Consolidated Platform Configuration Guide, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)

First Published: April 12, 2013

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

Text Part Number: OL-29323-01

CONTENTS

Preface

This preface contains the following topics:

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.

Convention	Description
[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.
[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 uses 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 **Means *reader be warned.* In this situation, you might perform an action that could result in bodily injury.**

Related Documentation

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

- Cisco 5700 Series Wireless Controller documentation, located at:

  http://www.cisco.com/go/wlc5700_sw

- Cisco Validated Designs documents, located at:

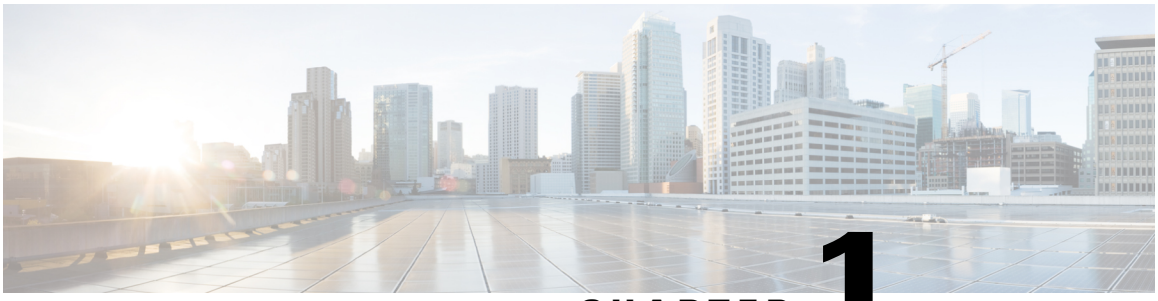
  http://www.cisco.com/go/designzone

Obtaining Documentation and Submitting a Service Request

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

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

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



CHAPTER 1

Using the Command-Line Interface

This chapter contains the following topics:

Information About Using the Command-Line Interface

This section describes the Cisco IOS command-line interface (CLI) and how to use it.

Command Modes

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

When you start a session using Telnet, SSH, or console on the controller, 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 controller 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 controller 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.	`Controller>`	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.	`Controller#`	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.	`Controller(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 controller.
VLAN configuration	While in global configuration mode, enter the **vlan** *vlan-id* command.	`Controller(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 controller startup configuration file.
Interface configuration	While in global configuration mode, enter the **interface** command (with a specific interface).	`Controller(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.

Mode	Access Method	Prompt	Exit Method	About This Mode
Line configuration	While in global configuration mode, specify a line with the **line vty** or **line console** command.	`Controller(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.

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:** `Controller# help`	Obtains a brief description of the help system in any command mode.

Step 2	*abbreviated-command-entry* **?**
	
	Example:
	`Controller# di?`
	`dir disable disconnect`
		Obtains a list of commands that begin with a particular character string.

Step 3	*abbreviated-command-entry* <Tab>
	
	Example:
	`Controller# sh conf<tab>`
	`Controller# show configuration`
		Completes a partial command name.

	Command or Action	Purpose
Step 4	**?** **Example:** `Controller> ?`	Lists all commands available for a particular command mode.
Step 5	*command* **?** **Example:** `Controller> show ?`	Lists the associated keywords for a command.

Step 6	*command keyword* **?**
	
	Example:
	`Controller(config)# cdp holdtime ?`
	`  <10-255> Length of time (in sec) that receiver must keep this packet`
		Lists the associated arguments for a keyword.

Understanding Abbreviated Commands

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

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

```
Controller# 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 controller.

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 controller to recognize the command.	Reenter the command followed by a question mark (?) with a space between the command and the question mark. The possible keywords that you can enter with the command appear.
`% Incomplete command.`	You did not enter all the keywords or values required by this command.	Reenter the command followed by a question mark (?) with a space between the command and the question mark. The possible keywords that you can enter with the command appear.
`% Invalid input detected at '^' marker.`	You entered the command incorrectly. The caret (^) marks the point of the error.	Enter a question mark (?) to display all the commands that are available in this command mode. The possible keywords that you can enter with the command appear.

Configuration Logging

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

Note Only CLI or HTTP changes are logged.

How to Use the CLI to Configure Features

Configuring the Command History

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

Changing the Command History Buffer Size

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

SUMMARY STEPS

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

DETAILED STEPS

	Command or Action	Purpose
Step 1	**terminal history** [**size** *number-of-lines*] **Example:** `Controller# terminal history size 200`	Changes the number of command lines that the controller records during the current terminal session in the privileged EXEC mode. You can configure the size from 0 through 256.
Step 2	**history** [**size** *number-of-lines*] **Example:** `Controller(config)# history size 200`	Configures the number of command lines the controller records for all sessions on a particular line in the configuration mode. You can configure the size from 0 through 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.

	Command or Action	Purpose
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:** `Controller# 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. These procedures are optional.

SUMMARY STEPS

1. **terminal no history**
2. **no history**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**terminal no history** **Example:** `Controller# terminal no history`	Disables the feature during the current terminal session in the privileged EXEC mode.
Step 2	**no history** **Example:** `Controller(config)# no history`	Disables command history for the line in the configuration mode.

Enabling and Disabling Editing Features

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

SUMMARY STEPS

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

DETAILED STEPS

	Command or Action	Purpose
Step 1	**no editing** **Example:** `Controller(config)# no editing`	Disables the enhanced editing mode.
Step 2	**terminal editing** **Example:** `Controller# terminal editing`	Reenables the enhanced editing mode for the current terminal session in the privileged EXEC mode.
Step 3	**editing** **Example:** `Controller(config)# editing`	Reconfigures a specific line to have enhanced editing mode.

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.

SUMMARY STEPS

1. **Ctrl-B** or use the **left arrow** key
2. **Ctrl-F** or use the **right arrow** key
3. **Ctrl-A**
4. **Ctrl-E**
5. **Esc B**
6. **Esc F**
7. **Ctrl-T**
8. **Ctrl-Y**
9. **Esc Y**
10. **Delete** or **Backspace** key
11. **Ctrl-D**
12. **Ctrl-K**
13. **Ctrl-U** or **Ctrl-X**
14. **Ctrl-W**
15. **Esc D**
16. **Esc C**
17. **Esc L**
18. **Esc U**
19. **Ctrl-V** or **Esc Q**
20. **Return** key
21. **Space** bar
22. **Ctrl-L** or **Ctrl-R**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**Ctrl-B** or use the **left arrow** key	Moves the cursor back one character.
Step 2	**Ctrl-F** or use the **right arrow** key	Moves the cursor forward one character.
Step 3	**Ctrl-A**	Moves the cursor to the beginning of the command line.
Step 4	**Ctrl-E**	Moves the cursor to the end of the command line.
Step 5	**Esc B**	Moves the cursor back one word.
Step 6	**Esc F**	Moves the cursor forward one word.
Step 7	**Ctrl-T**	Transposes the character to the left of the cursor with the character located at the cursor.
Step 8	**Ctrl-Y**	Recalls the most recent entry in the buffer.

	Command or Action	Purpose
		Recall commands from the buffer and paste them in the command line. The controller provides a buffer with the last ten items that you deleted.
Step 9	**Esc Y**	Recalls the next buffer entry. The buffer contains only the last 10 items that you have deleted or cut. If you press **Esc Y** more than ten times, you cycle to the first buffer entry.
Step 10	**Delete** or **Backspace** key	Erases the character to the left of the cursor.
Step 11	**Ctrl-D**	Deletes the character at the cursor.
Step 12	**Ctrl-K**	Deletes all characters from the cursor to the end of the command line.
Step 13	**Ctrl-U** or **Ctrl-X**	Deletes all characters from the cursor to the beginning of the command line.
Step 14	**Ctrl-W**	Deletes the word to the left of the cursor.
Step 15	**Esc D**	Deletes from the cursor to the end of the word.
Step 16	**Esc C**	Capitalizes at the cursor.
Step 17	**Esc L**	Changes the word at the cursor to lowercase.
Step 18	**Esc U**	Capitalizes letters from the cursor to the end of the word.
Step 19	**Ctrl-V** or **Esc Q**	Designates a particular keystroke as an executable command, perhaps as a shortcut.
Step 20	**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.
Step 21	**Space** bar	Scrolls down one screen.
Step 22	**Ctrl-L** or **Ctrl-R**	Redisplays the current command line if the controller 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 extend 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:
	`Controller(config)# access-list 101 permit tcp 10.15.22.25 255.255.255.0 10.15.22.35`
	`Controller(config)# $ 101 permit tcp 10.15.22.25 255.255.255.0 10.15.22.35 255.25`
	`Controller(config)# $t tcp 10.15.22.25 255.255.255.0 131.108.1.20 255.255.255.0 eq`
	`Controller(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:** `Controller(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:** `Controller# 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

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

Accessing the CLI through a Console Connection or through Telnet

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

If your controller is already configured, you can access the CLI through a local console connection or through a remote Telnet session, but your controller must first be configured for this type of access.

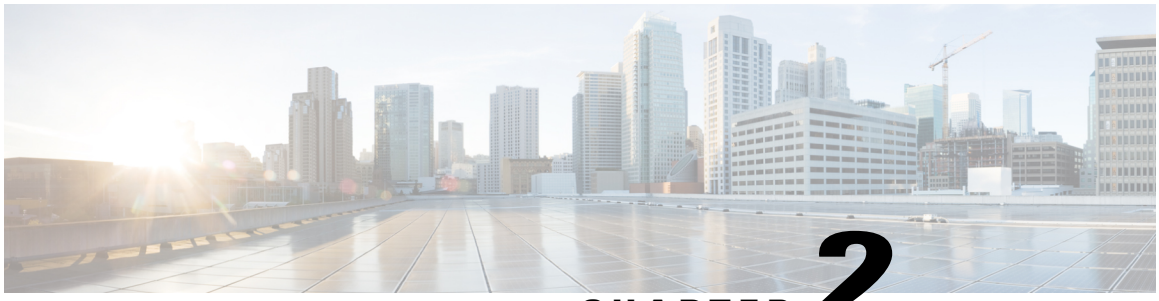
You can use one of these methods to establish a connection with the controller:

- Connect the controller 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 controller hardware installation guide.
- Use any Telnet TCP/IP or encrypted Secure Shell (SSH) package from a remote management station. The controller must have network connectivity with the Telnet or SSH client, and the controller must have an enable secret password configured.
    - The controller supports up to 16 simultaneous Telnet sessions. Changes made by one Telnet user are reflected in all other Telnet sessions.
    - The controller 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

System Management



CHAPTER 2

Administering the System

This module contains the following topics:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Administering the Controller

System Time and Date Management

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

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:

- 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.

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 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

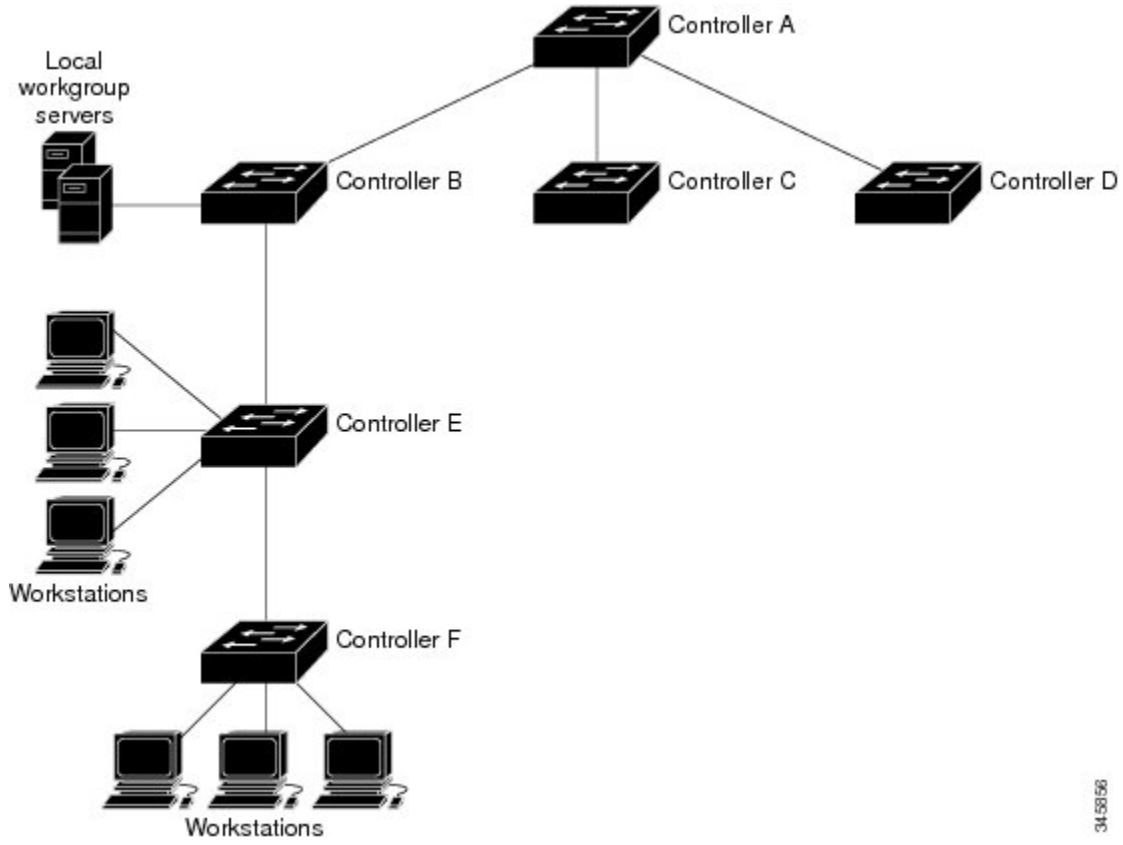
The time kept on a device is a critical resource; you should use the security features of NTP to avoid the accidental or malicious setting of an incorrect time. Two mechanisms are available: an access list-based restriction scheme and an encrypted authentication mechanism.

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.

The following figure shows a typical network example using NTP. Controller A is the NTP master, with the Controller B, C, and D configured in NTP server mode, in server association with Controller A. Controller E is configured as an NTP peer to the upstream and downstream controllers, Controller B and Controller F, respectively.

Figure 1: Typical NTP Network Configuration



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 controller. 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.

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 controller, 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 3: 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.

The MOTD and login banners are not configured.

Default Banner Configuration

The MOTD and login banners are not configured.

MAC Address Table

The MAC address table contains address information that the controller 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 controller 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 controller 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).

MAC Address Table Creation

With multiple MAC addresses supported on all ports, you can connect any port on the controller to other network devices. The controller 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 controller 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 controller maintains an address table for each VLAN, and STP can accelerate the aging interval on a per-VLAN basis.

The controller sends packets between any combination of ports, based on the destination address of the received packet. Using the MAC address table, the controller 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 controller 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.

Default MAC Address Table Settings

The following table shows the default settings for the MAC address table.

Table 4: 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.

How to Administer the Controller

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 as a last resort. If you have an outside source to which the controller can synchronize, you do not need to manually set the system clock.

Setting the System Clock

If you have an outside source on the network that provides time services, such as an NTP server, you do not need to manually set the system clock.

Beginning in privileged EXEC mode, follow these steps to set the system clock:

SUMMARY STEPS

1. 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	Use one of the following: • **clock set** *hh:mm:ss day month year* • **clock set** *hh:mm:ss month day year* **Example:** `Controller# clock set 13:32:00 23 March 2013`	Sets the system clock using one of these formats. • *hh:mm:ss*—Specifies the time in hours (24-hour format), minutes, and seconds. The time specified is relative to the configured time zone. • *day*—Specifies the day by date in the month. • *month*—Specifies the month by name. • *year*—Specifies the year (no abbreviation).

Configuring the Time Zone

SUMMARY STEPS

1. **configure terminal**
2. **clock timezone** *zone hours-offset* [*minutes-offset*]
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**clock timezone** *zone hours-offset* [*minutes-offset*] **Example:** `Controller(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 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Configuring Summer Time (Daylight Saving Time)

To configure summer time (daylight saving time) in areas where it starts and ends on a particular day of the week each year, perform this task:

SUMMARY STEPS

1. **configure terminal**
2. **clock summer-time** *zone* **date** *date month year hh:mm date month year hh:mm* [*offset*]]
3. **clock summer-time** *zone* **recurring** [*week day month hh:mm week day month hh:mm* [*offset*]]
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**clock summer-time** *zone* **date** *date month year hh:mm date month year hh:mm* [*offset*]] **Example:** `Controller(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 3	**clock summer-time** *zone* **recurring** [*week day month hh:mm week day month hh:mm* [*offset*]] **Example:** `Controller(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**). • (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 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Configuring a System Name

SUMMARY STEPS

1. **configure terminal**
2. **hostname** *name*
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**hostname** *name* **Example:** `Controller(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 Controller. 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 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Setting Up DNS

If you use the controller 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.

SUMMARY STEPS

1. **configure terminal**
2. **ip domain-name** *name*
3. **ip name-server** *server-address1* [*server-address2 ... server-address6*]
4. **ip domain-lookup** [**nsap** | **source-interface** *interface*]
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**ip domain-name** *name* **Example:** `Controller(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). 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 controller 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 3	**ip name-server** *server-address1* [*server-address2 ... server-address6*] **Example:** `Controller(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 controller sends DNS queries to the primary server first. If that query fails, the backup servers are queried.
Step 4	**ip domain-lookup** [**nsap**	**source-interface** *interface*] **Example:** `Controller(config)# ip domain-lookup`	(Optional) Enables DNS-based hostname-to-address translation on your controller. 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 5	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

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 controller

SUMMARY STEPS

1. **configure terminal**
2. **banner motd** *c message c*
3. **end**

DETAILED STEPS

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

Step 2	**banner motd** *c message c*
	
	Example:
	`Controller(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 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Configuring a Login Banner

You can configure a login banner to be displayed on all connected terminals. This banner appears after the MOTD banner and before the login prompt.

SUMMARY STEPS

1. **configure terminal**
2. **banner login** *c message c*
3. **end**

DETAILED STEPS

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

Step 2	**banner login** *c message c*
	
	Example:
	
	`Controller(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.

Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Managing the MAC Address Table

Changing the Address Aging Time

SUMMARY STEPS

1. **configure terminal**
2. **mac address-table aging-time** [*0* | *10-1000000*] [**routed-mac** | **vlan** *vlan-id*]
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**mac address-table aging-time** [*0* | *10-1000000*] [**routed-mac** | **vlan** *vlan-id*] **Example:** `Controller(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 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Configuring MAC Address Change Notification Traps

SUMMARY STEPS

1. **configure terminal**
2. **snmp-server host** *host-addr community-string notification-type* { **informs** | **traps** } {**version** {**1** | **2c** | **3**}} {**vrf** *vrf instance name*}
3. **snmp-server enable traps mac-notification change**
4. **mac address-table notification change**
5. **mac address-table notification change** [**interval** *value*] [**history-size** *value*]
6. **interface** *interface-id*
7. **snmp trap mac-notification change** {**added** | **removed**}
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**snmp-server host** *host-addr community-string notification-type* { **informs** | **traps** } {**version** {**1** | **2c** | **3**}} {**vrf** *vrf instance name*} **Example:** `Controller(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 3	**snmp-server enable traps mac-notification change** **Example:** `Controller(config)# snmp-server enable traps mac-notification change`	Enables the controller to send MAC address change notification traps to the NMS.
Step 4	**mac address-table notification change** **Example:** `Controller(config)# mac address-table notification change`	Enables the MAC address change notification feature.
Step 5	**mac address-table notification change** [**interval** *value*] [**history-size** *value*] **Example:** `Controller(config)# mac address-table notification change interval 123`	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.

	Command or Action	Purpose
	`Controller(config)#mac address-table notification change history-size 100`	• (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 6	**interface** *interface-id* **Example:** `Controller(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 7	**snmp trap mac-notification change** {**added** | **removed**} **Example:** `Controller(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 8	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.

Configuring MAC Address Move Notification Traps

When you configure MAC-move notification, an SNMP notification is generated and sent to the network management system whenever a MAC address moves from one port to another within the same VLAN.

Beginning in privileged EXEC mode, follow these steps to configure the controller to send MAC address-move notification traps to an NMS host:

SUMMARY STEPS

1. **configure terminal**
2. **snmp-server host** *host-addr* {**traps** | **informs**} {**version** {**1** | **2c** | **3**}} *community-string notification-type*
3. **snmp-server enable traps mac-notification move**
4. **mac address-table notification mac-move**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**snmp-server host** *host-addr* {**traps** | **informs**} {**version** {**1** | **2c** | **3**}} *community-string notification-type* **Example:** `Controller(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 3	**snmp-server enable traps mac-notification move** **Example:** `Controller(config)# snmp-server enable traps mac-notification move`	Enables the controller to send MAC address move notification traps to the NMS.
Step 4	**mac address-table notification mac-move** **Example:** `Controller(config)# mac address-table notification mac-move`	Enables the MAC address move notification feature.
Step 5	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Configuring MAC Threshold Notification Traps

When you configure MAC threshold notification, an SNMP notification is generated and sent to the network management system when a MAC address table threshold limit is reached or exceeded.

SUMMARY STEPS

1. **configure terminal**
2. **snmp-server host** *host-addr* {**traps** | **informs**} {**version** {**1** | **2c** | **3**}} *community-string notification-type*
3. **snmp-server enable traps mac-notification threshold**
4. **mac address-table notification threshold**
5. **mac address-table notification threshold** [**limit** *percentage*] | [**interval** *time*]
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**snmp-server host** *host-addr* {**traps** | **informs**} {**version** {**1** | **2c** | **3**}} *community-string notification-type* **Example:** `Controller(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 3	**snmp-server enable traps mac-notification threshold** **Example:** `Controller(config)# snmp-server enable traps mac-notification threshold`	Enables MAC threshold notification traps to the NMS.

	Command or Action	Purpose
Step 4	**mac address-table notification threshold** **Example:** `Controller(config)# mac address-table notification threshold`	Enables the MAC address threshold notification feature.

Step 5	**mac address-table notification threshold** [**limit** *percentage*] | [**interval** *time*]
	
	Example:
	`Controller(config)# mac address-table notification threshold interval 123`
	`Controller(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 6	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Adding and Removing Static Address Entries

SUMMARY STEPS

1. **configure terminal**
2. **mac address-table static** *mac-addr* **vlan** *vlan-id* **interface** *interface-id*
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**mac address-table static** *mac-addr* **vlan** *vlan-id* **interface** *interface-id*	Adds a static address to the MAC address table. • *mac-addr*—Specifies the destination MAC unicast address to add to the address table. Packets with this destination address received in the specified VLAN are forwarded to the specified interface.

	Command or Action	Purpose
	Example: `Controller(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 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring Unicast MAC Address Filtering

SUMMARY STEPS

1. **configure terminal**
2. **mac address-table static** *mac-addr* **vlan** *vlan-id* **drop**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**mac address-table static** *mac-addr* **vlan** *vlan-id* **drop** **Example:** `Controller(config)# mac address-table static c2f3.220a.12f4 vlan 4 drop`	Enables unicast MAC address filtering and configure the controller 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.

	Command or Action	Purpose
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Monitoring and Maintaining Administration of the Controller

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.

Command	Purpose
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 Controller Administration

Setting the System Clock: Example

This example shows how to manually set the system clock to 1:32 p.m. on July 23, 2001:

```
Controller# clock set 13:32:00 23 July 2001
```

Configuring Summer Time: Examples

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:

```
Controller(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:

```
Controller(config)#clock summer-time PST date
20 March 2013 2:00 20 November 2013 2:00
```

Configuring a MOTD Banner: Example

This example shows how to configure a MOTD banner by using the pound sign (#) symbol as the beginning and ending delimiter:

```
Controller(config)# banner motd #

This is a secure site. Only authorized users are allowed.
For access, contact technical support.

#

Controller(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:
```

Configuring a Login Banner: Example

This example shows how to configure a login banner by using the dollar sign ($) symbol as the beginning and ending delimiter:

```
Controller(config)# banner login $

Access for authorized users only. Please enter your username and password.

$

Controller(config)#
```

Configuring MAC Address Change Notification Traps: Example

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:

```
Controller(config)# snmp-server host 172.20.10.10 traps private mac-notification
Controller(config)# snmp-server enable traps mac-notification change
Controller(config)# mac address-table notification change
Controller(config)# mac address-table notification change interval 123
Controller(config)# mac address-table notification change history-size 100
Controller(config)# interface gigabitethernet1/2/1
Controller(config-if)# snmp trap mac-notification change added
```

Configuring MAC Threshold Notification Traps: Example

This example shows how to specify 172.20.10.10 as the NMS, enable the MAC address threshold notification feature, set the interval time to 123 seconds, and set the limit to 78 per cent:

```
Controller(config)# snmp-server host 172.20.10.10 traps private mac-notification
Controller(config)# snmp-server enable traps mac-notification threshold
Controller(config)# mac address-table notification threshold
Controller(config)# mac address-table notification threshold interval 123
Controller(config)# mac address-table notification threshold limit 78
```

Adding the Static Address to the MAC Address Table: Example

This example shows how to add the static address c2f3.220a.12f4 to the MAC address table. When a packet is received in VLAN 4 with this MAC address as its destination address, the packet is forwarded to the specified port:

```
Controller(config)# mac address-table static c2f3.220a.12f4 vlan 4 interface
gigabitethernet1/1/1
```

Configuring Unicast MAC Address Filtering: Example

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:

```
Controller(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	
Network management configuration	
Layer 2 configuration	
VLAN configuration	
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

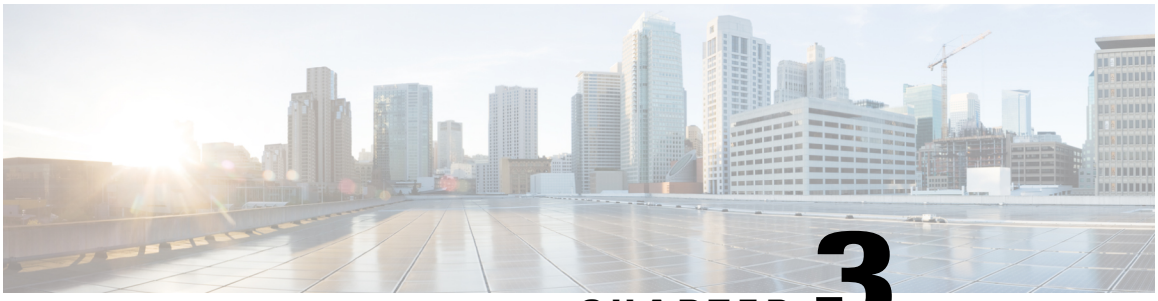
MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

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 Controller Administration

Release	Modification
	This feature was introduced.



CHAPTER 3

Performing Controller Setup Configuration

Controller setup configuration tasks include how to assign the IP address for your controller by using a variety of automatic and manual methods.

This module contains the following topics:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Performing Controller Setup Configuration

Review the sections in this module before performing your initial controller configuration tasks that include IP address assignments and DHCP autoconfiguration.

Controller Boot Process

To start your controller, you need to follow the procedures in the hardware installation guide for installing and powering on the controller and setting up the initial controller configuration (IP address, subnet mask, default gateway, secret and Telnet passwords, and so forth).

The normal boot process involves the operation of the boot loader software 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 controller.

The boot loader provides access to the 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 mechanism provides enough access to the system so that if it is necessary, you can reinstall the operating system software image by using the **emergency-install** command and restart the operating system.

Before you can assign controller information, make sure 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 these of the controller console port:

- 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.

Software Installer Features

The following software installer features are supported on your controller:

- Software bundle installation on a standalone controller.
- Software rollback to a previously installed package set.
- Emergency installation in the event that no valid installed packages reside on the boot flash.

Note Software installation and rollback must be performed while running only in installed mode. You can use the **software expand** EXEC command to convert bundle boot mode to install mode.

Software Boot Modes

Your controller supports two modes to boot the software packages:

- Installed mode
- Bundle mode

Related Topics

Displaying Software Install: Examples, on page 59

Emergency Installation: Example, on page 60

Installed Boot Mode

You can boot your controller in installed mode by booting the software package provisioning file that resides in flash:

```
controller: boot flash:packages.conf
```

The provisioning file contains a list of software packages to boot, mount, and run. The ISO file system in each installed package is mounted to the root file system directly from flash.

Note The packages and provisioning file used to boot in installed mode must reside in flash. Booting in installed mode from usbflash0: or tftp: is not supported.

Related Topics

Displaying Software Install: Examples, on page 59

Emergency Installation: Example, on page 60

Bundle Boot Mode

You can boot your controller in bundle boot mode by booting the bundle (.bin) file:

```
controller:  boot flash:ct5700-ipservicesk9.SSA.03.09.07.EMP.150-9.07.EMP.bin
```

The provisioning file contained in a bundle is used to decide which packages to boot, mount, and run. Packages are extracted from the bundle and copied to RAM. The ISO file system in each package is mounted to the root file system.

Unlike install boot mode, additional memory that is equivalent to the size of the bundle is used when booting in bundle mode.

Note Auto install and smart install functionality is not supported in bundle boot mode.

Note The AP image predownload feature is not supported in bundle boot mode.

Related Topics

Displaying Software Install: Examples, on page 59

Emergency Installation: Example, on page 60

Controller Information Assignment

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

Use the controller 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.

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 controller receives the dynamically assigned IP address and reads the configuration file.

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

DHCP-Based Autoconfiguration Overview

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

During DHCP-based autoconfiguration, your controller (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 controller. However, you need to configure the DHCP server for various lease options associated with IP addresses. If you are using DHCP to relay the configuration file location on the network, you might also need to configure a Trivial File Transfer Protocol (TFTP) server and a Domain Name System (DNS) server.

The DHCP server for your controller can be on the same LAN or on a different LAN than the controller. If the DHCP server is running on a different LAN, you should configure a DHCP relay device between your controller 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.

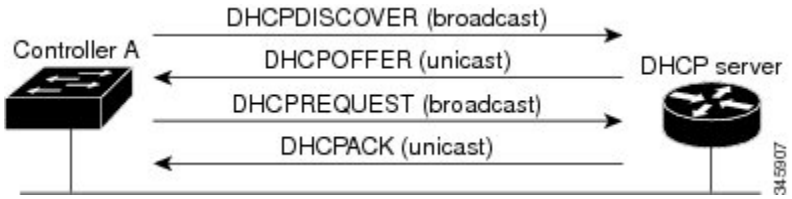
DHCP-based autoconfiguration replaces the BOOTP client functionality on your controller.

DHCP Client Request Process

When you boot up your controller, the DHCP client is invoked and requests configuration information from a DHCP server when the configuration file is not present on the controller. 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 2: DHCP Client and Server Message Exchange



The client, Controller 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 controller 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 controller accepts replies from a BOOTP server and configures itself, the controller broadcasts, instead of unicasts, TFTP requests to obtain the controller configuration file.

The DHCP hostname option allows a group of controllers to obtain hostnames and a standard configuration from the central management DHCP server. A client (controller) 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 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 controller by the controller hardware address.
- If you want the controller 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 controller) (required)
- If you want the controller 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 controller 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 controller is not configured. If the router IP address or the TFTP server name are not found, the controller might send broadcast, instead of unicast, TFTP requests. Unavailability of other lease options does not affect autoconfiguration.
- The controller can act as a DHCP server. By default, the Cisco IOS DHCP server and relay agent features are enabled on your controller but are not configured.

Purpose of the TFTP Server

Based on the DHCP server configuration, the controller attempts to download one or more configuration files from the TFTP server. If you configured the DHCP server to respond to the controller 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 controller 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 controller 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 controller'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 controller 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 controller 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 controllers. 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 controller, or if it is to be accessed by the controller 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 controller.

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

The DNS server can be on the same or on a different LAN as the controller. If it is on a different LAN, the controller 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 controller obtains its configuration information in these ways:

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

  The controller receives its IP address, subnet mask, TFTP server address, and the configuration filename from the DHCP server. The controller 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 controller, but the TFTP server address is not provided in the DHCP reply (one-file read method).

  The controller receives its IP address, subnet mask, and the configuration filename from the DHCP server. The controller 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 controller and provided in the DHCP reply. The configuration filename is not provided (two-file read method).

  The controller receives its IP address, subnet mask, and the TFTP server address from the DHCP server. The controller 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 controller reads the cisconet.cfg file.)

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

After obtaining its hostname from the default configuration file or the DHCP reply, the controller 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 controller cannot read the network-confg, cisconet.cfg, or the hostname file, it reads the router-confg file. If the controller cannot read the router-confg file, it reads the ciscortr.cfg file.

Note The controller 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 controller, you enter the boot loader mode only through the console connection configured for 9600 bps. Unplug the controller power cord, and press the **Mode** button while reconnecting the power cord. You can release the **Mode** button after all the amber system LEDs turn on and remain solid. Then the boot loader controller prompt appears.

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

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

Each line in these files contains an environment variable name and an equal sign followed by the value of the variable. A variable has no value if it is not 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.

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.

Scheduled Reload of the Software Image

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

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

If you modify your configuration file, the controller 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 Controller Setup Configuration

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

Configuring DHCP Autoconfiguration (Only Configuration File)

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

SUMMARY STEPS

1. **configure terminal**
2. **ip dhcp** *poolname*
3. **bootfile** *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:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**ip dhcp** *poolname* **Example:** `Controller(config)# ip dhcp pool pool`	Creates a name for the DHCP Server address pool, and enters DHCP pool configuration mode.
Step 3	**bootfile** *filename* **Example:** `Controller(dhcp-config)# bootfile 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:** `Controller(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:** `Controller(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:** `Controller(dhcp-config)# option 150 10.10.10.1`	Specifies the IP address of the TFTP server.
Step 7	**exit** **Example:** `Controller(dhcp-config)# exit`	Returns to global configuration mode.

	Command or Action	Purpose
Step 8	**tftp-server flash**:*filename.text* **Example:** `Controller(config)# tftp-server flash:config-boot.text`	Specifies the configuration file on the TFTP server.
Step 9	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet1/0/4`	Specifies the address of the client that will receive the configuration file.
Step 10	**no switchport** **Example:** `Controller(config-if)# no switchport`	Puts the interface into Layer 3 mode.
Step 11	**ip address** *address mask* **Example:** `Controller(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:** `Controller(config-if)# end`	Returns to privileged EXEC mode.

Manually Assigning IP Information to Multiple SVIs

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

Note

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:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**interface vlan** *vlan-id* **Example:** `Controller(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:** `Controller(config-vlan)# ip address 10.10.10.1 255.255.255.0`	Enters the IP address and subnet mask.
Step 4	**exit** **Example:** `Controller(config-vlan)# exit`	Returns to global configuration mode.
Step 5	**ip default-gateway** *ip-address* **Example:** `Controller(config)# ip default-gateway 6.100.55.255`	Enters the IP address of the next-hop router interface that is directly connected to the controller where a default gateway is being configured. The default gateway receives IP packets with unresolved destination IP addresses from the controller. Once the default gateway is configured, the controller has connectivity to the remote networks with which a host needs to communicate. **Note** When your controller is configured to route with IP, it does not need to have a default gateway set.

	Command or Action	Purpose
		Note The controller capwap relays on default-gateway configuration to support routed access point join the controller.
Step 6	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 7	**show interfaces vlan** *vlan-id* **Example:** `Controller# show interfaces vlan 99`	Verifies the configured IP address.
Step 8	**show ip redirects** **Example:** `Controller# show ip redirects`	Verifies the configured default gateway.

Modifying the Controller Startup Configuration

Specifying the Filename to Read and Write the System Configuration

By default, the Cisco IOS software uses the file `config.text` 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 controller for this task.

SUMMARY STEPS

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

Booting the Controller in Installed Mode

SUMMARY STEPS

1. **cp** *source_file_path destination_file_path*
2. **software expand file** *source_file_path*
3. **reload**
4. **boot flash:packages.conf**
5. **show version**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**cp** *source_file_path destination_file_path* **Example:** `Controller: cp tftp://10.0.0.2/ct5700-ipservicesk9.SSA.03.09.07.EMP.150-9.07.EMP.bin to flash`	(Optional) Copies the bin file (`image.bin`) from the FTP or TFTP server to flash or USB flash.

Step 2	**software expand file** *source_file_path*
	
	Example:
	Expanding the bin file from Flash:
	
	`Controller# software expand file flash:ct5700-ipservicesk9.SSA.03.09.26.EXP.150-9.26.EXP.bin`
	`Controller# $flash:ct5700-ipservicesk9.SSA.03.09.26.EXP.150-9.26.EXP.bin`
	`Preparing expand operation ...`
	`[1]: Expanding bundle flash:ct5700-ipservicesk9.SSA.03.09.26.EXP.150-9.26.EXP.bin`
	`[1]: Copying package files`
	`[1]: Package files copied`
	`[1]: Finished expanding bundle flash:ct5700-ipservicesk9.SSA.03.09.26.EXP.150-9.26.EXP.bin`
	
	`138504 -rwx 237922344 Nov 29 2012 14:53:57 +00:00 ct5700-ipservicesk9.SSA.03.09.26.EXP.150-9.26.EXP.bin`
	`154743 -rwx 78911772 Dec 3 2012 15:18:16 +00:00 ct5700-base.SSA.03.09.26.EXP.pkg`
	`154744 -rwx 2269876 Dec 3 2012 15:18:20 +00:00 ct5700-drivers.SSA.03.09.26.EXP.pkg`
	`154745 -rwx 29854608 Dec 3 2012 15:18:16 +00:00 ct5700-infra.SSA.03.09.26.EXP.pkg`
	`154746 -rwx 43072360 Dec 3 2012 15:18:18 +00:00 ct5700-iosd-ipservicesk9.SSA.150-9.26.EXP.pkg`
	`154747 -rwx 22020832 Dec 3 2012 15:18:17 +00:00 ct5700-platform.SSA.03.09.26.EXP.pkg`
	`154742 -rwx 1207 Dec 3 2012 15:18:38 +00:00 packages.conf`
	`154748 -rwx 61788880 Dec 3 2012 15:18:20 +00:00 ct5700-wcm.SSA.03.09.26.EXP.pkg`
		Expands the bin file stored in flash, FTP, TFTP, HTTP, or HTTPS server on the booted controller.
		Note Ensure that the `packages.conf` file is available in the expanded list.

Step 3	**reload** **Example:** `Controller: reload`	Reloads the controller. **Note** You can boot the controller manually or automatically using the `packages.conf` file. If you are booting manually, you can proceed to Step 4. Otherwise, the controller boots up automatically.
Step 4	**boot flash:packages.conf** **Example:** `switch: boot flash:packages.conf`	Boots the controller with the `packages.conf` file.

	Command or Action	Purpose

Step 5	**show version**
	
	Example:
	`controller# show version`
	
	`Switch Ports Model SW Version SW Image Mode`
	`------ ----- ----- ---------- ---------- ----`
	`1 6 WS-C5700-6DS-S 03.09.26.EXP ct5700-k9 INSTALL`
		Verifies that the controller is in the **INSTALL** mode.

Booting the Controller in Bundle Mode

There are several methods by which you can boot the controller—either by copying the bin file from the TFTP server and then boot the controller, or by booting the controller straight from flash or USB flash using the commands **boot flash:<image.bin>** or **boot usbflash0:<image.bin>** .

The following procedure explains how to boot the controller from th TFTP server in the bundle mode.

SUMMARY STEPS

1. **cp** *source_file_path destination_file_path*
2. **controller:BOOT=<source path of .bin file>**
3. **boot**
4. **show version**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**cp** *source_file_path destination_file_path* **Example:** `Controller: cp tftp://10.0.0.2/ct5700-ipservicesk9.SSA.03.09.07.EMP.150-9.07.EMP.bin to flash`	(Optional) Copies the bin file (`image.bin`) from the FTP or TFTP server to flash or USB flash.

Step 2	**controller:BOOT=<source path of .bin file>**
	
	Example:
	`Controller:`
	`controller:BOOT=tftp://10.0.0.2/ct5700-ipservicesk9.SSA.03.09.07.EMP.150-9.07.EMP.bin`
		Sets the boot parameters.

Step 3	**boot** **Example:** `switch: boot`	Boots the controller.

	Command or Action	Purpose

Step 4	**show version**
	
	Example:
	`controller# show version`
	`Switch Ports Model SW Version SW Image Mode`
	`------ ----- ----- ---------- ---------- ----`
	`1 6 WS-C5700-6DS-S 03.09.40.EXP ct5700-k9 BUNDLE`
		Verifies that the controller is in the **BUNDLE** mode.

Configuring a Scheduled Software Image Reload

Before You Begin

This task describes how to configure your controller 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:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**copy running-config startup-config** **Example:** `copy running-config startup-config`	You should save your controller configuration information to the startup configuration before using the **reload** command.
Step 3	**reload in** [*hh*:]*mm* [*text*] **Example:** `Controller(config)# reload in 12`	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. System configuration has been modified. Save? [yes/no]: **y**
Step 4	**reload at***hh*: *mm* [*month day* | *day month*] [*text*]	Specifies the time in hours and minutes for the reload to occur.

	Command or Action	Purpose
	Example: `Controller(config)# reload at 14:00`	**Note** Use the **at** keyword only if the controller 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 controller. To schedule reloads across several controllers to occur simultaneously, the time on each controller must be synchronized with NTP.
Step 5	**reload cancel** **Example:** `Controller(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 controller.

Monitoring Controller Setup Configuration

Verifying the Controller Running Configuration: Example

