in the directory and the directory are removed.

⚠

Caution   When 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)

Copying Files from One Switch in a Stack to Another Switch in the Same Stack

To copy a file from one switch in a stack to another switch in the same stack, use the **flash-X:** notation, where **X** is the switch number.

To view all switches in a stack, use the **show switch** command in privileged EXEC mode, as in the following example of a 9-member switch stack:

```
Switch# show switch
Switch/Stack Mac Address : 0006.f6b9.b580 - Local Mac Address Mac persistency wait time:
Indefinite
                                       H/W    Current
Switch#    Role     Mac Address      Priority Version  State
-----------------------------------------------------------
*1      Active   0006.f6b9.b580     15      P3B      Ready
 2      Standby  0006.f6ba.0c80     14      P3B      Ready
 3      Member   0006.f6ba.3300     7       P3B      Ready
```

```
4        Member    0006.f6b9.df80    6      P3B      Ready
5        Member    0006.f6ba.3880    13     P1A      Ready
6        Member    1ce6.c7b6.ef00    4      PP       Ready
7        Member    2037.06ce.2580    3      P2A      Ready
8        Member    2037.0653.7e00    2      P5A      Ready
9        Member    2037.0653.9280    1      P5B      Ready
```

To view all file systems available to copy on a specific switch, use the **copy** command as in the following example of a 5-member stack:

```
Switch# copy flash: ?
  crashinfo-1:     Copy to crashinfo-1: file system
  crashinfo-2:     Copy to crashinfo-2: file system
  crashinfo-3:     Copy to crashinfo-3: file system
  crashinfo-4:     Copy to crashinfo-4: file system
  crashinfo-5:     Copy to crashinfo-5: file system
  crashinfo:       Copy to crashinfo: file system
  flash-1:         Copy to flash-1: file system
  flash-2:         Copy to flash-2: file system
  flash-3:         Copy to flash-3: file system
  flash-4:         Copy to flash-4: file system
  flash-5:         Copy to flash-5: file system
  flash:           Copy to flash: file system
  ftp:             Copy to ftp: file system
  http:            Copy to http: file system
  https:           Copy to https: file system
  null:            Copy to null: file system
  nvram:           Copy to nvram: file system
  rcp:             Copy to rcp: file system
  revrcsf:         Copy to revrcsf: file system
  running-config   Update (merge with) current system configuration
  scp:             Copy to scp: file system
  startup-config   Copy to startup configuration
  stby-crashinfo:  Copy to stby-crashinfo: file system
  stby-flash:      Copy to stby-flash: file system
  stby-nvram:      Copy to stby-nvram: file system
  stby-rcsf:       Copy to stby-rcsf: file system
  stby-usbflash0:  Copy to stby-usbflash0: file system
  syslog:          Copy to syslog: file system
  system:          Copy to system: file system
  tftp:            Copy to tftp: file system
  tmpsys:          Copy to tmpsys: file system
  usbflash0-1:     Copy to usbflash0-1: file system
  usbflash0-2:     Copy to usbflash0-2: file system
  usbflash0-3:     Copy to usbflash0-3: file system
  usbflash0-4:     Copy to usbflash0-4: file system
  usbflash0-5:     Copy to usbflash0-5: file system
  usbflash0:       Copy to usbflash0: file system

Switch#
```

This example shows how to copy a config file stored in the flash partition of switch 2 to the flash partition of switch 4. It assumes that switch 2 and switch 4 are in the same stack.

```
Switch# copy flash-2:config.txt flash-4:config.txt
```

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 Files

You can create a file and write files into it, list the files in a file, and extract the files from a file as described in the next sections.

Beginning in privileged EXEC mode, follow these steps to create a file, display the contents, and extract it:

SUMMARY STEPS

1. **archive tar /create** *destination-url* **flash:** /*file-url*
2. **archive tar /table** *source-url*
3. **archive tar /xtract** *source-url* **flash:**/*file-url* [*dir/file...*]
4. **more** [/**ascii** | /**binary** | /**ebcdic**] /*file-url*

DETAILED STEPS

	Command or Action	**Purpose**

Step 1	**archive tar /create** *destination-url* **flash:** /*file-url*
	
	Example:
	
	switch# archive tar /create
	tftp:172.20.10.30/saved.
	flash:/new-configs
		Creates a file and adds files to it.
		
		For destination-url, specify the destination URL alias for the local or network file system and the name of the file to create:
		
		• Local flash file system syntax:
		
		  flash:
		• FTP syntax:
		
		  ftp:[[//*username*[:*password*]@*location*]/*directory*]/-*filename*.

	Command or Action	**Purpose**
		• RCP syntax:
		rcp:[[//*username@location*]/*directory*]/-*filename*.
		• TFTP syntax:
		tftp:[[//*location*]/*directory*]/-*filename*.
		For **flash:**/*file-url*, specify the location on the local flash file system in which the new file is created. You can also specify an optional list of files or directories within the source directory to add to the new file. If none are specified, all files and directories at this level are written to the newly created file.
Step 2	**archive tar /table** *source-url*	Displays the contents of a file.
	Example:	For *source-url*, specify the source URL alias for the local or network file system. The -*filename*. is the file to display. These options are supported:
| | ```
switch# archive tar /table
flash: /new_configs
``` | • Local flash file system syntax: |
		flash:
		• FTP syntax:
		ftp:[[//*username*[:*password*]@*location*]/*directory*]/-*filename*.
		• RCP syntax:
		rcp:[[//*username@location*]/*directory*]/-*filename*.
		• TFTP syntax:
		tftp:[[//*location*]/*directory*]/-*filename*.
		You can also limit the file displays by specifying a list of files or directories after the file. Only those files appear. If none are specified, all files and directories appear.
Step 3	**archive tar /xtract** *source-url* **flash:**/*file-url* [*dir/file...*]	Extracts a file into a directory on the flash file system.
	Example:	For *source-url*, specify the source URL alias for the local file system. The -*filename*. is the file from which to extract files. These options are supported:
| | ```
switch# archive tar /xtract
tftp:/172.20.10.30/saved.
flash:/new-configs
``` | • Local flash file system syntax: |
		flash:
		• FTP syntax:
		ftp:[[//*username*[:*password*]@*location*]/*directory*]/-*filename*.
		• RCP syntax:
		rcp:[[//*username@location*]/*directory*]/-*filename*.
		• TFTP syntax:
		tftp:[[//*location*]/*directory*]/-*filename*.
		For **flash:**/*file-url* [*dir/file...*], specify the location on the local flash file system from which the file is extracted. Use

	Command or Action	Purpose
		the *dir/file...* option to specify a list of files or directories within the file to be extracted. If none are specified, all files and directories are extracted.
Step 4	**more** [/**ascii**	/**binary**	/**ebcdic**] /*file-url* **Example:** ```switch# more flash:/new-configs```	Displays the contents of any readable file, including a file on a remote file system.

Working with Configuration Files

Information on 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.

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 or Ethernet management 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 or Ethernet management 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 byusing 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:

SUMMARY STEPS

1. Copy an existing configuration from a switch to a server.
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.

DETAILED STEPS

Step 1 Copy an existing configuration from a switch to a server.

Step 2   Open the configuration file in a text editor, such as vi or emacs on UNIX or Notepad on a PC.

Step 3   Extract the portion of the configuration file with the desired commands, and save it in a new file.

Step 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).