```
Controller# 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 gigabitethernet6/0/1
 no switchport
 ip address 172.20.137.50 255.255.255.0
!
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
```

Displaying Software Install: Examples

This example displays software bootup in install mode:

```
switch: boot flash:packages.conf

Getting rest of image
Reading full image into memory....done
Reading full base package into memory...: done = 74596432
Nova Bundle Image
--------------------------------------
Kernel Address : 0x6042f354
Kernel Size : 0x318412/3245074
Initramfs Address : 0x60747768
Initramfs Size : 0xdc08e8/14420200
Compression Format: .mzip

Bootable image at @ ram:0x6042f354
Bootable image segment 0 address range [0x81100000, 0x81b80000] is in range [0x80180000,
0x90000000].
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@boot_system:
 377
Loading Linux kernel with entry point 0x811060f0 ...
Bootloader: Done loading app on core_mask: 0xf

### Launching Linux Kernel (flags = 0x5)

All packages are Digitally Signed
Starting System Services
Nov 7 09:57:05 %IOSXE-1-PLATFORM: process stack-mgr: %STACKMGR-1-DISC_START: Switch 2 is
starting stack discovery
#########################################################################################################################################################
Nov 7 09:59:07 %IOSXE-1-PLATFORM: process stack-mgr: %STACKMGR-1-DISC_DONE: Switch 2 has
finished stack discovery
Nov 7 09:59:07 %IOSXE-1-PLATFORM: process stack-mgr: %STACKMGR-1-SWITCH_ADDED: Switch 2 has
 been added to the stack
Nov 7 09:59:14 %IOSXE-1-PLATFORM: process stack-mgr: %STACKMGR-1-ACTIVE_ELECTED: Switch 2
has been elected ACTIVE

Restricted Rights Legend

Use, duplication, or disclosure by the Government is
subject to restrictions as set forth in subparagraph
(c) of the Commercial Computer Software - Restricted
Rights clause at FAR sec. 52.227-19 and subparagraph
(c) (1) (ii) of the Rights in Technical Data and Computer
Software clause at DFARS sec. 252.227-7013.

cisco Systems, Inc.
170 West Tasman Drive
San Jose, California 95134-1706



Cisco IOS Software, IOS-XE Software, Catalyst L3 Switch Software (CAT3K_CAA-UNIVERSALK9-M),
 Version 03.09.12.EMD EARLY DEPLOYMENT ENGINEERING NOVA_WEEKLY BUILD, synced to
DSGS_PI2_POSTPC_FLO_DSBU7_NG3K_1105
Copyright (c) 1986-2012 by Cisco Systems, Inc.
Compiled Sun 04-Nov-12 22:53 by gereddy
License level to iosd is ipservices
```

Related Topics

Software Boot Modes, on page 43

Installed Boot Mode, on page 43

Bundle Boot Mode, on page 43

Emergency Installation: Example

This sample output is an example when the **emergency-install** boot command is initiated:

```
switch: emergency-install tftp://172.20.249.254/katana/ct5760.renum.bin

The bootflash will be erased during install operation, continue (y/n)?y
Starting emergency recovery (tftp://172.20.249.254/katana/ct5760.renum.bin)...
Loading "sda9:ct5760-recovery.bin"...
Reading full image into memory.....................done
Verifying image digital signature.
Nova Bundle Image
-------------------------------------
Kernel Address    : 0x8b35b598
Kernel Size       : 0x367550/3568976
Initramfs Address : 0x8b6c2ae8
Initramfs Size    : 0xbfe484/12575876
Compression Format: unknown


File "sda9:ct5760-recovery.bin" uncompressed and installed, entry point: 0x8b35b598
Image validated
\ufffd

Initiating Emergency Installation of bundle tftp://172.20.249.254/katana/ct5760.renum.bin


Downloading bundle tftp://172.20.249.254/katana/ct5760.renum.bin...

Validating bundle tftp://172.20.249.254/katana/ct5760.renum.bin...
Installing bundle tftp://172.20.249.254/katana/ct5760.renum.bin...
Verifying bundle tftp://172.20.249.254/katana/ct5760.renum.bin...
Package ct5760-base.SPA.03.02.00.pkg is Digitally Signed
Package ct5760-drivers.SPA.03.02.00.SE.pkg is Digitally Signed
Package ct5760-infra.SPA.03.02.00.pkg is Digitally Signed
Package ct5760-iosd-ipservicesk9.SPA.150-1.EX.pkg is Digitally Signed
Package ct5760-platform.SPA.03.02.00.SE.pkg is Digitally Signed
Package ct5760-wcm.SPA.10.0.10.48.pkg is Digitally Signed
Preparing flash...
Syncing device...
Emergency Install successful... Rebooting
Restarting system.
Xmodem file system is available.
Base ethernet MAC Address: 20:37:06:4d:64:00
Verifying bootloader digital signature.

The system is not configured to boot automatically.  The
following command will finish loading the operating system
software:

    boot
```

Related Topics

Software Boot Modes, on page 43

Installed Boot Mode, on page 43
Bundle Boot Mode, on page 43

Configuration Examples for Performing Controller Setup

Configuring a Controller to Download Configurations from a DHCP Server: Example

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

```
Controller# configure terminal
Controller(conf)# boot host dhcp
Controller(conf)# boot host retry timeout 300
Controller(conf)# banner config-save ^C Caution - Saving Configuration File to NVRAM May
Cause You to No longer Automatically Download Configuration Files at Reboot^C
Controller(config)# vlan 99
Controller(config-vlan)# interface vlan 99
Controller(config-if)# no shutdown
Controller(config-if)# end
Controller# 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)
Controller#
```

Scheduling Software Image Reload: Examples

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

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

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

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

Additional References For Performing Controller Setup

Related Documents

Related Topic	Document Title
Controller setup commands Boot loader commands	*System Management Command Reference (Cisco WLC 5700 Series)*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

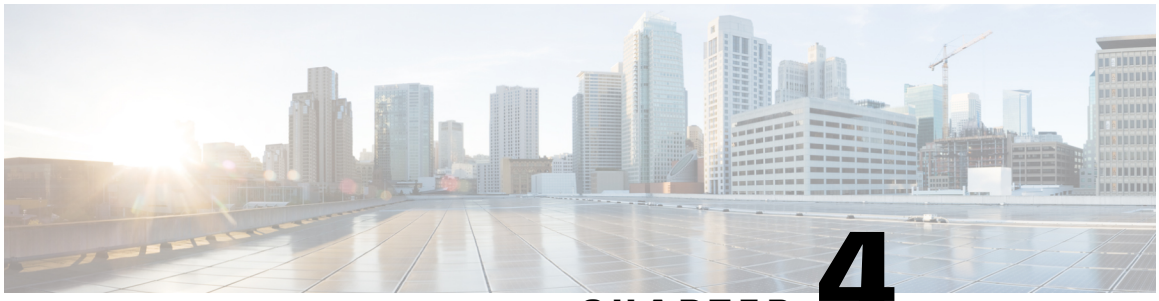
MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

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 Controller Setup Configuration

Release	Modification
	This feature was introduced.



CHAPTER 4

Configuring Right-To-Use Licenses

This module contains the following topics:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Right-To-Use AP-Count Licenses

The following are the restrictions you must keep in mind when using license for the Cisco 5700 Series Wireless Controller:

- The license you have purchased is applicable only for the Cisco 5700 Series Wireless Controller. You cannot use the same license for the earlier version of the Cisco 5700 Series Wireless Controllers.
- The CLI commands that you run for the Cisco 5700 Series Wireless Controller are applicable only for these controllers. You cannot run these commands for the earlier version of the controllers.

Information About Configuring RTU Licenses

Right-To-Use AP-Count Licensing

Right-to-use licensing (RTU) allows you to order and activate a specific license type, and then to manage license usage on your Cisco 5700 Series Wireless Controller .

You can order your controller with support for any number of access points as the adder access point count licenses but the total number of the licenses ordered should not exceed 1000. You can also order the adder access point count licenses after receiving the controller.

For example, if you have ordered 700 new adder licenses, you can add only those 700 adder licenses to the controller. The licenses can be added in the increments of 1, but the total number of licenses added for the controller should not exceed 1000.

You can configure controller to manage the access point count licenses from the CLI and view the number of access points currently in use from both the CLI and GUI.

The following are the two different types of access point licenses:

1 Permanent licenses for the access points

- Adder access point count license—You can purchase the adder license to increase the controller capacity at the later point of time. You can transfer the adder access point count license from one controller to another.

2 Evaluation licenses for the access points

- You can activate the evaluation licenses to evaluate more access point count licenses before purchasing.
- Maximum number of access points that can be evaluated is 1000.
- The evaluation period for trying for the access point licenses is 90 days.
- You can activate and deactivate the evaluation license for the access points from the CLI.

Right-to-Use AP-Count Evaluation Licenses

If you are considering upgrading to a license with a higher access point count, you can try an evaluation license before upgrading to a permanent version of the license. For example, if you are using a permanent license with a 50-access-point count and want to try an evaluation license with a 100-access-point count, you can try out the evaluation license for 90 days.

When an evaluation license is activated, the permanent AP-count licenses are ignored. The maximum supported licenses of 1000 access points are available for 90 days .

To prevent disruptions in operation, the controller does not change licenses when an evaluation license expires. A warning expiry message is displayed daily starting five days prior to the expiry date. After 90 days, the evaluation license expires with a warning message. You have to disable the evaluation license and then purchase the permanent license.

When the controller reboots after the evaluation license expiry, the license defaults to a permanent license.

Right-To-Use Adder AP-Count Rehosting Licenses

Revoking a license from one device and installing it on another is called rehosting. You might want to rehost a license in order to change the purpose of a device . For example, if you want to move your Office Extend or indoor access points to a different controller, you could transfer the adder ap-count license from one controller to another.

In order to rehost a license, you must deactivate the adder ap-count license from one device and activate the same license on another device.

Evaluation licenses cannot be rehosted.

How to Configure RTU Licenses

Activating an AP-Count Evaluation License (CLI)

When an evaluation license is activated, the maximum supported ap-count licenses are made available. A maximum of 1000 access points can be evaluated for 90 days by enabling the evaluation ap-count licenses.

SUMMARY STEPS

1. **license right-to-use activate ap-count evaluation**
2. **show license right-to-use summary**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**license right-to-use activate ap-count evaluation** **Example:** `Controller# license right-to-use activate ap-count evaluation`	Enables the ap-count evaluation licenses on the controller. By default during activation, the EULA gets displayed. If the **acceptEULA** is passed, the EULA content is not displayed, and you can activate the evaluation license. This option is useful for automation and scripting.
Step 2	**show license right-to-use summary** **Example:** `Controller# show license right-to-use summary`	Verifies that the evaluation license is activated on the controller.

Activating an AP-Count Permanent License

You can deactivate an evaluation ap-count license and activate the permanent ap-count license on the controller.

After activating ap-count permanent or adder license, if the **show license right-to-use summary** command still shows evaluation ap-count licenses, then you have to deactivate the previously used evaluation license

that was not deactivated earlier. Deactivate the evaluation license to enable permanent or adder ap-count licenses.

SUMMARY STEPS

1. **license right-to-use deactivate ap-count evaluation**
2. **show license right-to-use summary**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**license right-to-use deactivate ap-count evaluation** **Example:** `Controller# license right-to-use deactivate ap-count evaluation`	Deactivates particular evaluation license level and activates the permanent ap-count licenses on the controller.
Step 2	**show license right-to-use summary** **Example:** `Controller# show license right-to-use summary`	Verifies the number of permanent ap-count licenses activated on the controller.

Obtaining an Upgrade or Capacity Adder License

You can use the capacity adder licenses to increase the number of access points supported by the controller.

SUMMARY STEPS

1. **license right-to-use activate ap-count** *ap-number* **slot** *1*
2. **show license right-to-use summary**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**license right-to-use activate ap-count** *ap-number* **slot** *1* **Example:** `Controller# license right-to-use activate ap-count 500 slot 1`	Obtains an upgrade or increases the license capacity by adding new adder licenses.
Step 2	**show license right-to-use summary** **Example:** `Controller# show license right-to-use summary`	Verifies the number of permanent ap-count licenses activated on the controller.

Transferring Licenses to a Replacement Controller after an RMA

The replacement controller comes with same permanent ap-count licences.

SUMMARY STEPS

1. **license right-to-use deactivate ap-count** *count* **slot** *1* **acceptEULA**
2. **license right-to-use activate ap-count** *count* **slot** *1* **acceptEULA**
3. **show license right-to-use summary**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**license right-to-use deactivate ap-count** *count* **slot** *1* **acceptEULA** **Example:** `Controller# license right-to-use deactivate ap-count 55 slot 1 acceptEULA`	Deactivates the permanent ap-count licenses on earlier controller to be replaced.
Step 2	**license right-to-use activate ap-count** *count* **slot** *1* **acceptEULA** **Example:** `Controller# license right-to-use activate ap-count 55 slot 1 acceptEULA`	Activates the same permanent ap-count licenses on the replacement controller.
Step 3	**show license right-to-use summary** **Example:** `Controller# show license right-to-use summary`	Verifies the number of ap-count licenses points activated on the replacement controller.

Monitoring and Maintaining RTU Licenses

Viewing Right-To-Use AP-Count Licenses (GUI)

You can view the details of access point licenses installed on the controller using the **Licenses** and **License Usage** pages from the controller GUI.

Step 1 Choose **Administration** > **Software Activation** > **Licenses**.

Example:
The **Licenses** page appears.

This page lists all of the access point licenses installed on the controller. You can view the following details for the license:

- License name
- License type (adder, permanent, or evaluation)
- Count (the maximum number of access points allowed for this license)
- Period left for the license

In addition, you can view the following details of the AP-Count Licenses:

- Whether the ap-count license is enabled or not
- The maximum number of access points allowed for this license
- The number of access points currently using this license
- The remaining ap-count licenses

Step 2 Choose **Administration** > **Software Activation** > **License Usage**.

Example:
The **License Usage** page appears.

In the **License Usage** page, you can view the consolidated list of all the licenses based on their usage duration, use of the license, and the end-user license agreement (EULA) acceptance state.

Step 3 Choose **Administration** > **Software Activation** > **Eula** > **Adder**, **Evaluation**, or **Permanent**.

Example:
The **Eula** page for the selected license appears.

You can read the terms and the conditions for the **Adder**, **Evaluation**, or **Permanent** License.

Viewing Right-To-Use AP-Count Licenses (CLI)

You can view the detailed information of ap-count licenses installed on the controller using the **show license right-to-use** commands from the controller CLI.

SUMMARY STEPS

1. **show license right-to-use detail**
2. **show license right-to-use detail** | *output modifiers*
3. **show license right-to-use eula**
4. **show license right-to-use**
5. **show license right-to-use usage**
6. **show license right-to-use** | *output modifiers*
7. **show license right-to-use summary**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show license right-to-use detail** **Example:** `Controller# show license right-to-use detail`	Displays details of all the licenses installed on the stack 1.

Step 2	**show license right-to-use detail** | *output modifiers*
	
	Example:
	`Controller# show license right-to-use detail | append <url>`
	
	Example:
	`Controller# show license right-to-use detail | begin apcount`
	
	Example:
	`Controller# show license right-to-use detail | count ap`
	
	Example:
	`Controller# show license right-to-use detail | exclude ap`
	
	Example:
	`Controller# show license right-to-use detail | format <spec file location>`
	
	Example:
	`Controller# show license right-to-use detail | include apcount`
	
	Example:
	`Controller# show license right-to-use detail | redirect <url>`
	
	Example:
	`Controller# show license right-to-use detail | section include apcount`
	`Controller# show license right-to-use detail | section exclude apcount`
	`Controller# show license right-to-use detail | section license`
	
	Example:
	`Controller# show license right-to-use detail | tee <url>`
		Displays the details of licenses based on the filtered search using output modifiers such as append, begin, count, exclude, format, include, redirect, section, and tee.
		
		Appends the redirected license information output to URL. The URLs supporting append operation are crash information, flash, FTP, HTTP, HTTPS, NVRAM, RCP, SCP, TFTP, UNIX, and USB flash0.
		
		Displays the license information that begins with the lines which match the regular expression.
		
		Displays the number of lines that match the regular expression.
		
		Displays the details of license information that excludes lines which match the regular expression.
		
		Displays the details of license information based on the format specified in the spec file.
		
		Displays the lines that match the regular expression.
		
		Redirects the license information output to URL. The URLs that support the redirect operation are crash information, flash, FTP, HTTP, HTTPS, NVRAM, RCP, SCP, TFTP, UNIX, and USB flash0.
		
		Filters the section of the license information output based on the include, exclude, or other regular expression options specified.
		
		Copies the license information output to URL. The URLs that support the copy operation are crash information, flash, FTP, HTTP, HTTPS, NVRAM, RCP, SCP, TFTP, UNIX, and USB flash0.

	Command or Action	Purpose

Step 3	**show license right-to-use eula**
	
	Example:
	`Controller# show license right-to-use eula adder`
	`Controller# show license right-to-use eula evaluation`
	`Controller# show license right-to-use eula permanent`
		Displays the EULA content for the adder, evaluation, and permanent AP-count licenses.

Step 4	**show license right-to-use** **Example:** `Controller# show license right-to-use`	Displays the licenses that got activated with EULA.
Step 5	**show license right-to-use usage** **Example:** `Controller# show license right-to-use usage`	Displays the usage details of all licenses.
Step 6	**show license right-to-use |** *output modifiers* **Example:** `Controller# show license right-to-use | append`	Displays the details of licenses based on the filtered search using output modifiers such as append, begin, count, exclude, format, include, redirect, section, and tee.
Step 7	**show license right-to-use summary** **Example:** `Controller# show license right-to-use summary`	Displays the summary of licenses that are currently in use.

Examples: RTU Licenses Configuration

This examples shows how to activate ap-count evaluation license:

```
Controller# license right-to-use activate ap-count evaluation
Controller# show license right-to-use summary
```

This examples shows how to activate ap-count permanent license:

```
Controller# license right-to-use deactivate ap-count evaluation
Controller# show license right-to-use summary
```

This examples shows how to obtain an upgrade or adder license:

```
Controller# license right-to-use activate ap-count 700 slot 1
Controller# show license right-to-use summary
```

This example shows how to transfer licenses to a replacement controller after an RMA. Deactivate the licenses from the controller to be replaced and activate or add the same number of licenses in the replacement controller:

```
Controller# license right-to-use deactivate ap-count 250 slot 1
```

```
Controller# license right-to-use activate ap-count 250 slot 1
Controller# show license right-to-use summary
```

Additional References for RTU Licensing

Related Documents

Related Topic	Document Title
RTU commands	*System Management Command Reference (Cisco WLC 5700 Series)*
RTU AP image preload feature	*System Management Configuration Guide (Cisco WLC 5700 Series)*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

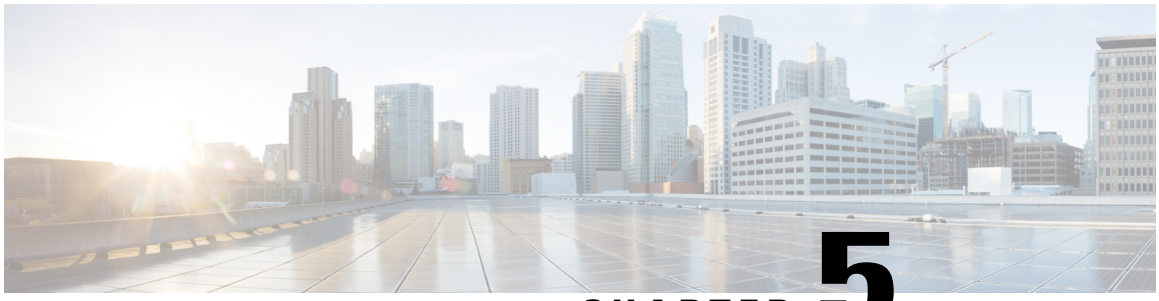
MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

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 RTU Licensing

Release	Modification
	This feature was introduced.



CHAPTER 5

Configuring Administrator Usernames and Passwords

This module contains the following topics:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Configuring Administrator Usernames and Passwords

You can configure administrator usernames and passwords to prevent unauthorized users from reconfiguring the controller and viewing configuration information. This section provides instructions for initial configuration and for password recovery.

You can also set administrator usernames and passwords to manage and configure one or more access points that are associated with the controller.

Strong Passwords

You can set strong administrator passwords such as encrypted passwords with ASCII keys for the administrator user for managing access points.

Use the following guidelines while creating strong passwords:

- There should be at least three of the following categories—lowercase letters, uppercase letters, digits, and special characters.
- The new password should not be the same as that of the associated username and the username should not be reversed.
- The characters in the password should not be repeated more than three times consecutively.
- The password should not be **cisco**, **ocsic**, **admin**, **nimda**, or any variant obtained by changing the capitalization of letters therein, or by substituting "1" "|" or "!" for i, and/or substituting "0" for "o", and/or substituting "$" for "s".
- The maximum number of characters accepted for the username and password is 32.

Encrypted Passwords

You can set three types of keys for the password:

- Randomly generated key—This key is generated randomly and it is the most secure option. To export the configuration file from one system to another, the key should also be exported.
- Static key—The simplest option is to use a fixed (static) encryption key. By using a fixed key, no key management is required, but if the key is somehow discovered, the data can be decrypted by anyone with the knowledge of that key. This is not a secure option and it is called obfuscation in the CLI.
- User defined key—You can define the key by yourself. To export the configuration file from one system to another, both systems should have the same key configured.

Configuring Administrator Usernames and Passwords

SUMMARY STEPS

1. **configure terminal**
2. **wireless security strong-password**
3. **username** *admin-username* **password {0** *unencrypted_password* **| 7** *hidden_password* **|** *unencrypted_text***}**
4. **username** *admin-username* **secret {0** *unencrypted_secret_text* **| 4** *SHA256 encrypted_secret_text* **| 5** *MD5 encrypted_secret_text* **| LINE}**
5. **ap mgmtuser username** *username* **password {0** *unencrypted password* **| 8** *AES encrypted password* **}secret {0** *unencrypted password* **| 8** *AES encrypted password* **}**
6. **ap dot1x username** *username* **password {0** *unencrypted password* **| 8** *AES encrypted password* **}**
7. **end**
8. **ap name apname mgmtuser username** *username***password** *password* **secret** *secret _text*
9. **ap name apname dot1x-user** *username* **password** *password*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**wireless security strong-password** **Example:** `Controller(config)# wireless security strong-password`	Enables strong password policy for the administrator user.
Step 3	**username** *admin-username* **password {0** *unencrypted_password* **| 7** *hidden_password* **|** *unencrypted_text***}** **Example:** `Controller(config)# username adminuser1 password 0 QZsek239@`	Specifies a username and password for an administrator. The administrator can configure the controller and view the configured information.
Step 4	**username** *admin-username* **secret {0** *unencrypted_secret_text* **| 4** *SHA256 encrypted_secret_text* **| 5** *MD5 encrypted_secret_text* **| LINE}** **Example:** `Controller(config)# username adminuser1 secret 0 QZsek239@`	Specifies the secret for the administrator.
Step 5	**ap mgmtuser username** *username* **password {0** *unencrypted password* **| 8** *AES encrypted password* **}secret {0** *unencrypted password* **| 8** *AES encrypted password* **}** **Example:** `Controller(config)# ap mgmtuser username cisco password 0 Qwci12@ secret 0 Qwci14@!`	Specifies administrator username and password for managing all the access points configured to the controller. You can also include the secret text to perform privileged access point management. **Note** If your password is not strong enough to fulfill the strong password policy, then the password is rejected with a valid error message. For example, the following password is rejected because it is not a strong password. `Controller# ap mgmtuser username cisco password 0 abcd secret 0 1234`
Step 6	**ap dot1x username** *username* **password {0** *unencrypted password* **| 8** *AES encrypted password* **}** **Example:** `Controller(config)# ap dot1x username cisco password 0 Qwci12@`	Specifies the 802.1X username and password for managing all the access points configured to the controller.
Step 7	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

	Command or Action	Purpose
Step 8	**ap name apname mgmtuser username** *username***password** *password* **secret** *secret _text* **Example:** `Controller# ap name APf0f7.55c7.7b23 mgmtuser username cisco password Qne35! secret Nzep592$`	Configures the administrator username, password, and secret text for managing a specific access point that is configured to the controller.
Step 9	**ap name apname dot1x-user** *username* **password** *password* **Example:** `Controller# ap name APf0f7.55c7.7b23 dot1x-user username cisco password Qne35!`	Configures the 802.1X username and password for a specific access point.

Examples: Administrator Usernames and Passwords Configuration

This example shows how to configure administrator usernames and passwords with the strong password policy in configuration mode:

```
Controller# configure terminal
Controller(config)# wireless security strong-password
Controller(config)# username adminuser1 password 0 QZsek239@
Controller(config)# ap mgmtuser username cisco password 0 Qwci12@ secret 0 Qwci14@!
Controller(config)# ap dot1x username cisco password 0 Qwci12@
Controller# end
```

This example shows how to configure administrator usernames and passwords for an access point in global EXEC mode:

```
Controller# wireless security strong-password
Controller# ap name APf0f7.55c7.7b23 mgmtuser username cisco password Qwci12@ secret Qwci14@
Controller# ap name APf0f7.55c7.7b23 dot1x-user username cisco password Qwci12@
Controller# end
```

Additional References for Administrator Usernames and Passwords

Related Documents

Related Topic	Document Title
System management commands	*System Management Command Reference Guide (Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

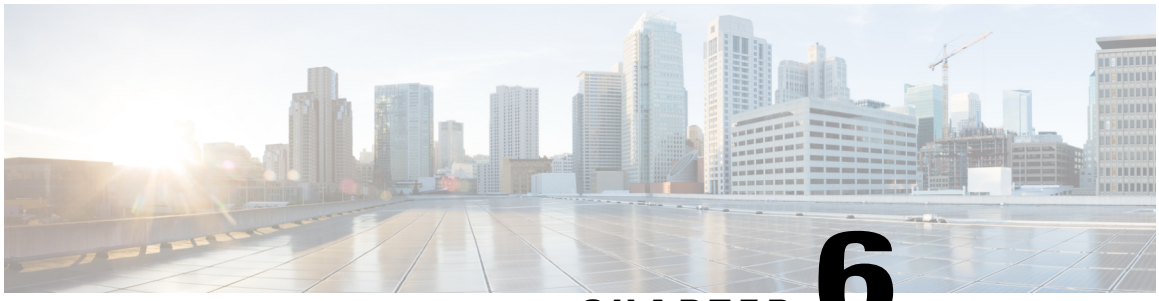
MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

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 Administrator Usernames and Passwords Configuration

Release	Modification
Cisco IOS XE Release 3.2SE	This feature was introduced.



CHAPTER 6

Configuring 802.11 parameters and Band Selection

This module contains the following topics:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 on Band Selection, 802.11 Bands, and Parameters

- Band-selection enabled WLANs do not support time-sensitive applications like voice and video because of roaming delays.
- Band selection can be used only with Cisco Aironet 1040, 1140, 1260, 3500, and 3600 Series access points.

- Band selection operates only on access points that are connected to a controller.
- The band-selection algorithm directs dual-band clients only from the 2.4-GHz radio to the 5-GHz radio of the same access point, and it only runs on an access point when both the 2.4-GHz and 5-GHz radios are up and running.
- You can enable both band selection and aggressive load balancing on the controller. They run independently and do not impact one another.
- It is not possible to enable or disable band selection and client load balancing globally through the controller GUI or CLI. You can, however, enable or disable band selection and client load balancing for a particular WLAN. Band selection and client load balancing are enabled globally by default.

Information About Configuring Band Selection, 802.11 Bands, and Parameters

802.11 Bands

You can configure the 802.11b/g/n (2.4-GHz) and 802.11a/n (5-GHz) bands for the controller to comply with the regulatory requirements in your country. By default, both 802.11b/g/n and 802.11a/n are enabled.

When a controller is configured to allow only 802.11g traffic, 802.11b client devices are able to successfully connect to an access point but cannot pass traffic. When you configure the controller for 802.11g traffic only, you must mark 11g rates as mandatory.

802.11n Parameters

You can manage 802.11n devices such as the Cisco Aironet 1140 Series Access Points on your network. The 802.11n devices support the 2.4- and 5-GHz bands and offer high-throughput data rates.

The 802.11n high-throughput rates are available on 1140 series access points for WLANs using WMM with no Layer 2 encryption or with WPA2/AES encryption enabled.

802.11h Parameters

802.11h informs client devices about channel changes and can limit the transmit power of those client devices.

How to Configure 802.11 Bands and Parameters

Configuring Band Selection (CLI)

SUMMARY STEPS

1. **configure terminal**
2. **wireless client band-select cycle-count** *cycle_count*
3. **wireless client band-select cycle-threshold** *milliseconds*
4. **wireless client band-select expire suppression** *seconds*
5. **wireless client band-select expire dual-band** *seconds*
6. **wireless client band-select client-rssi** *client_rssi*
7. **end**
8. **wlan** *wlan_profile_name wlan_ID SSID_network_name* **band-select**
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**wireless client band-select cycle-count** *cycle_count* **Example:** `Controller(config)# wireless client band-select cycle-count 3`	Sets the probe cycle count for band select. You can enter a value between 1 and 10 for the *cycle_count* parameter.
Step 3	**wireless client band-select cycle-threshold** *milliseconds* **Example:** `Controller(config)# wireless client band-select cycle-threshold 5000`	Sets the time threshold for a new scanning cycle period. You can enter a value for threshold between 1 and 1000 for the *milliseconds* parameter.
Step 4	**wireless client band-select expire suppression** *seconds* **Example:** `Controller(config)# wireless client band-select expire suppression 100`	Sets the suppression expire to the band select. You can enter a value for suppression between 10 to 200 for the *seconds* parameter.
Step 5	**wireless client band-select expire dual-band** *seconds*	Sets the dual band expire.

	Command or Action	Purpose
	Example: `Controller(config)# wireless client band-select expire dual-band 100`	You can enter a value for dual band between 10 and 300 for the *seconds* parameter.
Step 6	**wireless client band-select client-rssi** *client_rssi* **Example:** `Controller(config)# wireless client band-select client-rssi 40`	Sets the client RSSI threshold. You can enter a value for minimum dBm of a client RSSI to respond to a probe between 20 and 90 for the *client_rssi* parameter.
Step 7	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Step 8	**wlan** *wlan_profile_name wlan_ID SSID_network_name* **band-select**
	Example:
	`Controller(config)# wlan wlan1 25 ssid12`
	`Controller(config-wlan)# band-select`
		Configures band selection on specific WLANs.
		You can enter a value between 1 and 512 for the *wlan_ID* parameter.
		You can enter the up to 32 alphanumeric characters for *SSID_network_name* parameter.

Step 9	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring 802.11 Bands (CLI)

You can configure 802.11 bands and parameters.

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 5ghz shutdown**
3. **ap dot11 24ghz shutdown**
4. **ap dot11 {5ghz | 24ghz} beaconperiod** *time_unit*
5. **ap dot11 {5ghz | 24ghz} fragmentation** *threshold*
6. **ap dot11 {5ghz | 24ghz} dtpc**
7. **wireless client association limit** *number* **interval** *milliseconds*
8. **ap dot11 {5ghz | 24ghz} rate** *rate {disable | mandatory | supported}*
9. **no ap dot11 5ghz shutdown**
10. **no ap dot11 24ghz shutdown**
11. **ap dot11 24ghz dot11g**
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 5ghz shutdown** **Example:** `Controller(config)# ap dot11 5ghz shutdown`	Disables the 802.11a band. **Note** You must disable the 802.11a band before configuring the 802.11a network parameters.
Step 3	**ap dot11 24ghz shutdown** **Example:** `Controller(config)# ap dot11 24ghz shutdown`	Disables the 802.11b band. **Note** You must disable the 802.11b band before configuring the 802.11b network parameters.
Step 4	**ap dot11 {5ghz | 24ghz} beaconperiod** *time_unit* **Example:** `Controller(config)# ap dot11 5ghz beaconperiod 500`	Specifies the rate at which the SSID is broadcast by the access point. The beacon interval is measured in time units (TUs). One TU is 1024 microseconds. You can configure the access point to send a beacon every 20 to 1000 milliseconds.
Step 5	**ap dot11 {5ghz | 24ghz} fragmentation** *threshold* **Example:** `Controller(config)# ap dot11 5ghz fragmentation 300`	Specifies the size at which packets are fragmented. The threshold is a value between 256 and 2346 bytes (inclusive). Specify a low number for areas where communication is poor or where there is a great deal of radio interference.

Step 6	**ap dot11 {5ghz | 24ghz} dtpc**
	
	Example:
	`Controller(config)# ap dot11 5ghz dtpc`
	
	`Controller(config)# no ap dot11 24ghz dtpc`
		Enables access points to advertise their channels and transmit the power levels in beacons, and probe responses.
		The default value is enabled. Client devices using dynamic transmit power control (DTPC) receive the channel and power level information from the access points and adjust their settings automatically. For example, a client device used primarily in Japan could rely on DTPC to adjust its channel and power settings automatically when it travels to Italy and joins a network there.
		Note On access points that run Cisco IOS software, this feature is called world mode.
		The **no** form of the command disables the 802.11a or 802.11b DTPC setting.

Step 7	**wireless client association limit** *number* **interval** *milliseconds* **Example:** `Controller(config)# wireless client association limit 50 interval 1000`	Specifies the maximum allowed clients that can be configured. You can configure a maximum number of association request on a single access point slot at a given interval. The range of association limit that you can configure is from one through 100.

	Command or Action	Purpose
		The association request limit interval is measured between 100 to 10000 milliseconds.
Step 8	**ap dot11 {5ghz | 24ghz} rate** *rate {disable | mandatory | supported}* **Example:** `Controller(config)# ap dot11 5ghz rate 36 mandatory`	Specifies the rate at which data can be transmitted between the controller and the client. • *disabled*—Defines that the clients specify the data rates used for communication. • *mandatory*—Defines that the clients support this data rate in order to associate to an access point on the controller. • *supported*—Any associated clients that support this data rate may communicate with the access point using that rate. However, the clients are not required to be able to use this rate in order to associate. • *rate*—Specifies the rate at which data is transmitted. For the 802.11a and 802.11b bands, the data is transmitted at the rate of 1, 2, 5.5, 6, 9, 11, 12, 18, 24, 36, 48, or 54 Mbps.
Step 9	**no ap dot11 5ghz shutdown** **Example:** `Controller(config)# no ap dot11 5ghz shutdown`	Enables the 802.11a band. **Note** The default value is enabled.
Step 10	**no ap dot11 24ghz shutdown** **Example:** `Controller(config)# no ap dot11 24ghz shutdown`	Enables the 802.11b band. **Note** The default value is enabled.
Step 11	**ap dot11 24ghz dot11g** **Example:** `Controller(config)# ap dot11 24ghz dot11g`	Enables or disables 802.11g network support. The default value is enabled. You can use this command only if the 802.11b band is enabled. If you disable this feature, the 802.11b band is enabled without 802.11g support.
Step 12	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Configuring 802.11n Parameters (CLI)

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 {5ghz | 24ghz} dot11n**
3. **ap dot11 {5ghz | 24ghz} dot11n mcs tx** *rtu*
4. **wlan***wlan_profile_name wlan_ID SSID_network_name* **wmm require**
5. **ap dot11 {5ghz | 24ghz} shutdown**
6. **{ap | no ap} dot11 {5ghz | 24 ghz} dot11n a-mpdu tx priority {all | 0-7}**
7. **no ap dot11 {5ghz | 24ghz} shutdown**
8. **ap dot11 {5ghz | 24ghz} dot11n guard-interval {any | long}**
9. **ap dot11 {5ghz | 24ghz} dot11n rifs rx**
10. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 {5ghz | 24ghz} dot11n** **Example:** `Controller(config)# ap dot11 5ghz dot11n`	Enables 802.11n support on the network. The **no** form of the command disables the 802.11n support on the network.
Step 3	**ap dot11 {5ghz | 24ghz} dot11n mcs tx** *rtu* **Example:** `Controller(config)# ap dot11 5ghz dot11n mcs tx 20`	Specifies the modulation and coding scheme (MCS) rates at which data can be transmitted between the access point and the client. You can set a value from 0 through 23 for the **mcs tx** parameter. The **no** form of the command disables the MCS rates that is configured.

Step 4	**wlan***wlan_profile_name wlan_ID SSID_network_name* **wmm require**
	
	Example:
	`Controller(config)# wlan wlan1 25 ssid12`
	`Controller(config-wlan)# wmm require`
		Enables WMM on the WLAN and uses the 802.11n data rates that you configured.
		The **require** parameter requires client devices to use WMM. Devices that do not support WMM cannot join the WLAN.

Step 5	**ap dot11 {5ghz | 24ghz} shutdown** **Example:** `Controller(config)# ap dot11 5ghz shutdown`	Disables the network.

	Command or Action	Purpose
Step 6	**{ap | no ap} dot11 {5ghz | 24 ghz} dot11n a-mpdu tx priority {all | 0-7}** **Example:** `Controller(config)# ap dot11 5ghz dot11n a-mpdu tx priority all`	Specifies the aggregation method used for 802.11n packets. Aggregation is the process of grouping packet data frames together rather than transmitting them separately. Two aggregation methods are available: Aggregated MAC Protocol Data Unit (A-MPDU) and Aggregated MAC Service Data Unit (A-MSDU). Both A-MPDU and A-MSDU are performed in the software. You can specify the aggregation method for various types of traffic from the access point to the clients. The following table defines the priority levels (0-7) assigned per traffic type. ***Table 5: Traffic Type Priority Levels*** **User Priority** / **Traffic Type** 0 / Best effort 1 / Background 2 / Spare 3 / Excellent effort 4 / Controlled load 5 / Video, less than 100-ms latency and jitter 6 / Voice, less than 100-ms latency and jitter 7 / Network control You can configure each priority level independently, or you can use the all parameter to configure all of the priority levels at once. You can configure priority levels so that the traffic uses either A-MPDU transmission or A-MSDU transmission. • When you use the **ap** command along with the other options, the traffic associated with that priority level uses A-MPDU transmission. • When you use the **no ap** command along with the other options, the traffic associated with that priority level uses A-MSDU transmission. Configure the priority levels to match the aggregation method used by the clients. By default, A-MPDU is enabled for priority level 0,

	Command or Action	Purpose
		4 and 5 and the rest are disabled. By default, A-MPDU is enabled for all priorities except 6 and 7.
Step 7	**no ap dot11 {5ghz | 24ghz} shutdown** **Example:** `Controller(config)# no ap dot11 5ghz shutdown`	Reenables the network.
Step 8	**ap dot11 {5ghz | 24ghz} dot11n guard-interval {any | long}** **Example:** `Controller(config)# ap dot11 5ghz dot11n guard-interval long`	Configures the guard interval for the network.
Step 9	**ap dot11 {5ghz | 24ghz} dot11n rifs rx** **Example:** `Controller(config)# ap dot11 5ghz dot11n rifs rx`	Configures the Reduced Interframe Space (RIFS) for the network.
Step 10	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring 802.11h Parameters (CLI)

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 5ghz shutdown**
3. **{ap | no ap} dot11 5ghz channelswitch mode** *switch_mode*
4. **ap dot11 5ghz power-constraint** *value*
5. **no ap dot11 5ghz shutdown**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 5ghz shutdown** **Example:** `Controller(config)# ap dot11 5ghz shutdown`	Disables the 802.11a network.
Step 3	**{ap | no ap} dot11 5ghz channelswitch mode** *switch_mode* **Example:** `Controller(config)# ap dot11 5ghz channelswitch mode 0`	Enables or disables the access point to announce when it is switching to a new channel. You can enter a 0 or 1 for the **channelswitch** parameter to specify whether transmissions are restricted until the actual channel switch (0) or are not restricted (1). The default value is disabled.
Step 4	**ap dot11 5ghz power-constraint** *value* **Example:** `Controller(config)# ap dot11 5ghz power-constraint 200`	Configures the 802.11h power constraint value in a range from zero through 255. The default value for the value parameter is 3 dB.
Step 5	**no ap dot11 5ghz shutdown** **Example:** `Controller(config)# no ap dot11 5ghz shutdown`	Reenables the 802.11a network.
Step 6	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Monitoring Configuration Settings for Band Selection, 802.11 Bands, and Parameters

Monitoring Configuration Settings Using Band Selection and 802.11 Bands Commands

This section describes the new commands for band selection and 802.11 bands.

The following commands can be used to monitor band selection, and 802.11 bands and parameters the controller.

Table 6: Monitoring Configuration Settings Using Band Selection and 802.11 Bands Commands

Command	Purpose
show ap dot11 5ghz network	Displays 802.11a bands network parameters, 802.11a operational rates, 802.11n MCS settings, and 802.11n status information.
show ap dot11 24ghz network	Displays 802.11b bands network parameters, 802.11b/g operational rates, 802.11n MCS settings, and 802.11n status information.
show wireless dot11h	Displays 802.11h configuration parameters.
show wireless band-select	Displays band select configuration settings.

Example: Viewing the Configuration Settings for 5-GHz Band

```
Controller# show ap dot11 5ghz network
802.11a Network : Enabled
11nSupport : Enabled
  802.11a Low Band : Enabled
  802.11a Mid Band : Enabled
  802.11a High Band : Enabled