Step 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, ordownload 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
```

Note   You must restart the inetd daemon after modifying the /etc/inetd.conf and /etc/services files. To restart the daemon, either stop the inetd process and restart it, or enter a **fastboot** command (on the SunOS 4.x) or a **reboot** command (on Solaris 2.x or SunOS 5.x). 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 permissionon 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:

SUMMARY STEPS

1. Copy the configuration file to the appropriate TFTP directory on the workstation.
2. Verify that the TFTP server is properly configured.
3. Log into the switch through the console port, the Ethernet management port, or a Telnet session.
4. Download the configuration file from the TFTP server to configure the switch.

DETAILED STEPS

Step 1	Copy the configuration file to the appropriate TFTP directory on the workstation.
Step 2	Verify that the TFTP server is properly configured.
Step 3	Log into the switch through the console port, the Ethernet management port, or a Telnet session.
Step 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

copy tftp:[[[//location]/directory]/filename] flash[n]:/directory/startup-config
```

The configuration file downloads, and the commands are executed as the file is parsed line-by-line.

Example

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:

SUMMARY STEPS

1. Verify that the TFTP server is properly configured.
2. Log into the switch through the console port, the Ethernet management 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.

DETAILED STEPS

Step 1   Verify that the TFTP server is properly configured.

Step 2   Log into the switch through the console port, the Ethernet management port, or a Telnet session

Step 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*]

- **copy flash**[n]:/*directory*/**startup-config tftp:**[[[//*location*]/*directory*]/*filename*]

  The file is uploaded to the TFTP server.

Example

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 a Configuration File from the Switch to an FTP Server

You can copy a configuration file from the switch to an FTP server.

Understanding the FTP Username and Password

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 using FTP, the Cisco IOS software sends the first valid username it encounters in the following sequence:

1. The username specified in the **copy** EXEC command, if a username is specified.

2. The username set by the **ip ftp username** global configuration command, if the command is configured.

3. Anonymous.

The switch sends the first valid password it encounters in the following sequence:

1. The password specified in the **copy** command, if a password is specified.

2. The password set by the **ip ftp password** command, if the command is configured.

3. The switch forms a password *username* @*switchname.domain* . The variable *username* is the username associated with the current session, *switchname* is the configured host name, 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 the user on the switch.

If the server has a directory structure, the configuration file or image is written to or copied from the directory associated with the username on the server. For example, if the system image resides in the home directory of a user on the server, specify that user name as the remote username.

Refer to the documentation for your FTP server for more information.

Use the **ip ftp username** and **ip ftp password** global configuration commands to specify a username and password for all copies. Include the username in the **copy** EXEC command if you want to specify a username for that copy operation only.

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:

SUMMARY STEPS

1. **configure terminal**
2. **ip ftp username** *username*
3. **ip ftp password** *password*
4. **end**
5. Do one of the following:

    - **copy system:running-config ftp:** [[[//[*username* [:*password*]@]*location*]/*directory*]/*filename*]
    - **copy nvram:startup-config ftp:** [[[//[*username* [:*password*]@]*location*]/*directory*]/*filename*]

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal**	Enter global configuration mode on the switch.
		This step is required only if you override the default remote username or password (see Steps 2, 3, and 4).
Step 2	**ip ftp username** *username*	(Optional) Change the default remote username.
Step 3	**ip ftp password** *password*	(Optional) Change the default password.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	Do one of the following: • **copy system:running-config ftp:** [[[//[*username* [:*password*]@]*location*]/*directory*]/*filename*] • **copy nvram:startup-config ftp:** [[[//[*username* [:*password*]@]*location*]/*directory*]/*filename*]	Using FTP, copy the configuration file from a network server to the running configuration or to the startup configuration file.

Example

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:

SUMMARY STEPS

1. **configure terminal**
2. **ip ftp username** *username*
3. **ip ftp password** *password*
4. **end**
5. Do one of the following:

   - **copy system:running-config ftp:** [[[//[*username* [**:***password*]@]*location*]/*directory*]/*filename*]
     or
   - **copy nvram:startup-config ftp:** [[[//[*username* [**:***password*]@]*location*]/*directory*]/*filename*]

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal**	Enter global configuration mode on the switch.
		This step is required only if you override the default remote username or password (see Steps 2, 3, and 4).
Step 2	**ip ftp username** *username*	(Optional) Change the default remote username.
Step 3	**ip ftp password** *password*	(Optional) Change the default password.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	Do one of the following: • **copy system:running-config ftp:** [[[//[*username* [**:***password*]@]*location*]/*directory*]/*filename*] or • **copy nvram:startup-config ftp:** [[[//[*username* [**:***password*]@]*location*]/*directory*]/*filename*]	Using FTP, store the switch running or startup configuration file to the specified location.

Example

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 RCPserver 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:

SUMMARY STEPS

1. **configure terminal**
2. **ip rcmd remote-username** *username*
3. **end**
4. Do one of the following:

   - **copy rcp:**[[[//*username@*]*location*]/*directory*]/*filename*]**system:running-config**
   - **copy rcp:**[[[//*username@*]*location*]/*directory*]/*filename*]**nvram:startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal**	Enter global configuration mode on the switch.
		This step is required only if you override the default remote username (see Steps 2 and 3).
Step 2	**ip rcmd remote-username** *username*	(Optional) Change the default remote username.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	Do one of the following: • **copy rcp:**[[[//*username@*]*location*]/*directory*]/*filename*]**system:running-config** • **copy rcp:**[[[//*username@*]*location*]/*directory*]/*filename*]**nvram:startup-config**	Using RCP, copy the configuration file from a network server to the running configuration or to the startup configuration file.

Example

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 172.16.101.101to 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

SUMMARY STEPS

1. **configure terminal**
2. **ip rcmd remote-username** *username*
3. **end**
4. Do one of the following:

   - **copy system:running-config rcp:**[[[//*username@*]*location*]/*directory*]/*filename]*
   - **copy nvram:startup-config rcp:**[[[//*username@*]*location*]/*directory*]/*filename]*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal**	Enter global configuration mode on the switch.
		This step is required only if you override the default remote username (see Steps 2 and 3).
Step 2	**ip rcmd remote-username** *username*	(Optional) Specify the remote username.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	Do one of the following: • **copy system:running-config rcp:**[[[//*username@*]*location*]/*directory*]/*filename]*	Using RCP, copy the configuration file from a switch running configuration or startup configuration file to a network server.

Command or Action	**Purpose**
• **copy nvram:startup-config** **rcp:**[[[//*username@*]*location*]/*directory*]/*filename*]	

Example

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.

> ✎
>
> **Note**   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 12.4.

Note   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.

Information on Configuration Replacement and Rollback

Configuration Archive

The Cisco IOS configuration archive is intended to provide a mechanism to store, organize, and manage an archive of Cisco IOS configuration files to enhance the configuration rollback capability provided by the **configure replace** command. Before this feature was introduced, you could save copies of the running configuration using the **copy running-config** *destination-url* command, storing the replacement file either locally or remotely. However, this method lacked any automated file management. On the other hand, the Configuration Replace and Configuration Rollback feature provides the capability to automatically save copies of the running configuration to the Cisco IOS configuration archive. These archived files serve as checkpoint configuration references and can be used by the **configure replace** command to revert to previous configuration states.

The **archive config** command allows you to save Cisco IOS configurations in the configuration archive 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. This functionality provides a means for consistent identification of saved Cisco IOS configuration files. You can specify how many versions of the running configuration are kept in the archive. After the maximum number of files are saved in the archive, the oldest file is automatically deleted when the next, most recent file is saved. The **show archive** command displays information for all configuration files saved in the Cisco IOS configuration archive.

The Cisco IOS configuration archive, in which the configuration files are stored and available for use with the **configure replace** command, can be located on the following file systems: FTP, HTTP, RCP, TFTP.

Configuration Replace

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.

Configuration Rollback

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:

SUMMARY STEPS

1. **configure terminal**
2. **archive**
3. **path***url*
4. **maximum***number*
5. **time-period** *minutes*
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**archive**	Enter archive configuration mode.
Step 3	**path***url*	Specify the location and filename prefix for the files in the configuration archive
Step 4	**maximum***number*	(Optional) Set 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.
Step 5	**time-period** *minutes*	(Optional) Set the time increment for automatically saving an archive file of the running configuration in the configuration archive.
		minutes-Specify how often, in minutes, to automatically save an archive file of the running configuration in the configuration archive
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show running-config**	Verify the configuration.
Step 8	**copy running-config startup-config**	(Optional) Save 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:

SUMMARY STEPS

1. **archive config**
2. **configure terminal**
3. Make necessary changes to the running configuration.
4. **exit**
5. **configure replace** *target-url* [**list**] [**force**] [**time** *seconds*] [**nolock**]
6. **configure confirm**
7. **copy running-config startup-config**

DETAILED STEPS

Step 1   **archive config**

(Optional) Save the running configuration file to the configuration archive.

Note   Enter the **path** archive configuration command before using this command.

Step 2   **configure terminal**

Enter global configuration mode.

Step 3   Make necessary changes to the running configuration.

—

Step 4   **exit**

Return to privileged EXEC mode.

Step 5   **configure replace** *target-url* [**list**] [**force**] [**time** *seconds*] [**nolock**]

Replace 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 —Display 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 —Replace the running configuration file with the specified saved configuration file without prompting you for confirmation.

time*seconds*—Specify 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— Disable the locking of the running configuration file that prevents other users from changing the running configuration during a configuration replacement operation.

Step 6   **configure confirm**

(Optional) Confirm 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.

Step 7 **copy running-config startup-config**

(Optional) Save your entries in the configuration file.

Working with Software Images

Information on 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. For switch stacks, the **archive download-sw** and **archive upload-sw** privileged EXEC commands can only be used through the stack's active switch. Software images downloaded to the stack's active switch are automatically downloaded to the rest of the stack members.
	To upgrade a switch in the stack that has an incompatible software image, use the **archive copy-sw** privileged EXEC command to copy the software image from an existing stack member to the incompatible switch. That switch automatically reloads and joins the stack as a fully functioning member.

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 the 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.

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. The table 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
```

Table 140: 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

Field	Description
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
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 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. For switch stacks, the **archive download-sw** and **archive upload-sw** privileged EXEC commands can only be used through the stack's active switch. Software images downloaded to the stack's active switch are automatically downloaded to the rest of the stack members.

To upgrade a switch with an incompatible software image, use the **archive copy-sw** privileged EXEC command to copy the software image from an existing stack member to the incompatible switch. That switch automatically reloads and joins the stack as a fully functioning member.

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
  ```

> **Note** You must restart the inetd daemon after modifying the /etc/inetd.conf and /etc/services files. To restart the daemon, either stop the inetd process and restart it, or enter a fastboot command (on the SunOS 4.x) or a **reboot** command (on Solaris 2.x or SunOS 5.x). 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.

SUMMARY STEPS

1. Copy the image to the appropriate TFTP directory on the workstation. Make sure that the TFTP server is properly configured.
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**
4. **archive download-sw** / **leave-old-sw** / **reload tftp:** [[/ / *location*] / *directory*] / *image-name* **.tar**

DETAILED STEPS

Step 1 Copy the image to the appropriate TFTP directory on the workstation. Make sure that the TFTP server is properly configured.

—

Step 2 Log into the switch through the console port or a Telnet session.

—

Step 3   **archive download-sw** /**overwrite** /**reload tftp:** [[/ / *location*] / *directory*] / *image-name***.tar**

Download 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.

Step 4   **archive download-sw** /**leave-old-sw** /**reload tftp:** [[/ / *location*] / *directory*] / *image-name***.tar**

Download 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 cancels 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 keep 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.

Note   For the download and upload algorithms to operate properly, do not rename image names

Uploading an Image File 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:

SUMMARY STEPS

1. Make sure the TFTP server is properly configured
2. Log into the switch through the console port or a Telnet session.
3. **archive upload-sw tftp:**[[// *location*]/*directory*]/*image-name* **.tar**

DETAILED STEPS

Step 1 Make sure the TFTP server is properly configured

—

Step 2 Log into the switch through the console port or a Telnet session.

—

Step 3 **archive upload-sw tftp:**[[// *location*]/*directory*]/*image-name* **.tar**

Upload 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.

Note For the download and upload algorithms to operate properly, do not rename image names.

Copying Image Files 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. For switch stacks, the **archive download-sw** and **archive upload-sw** privileged EXEC commands can only be used through the stack's active switch. Software images downloaded to the stack's active switch are automatically downloaded to the rest of the stack members.

To upgrade a switch with an incompatible software image, use the **archive copy-sw** privileged EXEC command to copy the software image from an existing stack member to the incompatible switch. That switch automatically reloads and joins the stack as a fully functioning member.

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.

SUMMARY STEPS

1. Verify that the FTP server is properly configured.
2. Log into the switch through the console port or a Telnet session.
3. **configure terminal**
4. **ip ftp username** *username*
5. **ip ftp password***password*
6. **end**
7. **archive download-sw** /**overwrite** /**reload**
   ftp: [[/ / *username* [**:***password*] @*location*] / *directory*] / *image-name***.tar**
8. **archive download-sw** /**leave-old-sw** /**reload**
   ftp: [[/ / *username* [**:***password*] @*location*] / *directory*] / *image-name***.tar**

DETAILED STEPS

| Step 1 | Verify that the FTP server is properly configured. |

—

| Step 2 | Log into the switch through the console port or a Telnet session. |

—

| Step 3 | **configure terminal** |

Enter global configuration mode.

This step is required only if you override the default remote username or password (see Steps 4, 5, and 6).

Step 4   **ip ftp username** *username*

(Optional) Change the default remote username.

Step 5   **ip ftp password***password*

(Optional) Change the default password.

Step 6   **end**

Return to privileged EXEC mode.

Step 7   **archive download-sw** /**overwrite**/**reload ftp:** [[/ / *username* [**:***password*] @*location*] / *directory*] / *image-name***.tar**

Download 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.

Step 8   **archive download-sw** /**leave-old-sw**/**reload ftp:** [[/ / *username* [**:***password*] @*location*] / *directory*] / *image-name***.tar**

Download 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 cancels 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.

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:

SUMMARY STEPS

1. **configure terminal**
2. **ip ftp username***username*
3. **ip ftp password***password*
4. **end**
5. **archive upload-sw ftp:** [[/ / [*username* [**:** *password*] **@**] *location*] / *directory*] / *image-name***.tar**

DETAILED STEPS

Step 1   **configure terminal**

Enter global configuration mode.

This step is required only if you override the default remote username or password (see Steps 2, 3, and 4.)

Step 2   **ip ftp username***username*

Optional) Change the default remote username.

Step 3   **ip ftp password***password*

(Optional) Change the default password.

Step 4   **end**

Return to privileged EXEC mode.

Step 5   **archive upload-sw ftp:** [[/ / [*username* [**:** *password*] **@**] *location*] / *directory*] / *image-name***.tar**

Upload 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.

| **Note** | For the download and upload algorithms to operate properly, do not rename image names. |

Copying Image Files 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. For switch stacks, the **archive download-sw** and **archive upload-sw** privileged EXEC commands can only be used through the stack's active switch. Software images downloaded to the stack's active switch are automatically downloaded to the rest of the stack members. To upgrade a switch with an incompatible software image, use the **archive copy-sw** privileged EXEC command to copy the software image from an existing stack member to the incompatible switch. That switch automatically reloads and joins the stack as a fully functioning member. |

Preparing to Download or Upload an Image File 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 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.

SUMMARY STEPS

1. Verify that the RCP server is properly configured.
2. Log into the switch through the console port or a Telnet session.
3. **configure terminal**
4. **ip rcmd remote-username** *username*
5. **end**
6. **archive download-sw** /**overwrite** /**reload**
   rcp: [[[//*username*@] /*location*] /*directory*] /*image-name*.**tar**
7. **archive download-sw** /**leave-old-sw** /**reload**
   rcp: [[[// [*username*@] *location*] /*directory*] /*image-name*.**tar**

DETAILED STEPS

Step 1   Verify that the RCP server is properly configured.

—

Step 2   Log into the switch through the console port or a Telnet session.

—

Step 3   **configure terminal**

Enter global configuration mode.

This step is required only if you override the default remote username or password (see Steps 4, 5, and 6).

Step 4   **ip rcmd remote-username** *username*

(Optional) Specify the remote username.

Step 5   **end**

Return to privileged EXEC mode.

Step 6   **archive download-sw /overwrite /reload rcp:** [[[/ / *username*@] / *location*] / *directory*] / *image-name***.tar**

Download 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 theRCP server.
- For /*directory*/*image-name***.tar**, specify the directory (optional) and the image to download. Directory and image names are case sensitive.

Step 7   **archive download-sw /leave-old-sw /reload rcp:** [[[/ / [*username*@] *location*] / *directory*] / *image-name***.tar**

Download 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*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 cancels 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.

Note   For the download and upload algorithms to operate properly, do not rename image names.

Uploading an Image File 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:

SUMMARY STEPS

1. **configure terminal**
2. **ip rcmd remote-username***username*
3. **end**
4. **archive upload-sw rcp:** [[[/ / [*username* @] *location*] / *directory*] / *image-name***.tar**

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**configure terminal**	Enter global configuration mode.
		This step is required only if you override the default remote username or password (see Steps 2 and 3.)
Step 2	**ip rcmd remote-username***username*	Optional) Specify the remote username.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**archive upload-sw rcp:** [[[/ / [*username* @] *location*] / *directory*] / *image-name***.tar**	Upload 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.

Command or Action	Purpose
	• 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** 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.
	Note For the download and upload algorithms to operate properly, do not rename image names.

Copying an Image File from One Stack Member to Another

For switch stacks, the **archive download-sw** and **archive upload-sw** privileged EXEC commands can be used only through the stack's active switch. Software images downloaded to the stack's active switch are automatically downloaded to the rest of the stack members.

To upgrade a switch that has an incompatible software image, use the **archive copy-sw** privileged EXEC command to copy the software image from an existing stack member to the one that has incompatible software. That switch automatically reloads and joins the stack as a fully functioning member.

Note   To successfully use the **archive copy-sw** privileged EXEC command, you must have downloaded from a TFTP server the images for both the stack member switch being added and the stack's active switch. You use the **archive download-sw** privileged EXEC command to perform the download.

Beginning in privileged EXEC mode from the stack member that you want to upgrade, follow these steps to copy the running image file from the flash memory of a different stack member:

SUMMARY STEPS

1. **archive copy-sw** / **destination-system** *destination-stack-member-number* / **force-reload***source-stack-member-number*
2. **reload slot***stack-member-number*

DETAILED STEPS

	Command or Action	**Purpose**
Step 1	**archive copy-sw**/**destination-system** *destination-stack-member-number* / **force-reload***source-stack-member-number*	Copy the running image file from a stack member, and then unconditionally reload the updated stack member.
		Note At least one stack member must be running the image that is to be copied to the switch that is running the incompatible software
		For /**destination-system***destination-stack-member-number*, specify the number of the stack member (the destination) to which to copy the source running image file. If you do not specify this stack member number, the default is to copy the running image file to all stack members.
		Specify /**force-reload** to unconditionally force a system reload after successfully downloading the software image.
		For *source-stack-member-number*, specify the number of the stack member (the source) from which to copy the running image file. The stack member number range is 1 to 9.
Step 2	**reload slot***stack-member-number*	Reset the updated stack member, and put this configuration change into effect.

INDEX