802.11a Operational Rates
  802.11a 6M : Mandatory
  802.11a 9M : Supported
  802.11a 12M : Mandatory
  802.11a 18M : Supported
  802.11a 24M : Mandatory
  802.11a 36M : Supported
  802.11a 48M : Supported
  802.11a 54M : Supported
802.11n MCS Settings:
  MCS 0 : Supported
  MCS 1 : Supported
  MCS 2 : Supported
  MCS 3 : Supported
  MCS 4 : Supported
  MCS 5 : Supported
  MCS 6 : Supported
  MCS 7 : Supported
  MCS 8 : Supported
  MCS 9 : Supported
  MCS 10 : Supported
  MCS 11 : Supported
  MCS 12 : Supported
  MCS 13 : Supported
  MCS 14 : Supported
  MCS 15 : Supported
  MCS 16 : Supported
  MCS 17 : Supported
  MCS 18 : Supported
  MCS 19 : Supported
  MCS 20 : Supported
  MCS 21 : Supported
  MCS 22 : Supported
  MCS 23 : Supported
```

```
802.11n Status:
  A-MPDU Tx:
    Priority 0 : Enabled
    Priority 1 : Disabled
    Priority 2 : Disabled
    Priority 3 : Disabled
    Priority 4 : Enabled
    Priority 5 : Enabled
    Priority 6 : Disabled
    Priority 7 : Disabled
  A-MSDU Tx:
    Priority 0 : Enabled
    Priority 1 : Enabled
    Priority 2 : Enabled
    Priority 3 : Enabled
    Priority 4 : Enabled
    Priority 5 : Enabled
    Priority 6 : Disabled
    Priority 7 : Disabled
Guard Interval : Any
  Rifs Rx : Enabled
Beacon Interval : 100
CF Pollable mandatory : Disabled
CF Poll Request Mandatory : Disabled
CFP Period : 4
CFP Maximum Duration : 60
Default Channel : 36
Default Tx Power Level : 1
DTPC Status : Enabled
Fragmentation Threshold : 2346
Pico-Cell Status : Disabled
Pico-Cell-V2 Status : Disabled
TI Threshold : 0
Legacy Tx Beamforming setting : Disabled
Traffic Stream Metrics Status : Disabled
Expedited BW Request Status : Disabled
EDCA profile type check : default-wmm
Call Admision Control (CAC) configuration
Voice AC
  Voice AC - Admission control (ACM) : Disabled
  Voice Stream-Size : 84000
  Voice Max-Streams : 2
  Voice Max RF Bandwidth : 75
  Voice Reserved Roaming Bandwidth : 6
  Voice Load-Based CAC mode : Enabled
  Voice tspec inactivity timeout : Enabled
CAC SIP-Voice configuration
  SIP based CAC : Disabled
  SIP Codec Type : CODEC_TYPE_G711
  SIP call bandwidth : 64
  SIP call bandwith sample-size : 20
Video AC
  Video AC - Admission control (ACM) : Disabled
  Video max RF bandwidth : Infinite
  Video reserved roaming bandwidth : 0
```

Example: Viewing the Configuration Settings for 24-GHz Band

```
Controller# show ap dot11 24ghz network
802.11b Network : Enabled
11gSupport : Enabled
11nSupport : Enabled

  802.11b/g Operational Rates
  802.11b 1M : Mandatory
  802.11b 2M : Mandatory
  802.11b 5.5M : Mandatory
  802.11g 6M : Supported
  802.11g 9M : Supported
  802.11b 11M : Mandatory
```

```
    802.11g 12M : Supported
    802.11g 18M : Supported
    802.11g 24M : Supported
    802.11g 36M : Supported
    802.11g 48M : Supported
    802.11g 54M : Supported
  802.11n MCS Settings:
    MCS 0 : Supported
    MCS 1 : Supported
    MCS 2 : Supported
    MCS 3 : Supported
    MCS 4 : Supported
    MCS 5 : Supported
    MCS 6 : Supported
    MCS 7 : Supported
    MCS 8 : Supported
    MCS 9 : Supported
    MCS 10 : Supported
    MCS 11 : Supported
    MCS 12 : Supported
    MCS 13 : Supported
    MCS 14 : Supported
    MCS 15 : Supported
    MCS 16 : Supported
    MCS 17 : Supported
    MCS 18 : Supported
    MCS 19 : Supported
    MCS 20 : Supported
    MCS 21 : Supported
    MCS 22 : Supported
    MCS 23 : Supported
  802.11n Status:
    A-MPDU Tx:
      Priority 0 : Enabled
      Priority 1 : Disabled
      Priority 2 : Disabled
      Priority 3 : Disabled
      Priority 4 : Enabled
      Priority 5 : Enabled
      Priority 6 : Disabled
      Priority 7 : Disabled
    A-MSDU Tx:
      Priority 0 : Enabled
      Priority 1 : Enabled
      Priority 2 : Enabled
      Priority 3 : Enabled
      Priority 4 : Enabled
      Priority 5 : Enabled
    Priority 6 : Disabled
      Priority 7 : Disabled
    Guard Interval : Any
    Rifs Rx : Enabled
  Beacon Interval : 100
  CF Pollable Mandatory : Disabled
  CF Poll Request Mandatory : Disabled
  CFP Period : 4
  CFP Maximum Duration : 60
  Default Channel : 11
  Default Tx Power Level : 1
  DTPC Status : true
  Call Admission Limit : 105
  G711 CU Quantum : 15
  ED Threshold : -50
  Fragmentation Threshold : 2346
  PBCC Mandatory : Disabled
  Pico-Cell Status : Disabled
  Pico-Cell-V2 Status : Disabled
  RTS Threshold : 2347
  Short Preamble Mandatory : Enabled
  Short Retry Limit : 7
  Legacy Tx Beamforming setting : Disabled
  Traffic Stream Metrics Status : Disabled
  Expedited BW Request Status : Disabled
```

```
EDCA profile type : default-wmm
Call Admision Control (CAC) configuration
Voice AC
  Voice AC - Admission control (ACM) : Disabled
  Voice Stream-Size : 84000
  Voice Max-Streams : 2
  Voice Max RF Bandwidth : 75
  Voice Reserved Roaming Bandwidth : 6
  Voice Load-Based CAC mode : Enabled
  Voice tspec inactivity timeout : Enabled
CAC SIP-Voice configuration
  SIP based CAC : Disabled
  SIP Codec Type : CODEC_TYPE_G711
  SIP call bandwidth : 64
  SIP call bandwith sample-size : 20
Video AC
  Video AC - Admission control (ACM) : Disabled
  Video max RF bandwidth : Infinite
  Video reserved roaming bandwidth : 0
```

Example: Viewing the status of 802.11h Parameters

```
Controller# show wireless dot11h
Power Constraint: 0
Channel Switch: 0
Channel Switch Mode: 0
```

Example: Verifying the Band Selection Settings

```
Controller# show wireless band-select
Band Select Probe Response    : per WLAN enabling
Cycle Count                   : 2
Cycle Threshold (millisec)    : 200
Age Out Suppression (sec)     : 20
Age Out Dual Band (sec)       : 60
Client RSSI (dBm)             : 80
```

Configuration Examples for Band Selection, 802.11 Bands, and Parameters

Examples: Band Selection Configuration

This example shows how to set the probe cycle count and time threshold for a new scanning cycle period for band select:

```
Controller# configure terminal
Controller(config)# wireless client band-select cycle-count 3
Controller(config)# wireless client band-select cycle-threshold 5000
Controller(config)# end
```

This example shows how to set the suppression expire to the band select:

```
Controller# configure terminal
Controller(config)# wireless client band-select expire suppression 100
Controller(config)# end
```

This example shows how to set the dual band expire for the band select:

```
Controller# configure terminal
```

```
Controller(config)# wireless client band-select expire dual-band 100
Controller(config)# end
```

This example shows how to set the client RSSI threshold for the band select:

```
Controller# configure terminal
Controller(config)# wireless client band-select client-rssi 40
Controller(config)# end
```

This example shows how to configure band selection on specific WLANs:

```
Controller# configure terminal
Controller(config)# wlan wlan1 25 ssid12
Controller(config-wlan)# band-select
Controller(config)# end
```

Examples: 802.11 Bands Configuration

This example shows how to configure 802.11 bands using beacon interval, fragmentation, and dynamic transmit power control:

```
Controller# configure terminal
Controller(config)# ap dot11 5ghz shutdown
Controller(config)# ap dot11 24ghz shutdown
Controller(config)# ap dot11 5ghz beaconperiod 500
Controller(config)# ap dot11 5ghz fragmentation 300
Controller(config)# ap dot11 5ghz dtpc
Controller(config)# wireless client association limit 50 interval 1000
Controller(config)# ap dot11 5ghz rate 36 mandatory
Controller(config)# no ap dot11 5ghz shutdown
Controller(config)# no ap dot11 24ghz shutdown
Controller(config)# ap dot11 24ghz dot11g
Controller(config)#end
```

Examples: 802.11n Configuration

This example shows how to configure 802.11n parameters for 5-GHz band using aggregation method:

```
Controller# configure terminal
Controller(config)# ap dot11 5ghz dot11n
Controller(config)# ap dot11 5ghz dot11n mcs tx 20
Controller(config)# wlan wlan1 25 ssid12
Controller(config-wlan)# wmm require\
Controller(config-wlan)# exit
Controller(config)# ap dot11 5ghz shutdown
Controller(config)# ap dot11 5ghz dot11n a-mpdu tx priority all
Controller(config)# no ap dot11 5ghz shutdown
Controller(config)#exit
```

This example shows how to configure the guard interval for 5-GHz band:

```
Controller# configure terminal
Controller(config)# ap dot11 5ghz dot11n
Controller(config)# ap dot11 5ghz dot11n mcs tx 20
Controller(config)# wlan wlan1 25 ssid12
Controller(config-wlan)# wmm require\
Controller(config-wlan)# exit
Controller(config)# no ap dot11 5ghz shutdown
Controller(config)# ap dot11 5ghz dot11n guard-interval long
Controller(config)#end
```

This example shows how to configure the RIFS for 5-GHz band:

```
Controller# configure terminal
Controller(config)# ap dot11 5ghz dot11n
Controller(config)# ap dot11 5ghz dot11n mcs tx 20
Controller(config)# wlan wlan1 25 ssid12
Controller(config-wlan)# wmm require\
Controller(config-wlan)# exit
Controller(config)# ap dot11 5ghz shutdown
Controller(config)# ap dot11 5ghz dot11n rifs rx
Controller(config)#end
```

Examples: 802.11h Configuration

This example shows how to configure the access point to announce when it is switching to a new channel using restriction transmission:

```
Controller# configure terminal
Controller(config)# ap dot11 5ghz shutdown
Controller(config)# ap dot11 5ghz channelswitch mode 0
Controller(config)# no ap dot11 5ghz shutdown
Controller(config)#end
```

This example shows how to configure the 802.11h power constraint for 5-GHz band:

```
Controller# configure terminal
Controller(config)# ap dot11 5ghz shutdown
Controller(config)# ap dot11 5ghz power-constraint 200
Controller(config)# no ap dot11 5ghz shutdown
Controller(config)#end
```

Additional References for 802.11 Parameters and Band Selection

Related Documents

Related Topic	Document Title
System management commands	*System Management Command Reference, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

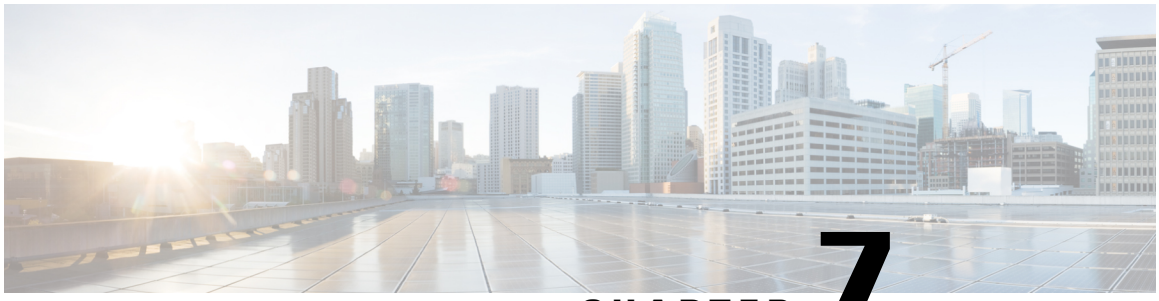
MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

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 802.11 parameters and Band Selection Configuration

Release	Modification
Cisco IOS XE Release 3.2SE	This feature was introduced.



CHAPTER 7

Configuring Aggressive Load Balancing

This module contains the following topics:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Aggressive Load Balancing

- You can configure aggressive load balancing only from the command-line interface.
- Aggressive load balancing is disabled by default, you have to enable it manually.
- You can enable load balancing either separately or together with the band select configurations.
- When the band select is enabled on the dual-band clients, the load balancing parameter selects only the lowest load radio from 5-GHz radios. For the 2.4-GHz clients, there is no probe information of the client on 5 GHz and therefore the load balancing algorithm can only be selected between radio on 2.4 GHz.

- You can operate load balancing of clients between access points on the same controller but not for the clients between access points on the different controller.
- The load balancing uses an existing association denial mechanism based on the number of client on the radio and the band select is implemented by the distributed probe response suppression on the access point only.

Information for Configuring Aggressive Load Balancing Parameters

Aggressive Load Balancing

Enabling aggressive load balancing on the controller allows lightweight access points to load balance wireless clients across access points. You can enable aggressive load balancing using the controller.

When a wireless client attempts to associate to a lightweight access point, association response packets are sent to the client with an 802.11 response packet including status code 17. The code 17 indicates that the AP is busy. The AP responds with an association response bearing 'success' if the AP threshold is not met, and with code 17 (AP busy) if the AP utilization threshold is reached or exceeded and another less busy AP heard the client request.

For example, if the number of clients on AP1 is more than the number of clients on AP2 plus the load-balancing window, then AP1 is considered to be busier than AP2. When a client attempts to associate to AP1, it receives an 802.11 response packet with status code 17, indicating that the access point is busy, and the client attempts to associate to a different access point.

You can configure the controller to deny client associations up to 10 times (if a client attempted to associate 11 times, it would be allowed to associate on the 11th try). You can also enable or disable load balancing on a particular WLAN, which is useful if you want to disable load balancing for a select group of clients (such as time-sensitive voice clients).

Clients are load balanced between access points on the same controller. Load balancing does not occur between access points on different controllers.

The maximum number of client associations that the access points can support is dependent upon the following factors:

- The maximum number of client associations differs for lightweight and autonomous Cisco IOS access points.
- There may be a limit per radio and an overall limit per AP.
- AP hardware (the 16-MB APs have a lower limit than the 32-MB and higher APs)

The Client Association Limits for Lightweight Access Points are as follows:

- For 16-MB APs, the limit is 128 clients per AP. This limit is applicable to 1100 and 1200 series APs.
- For 32-MB and higher APs, there is no per-AP limit.

The maximum Client Association Limits per-radio for all the Cisco IOS APs is 200 associations.

Note With 32-MB and higher lightweight Cisco IOS APs, with two radios, up to 200 + 200 = 400 associations are supported.

The maximum Client Association Limits per Autonomous Cisco IOS access point is around 80 to 127 clients per AP. This number varies depending on the following factors:

- AP model (whether it is 16 MB or 32 MB or higher)
- Cisco IOS software release
- Hardware configuration (two radios use more memory than one)
- Enabled features (WDS functionality in particular)

The per-radio limit is about 200 associations. One association will likely hit the per-AP limit first. Unlike Cisco Unified Wireless Network, autonomous Cisco IOS supports per-SSID/per-AP association limits. This limit is configured using the max-associations CLI, under dot11 SSID. The maximum number is 255 associations (which is also the default number).

How to Configure Aggressive Load Balancing

Configuring Aggressive Load Balancing

SUMMARY STEPS

1. **configure terminal**
2. **wireless load-balancing window** *client-count*
3. **wireless load-balancing denial** *denial-count*
4. **end**
5. **wlan** *wlan_profile_name wlan_ID SSID_network_name* **load-balance**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**wireless load-balancing window** *client-count* **Example:** `Controller(config)# wireless load-balancing window 1`	Sets the client window for aggressive load balancing. You can enter a value between 0 and 20 for the *client_count* parameter.
Step 3	**wireless load-balancing denial** *denial-count* **Example:** `Controller(config)# wireless load-balancing denial-count 1`	Sets the denial count for load balancing. You can enter a value between 0 and 10 for the *denial_count* parameter.

	Command or Action	Purpose
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Step 5	**wlan** *wlan_profile_name wlan_ID SSID_network_name* **load-balance**
	
	Example:
	`Controller(config)# wlan wlan1 25 ssid12`
	`Controller(config-wlan)# load-balance`
		Enables or disables aggressive load balancing on specific WLANs
		
		You can enter a value between 1 and 512 for the *wlan_ID* parameter.
		
		You can enter the up to 32 alphanumeric characters for *SSID_network_name* parameter.

Step 6	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Monitoring Aggressive Load Balancing

This section describes the new command for aggressive load balancing.

The following command can be used to monitor aggressive load balancing on the controller.

Table 7: Monitoring Aggressive Load Balancing Command

Command	Purpose
show wireless load-balancing	Displays the status of the load-balancing feature.

Examples: Aggressive Load Balancing Configuration

This example shows how to configure the load balancing denial count:

```
Controller# configure terminal
Controller(config)# wireless load-balancing denial-count 1
Controller(config)# end
Controller# show wireless load-balancing
```

This example shows how to configure the client window for aggressive load balancing:

```
Controller# configure terminal
Controller(config)# wireless load-balancing window 1
Controller(config)# end
Controller# show wireless load-balancing
```

This example shows how to configure load balancing on specific WLAN:

```
Controller# configure terminal
Controller(config)# wlan wlan1 25 ssid12
Controller(config-wlan)# load-balance
Controller(config)# end
Controller# show wireless load-balancing
```

Additional References for Aggressive Load Balancing

Related Documents

Related Topic	Document Title
System management commands	*System Management Command Reference Guide, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

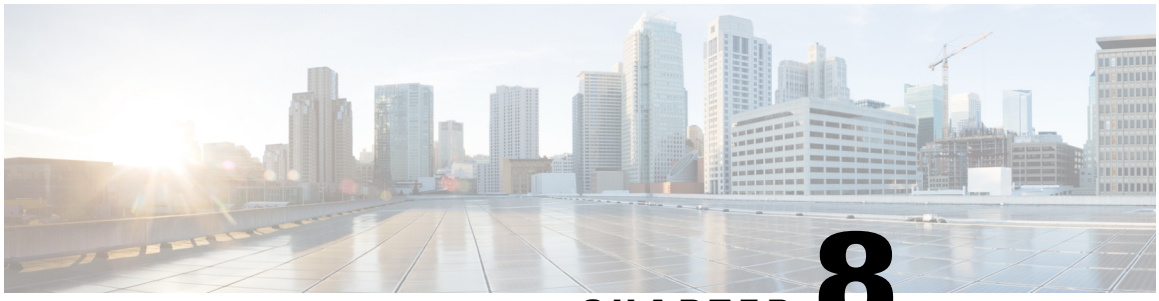
MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

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 Aggressive Load Balancing Configuration

Release	Modification
Cisco IOS XE Release 3.2SE	This feature was introduced.



CHAPTER 8

Configuring Client Roaming

This module contains the following topics:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Client Roaming

- There should be one active mobility controller to manage client roaming.
- The WLAN SSID on the mobility agents across which roaming is desired should be the same.

Restrictions for Configuring Client Roaming

The following are the restrictions that you should be aware while configuring client roaming:

- Cisco Compatible Extensions (CCX) support is enabled automatically for every WLAN on the controller and cannot be disabled. The controller stores the CCX version of the client in its client database and uses it to generate and respond to CCX frames appropriately. Clients must support CCXv4 or v5 (or CCXv2 for access point assisted roaming) to utilize these roaming enhancements.
- Client roaming between 600 Series Access points is not supported.

Information About Client Roaming

The controllers deliver high-end wireless services to the clients roaming across wireless network. Now, the wireless services are integrated with the switches, thus delivering a value-added Cisco unified new mobility architecture. This unified architecture enables client-roaming services to both wireless and wired clients with seamless, fast- roaming services.

The new mobility architecture supports fast client roaming services using logical categorization of network into Mobility Domains (MDs), Mobility Groups (MGs), Mobility Subdomains (MSDs), and Switch Peer Groups (SPGs) using systems such as Mobility Oracle (MO), Mobility Controller (MC), and Mobility Agent (MA).

- A **Mobility Domain** is the entire domain across which client roaming is supported. It is a collection of mobility groups. For example, a campus network can be considered as a mobility domain.
- A **Mobility Group** is a collection of mobility subdomains across which fast roaming is supported. The mobility group can be one or more buildings within a campus across which frequent roaming is supported.
- A **Mobility Subdomain** is an autonomous portion of the mobility domain network. Each mobility subdomain contains one mobility controller (MC) and a collection of SPGs. A subdomain is equivalent to an 802.11r key domain.
- A **Switch Peer Group** is a collection of mobility agents.
- The **Mobility Oracle** acts as the point of contact for mobility events that occur across mobility subdomains. The mobility oracle also maintains a local database of each client in the entire mobility domain, their home and current subdomain. There is only one MO for an entire mobility domain. The Cisco WLC 5700 Series Controllers or Cisco Unified Wireless Networking Solution controller can act as MO.
- The **Mobility Controller** provides mobility management services for inter-SPG roaming events. The MC sends the configuration like SPG name and SPG peer member list to all the mobility agents under its subdomain. The Cisco WLC 5700 Series Controllers, Cisco Catalyst 3850 Switch, or Cisco Unified Wireless Networking Solution controller can act as MC. The MC has MC functionality and MA functionality that is running internally into it.
- The **Mobility Agent** is the component that maintains client mobility state machine for a mobile client. All APs are connected to the mobility agent.

The New mobility architecture supports seamless roaming in the following scenarios:

- Intra-switch roaming—The client roaming between APs managed by same mobility agent.

- Intra-SPG roaming—The client roaming between mobility agents in the same SPG.
- Inter-SPG, Intra-subdomain roaming—The client roaming between mobility agents in different SPGs within the same subdomain.
- Inter-subdomain roaming—The client roaming between mobility agents across a subdomain.

Fast Roaming

New mobility architecture supports fast roaming when clients roam within a mobility group by eliminating the need for full authentication. Security polices should be same across the switches for fast roaming.

Local, anchor, foreign MAs and MCs

When a client joins an MA initially and its point of attachment has not changed, that MA is referred as local or associated MA. The MC to which this MA is associated is referred as local or associated MC.

When a client roams between two MAs, the MA to which the client was previously associated is the anchor MA (point of attachment) and the MA to which the client is currently associated is the foreign or associated MA (point of presence). The MCs to which these MAs are associated are referred as anchor, foreign, or associated MCs, respectively.

Inter-Subnet Roaming

Multiple-controller deployments support client roaming across access points managed by controllers in the same mobility group on different subnets. This roaming is transparent to the client because the session is sustained and a tunnel between the controllers allows the client to continue using the same DHCP-assigned or client-assigned IP address as long as the session remains active. The tunnel is torn down, and the client must reauthenticate when the client sends a DHCP Discover with a 0.0.0.0 client IP address or a 169.254.*.* client auto-IP address or when the operator-set user timeout is exceeded.

Voice-over-IP Telephone Roaming

802.11 voice-over-IP (VoIP) telephones actively seek out associations with the strongest RF signal to ensure the best quality of service (QoS) and the maximum throughput. The minimum VoIP telephone requirement of 20-millisecond or shorter latency time for the roaming handover is easily met by the Cisco Unified Wireless Network (Cisco UWN) solution, which has an average handover latency of 5 or fewer milliseconds when open authentication is used. This short latency period is controlled by controllers rather than allowing independent access points to negotiate roaming handovers.

The Cisco UWN solution supports 802.11 VoIP telephone roaming across lightweight access points managed by controllers on different subnets, as long as the controllers are in the same mobility group. This roaming is transparent to the VoIP telephone because the session is sustained and a tunnel between controllers allows the VoIP telephone to continue using the same DHCP-assigned IP address as long as the session remains active. The tunnel is torn down, and the VoIP client must reauthenticate when the VoIP telephone sends a DHCP Discover with a 0.0.0.0 VoIP telephone IP address or a 169.254.*.* VoIP telephone auto-IP address or when the operator-set user timeout is exceeded.

CCX Layer 2 Client Roaming

The controller supports five CCX Layer 2 client roaming enhancements:

- Access point assisted roaming—This feature helps clients save scanning time. When a CCXv2 client associates to an access point, it sends an information packet to the new access point listing the

characteristics of its previous access point. Roaming time decreases when the client recognizes and uses an access point list built by compiling all previous access points to which each client was associated and sent (unicast) to the client immediately after association. The access point list contains the channels, BSSIDs of neighbor access points that support the client's current SSID(s), and time elapsed since disassociation.

- Enhanced neighbor list—This feature focuses on improving a CCXv4 client's roam experience and network edge performance, especially when servicing voice applications. The access point provides its associated client information about its neighbors using a neighbor-list update unicast message.
- Enhanced neighbor list request (E2E)—The End-2-End specification is a Cisco and Intel joint program that defines new protocols and interfaces to improve the overall voice and roaming experience. It applies only to Intel clients in a CCX environment. Specifically, it enables Intel clients to request a neighbor list at will. When this occurs, the access point forwards the request to the controller. The controller receives the request and replies with the current CCX roaming sublist of neighbors for the access point to which the client is associated.

Note To see whether a particular client supports E2E, choose **Wireless** > **Clients** on the controller GUI, click the **Detail** link for the desired client, and look at the E2E Version text box in the Client Properties area.

- Roam reason report—This feature enables CCXv4 clients to report the reason why they roamed to a new access point. It also allows network administrators to build and monitor a roam history.
- Directed roam request—This feature enables the controller to send directed roam requests to the client in situations when the controller can better service the client on an access point different from the one to which it is associated. In this case, the controller sends the client a list of the best access points that it can join. The client can either honor or ignore the directed roam request. Non-CCX clients and clients running CCXv3 or below must not take any action. No configuration is required for this feature.

How to Configure Layer 2 or Layer 3 Roaming

Configuring Layer 2 or Layer 3 Roaming

Before You Begin

To configure the mobility agent for Layer 2 or Layer 3 roaming, the following requisites should be considered:

- SSID and security polices should be same across MAs for Layer 2 and Layer 3 roaming.
- Client VLAN ID should be same for Layer 2 roaming and different for Layer 3 roaming.
- Bridge domain ID and client VLAN IDs should be same for Layer 2 roaming. Either one or both of the bridge domain ID and client VLAN ID should be different for Layer 3 roaming.

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan_profile_name wlan_ID SSID_network_name*
3. **no mobility anchor sticky**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *wlan_profile_name wlan_ID SSID_network_name* **Example:** `Controller(config)#wlan wlan1`	Enters into WLAN configuration mode.
Step 3	**no mobility anchor sticky** **Example:** `Controller(config-wlan)#no mobility anchor sticky`	(Optional) Disables Layer 2 anchoring.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring CCX Client Roaming Parameters (CLI)

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 {5ghz | 24ghz} l2roam rf-params** {**default | custom** *min-rssi roam-hyst scan-thresh trans-time*}
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 {5ghz | 24ghz} l2roam rf-params** {**default | custom** *min-rssi roam-hyst scan-thresh trans-time*} **Example:** `Controller#ap dot11 5ghz l2roam rf-params custom -80`	Configures CCX Layer 2 client roaming parameters. To choose the default RF parameters, enter the **default** option. To fine-tune the RF parameters that affect client roaming, enter the **custom** option and then enter any one of the following options: • Minimum RSSI—Indicates minimum Received Signal Strength Indicator (RSSI) required for the client to associate to an access point. If the client's average received signal power dips below this threshold, reliable communication is usually impossible. Therefore, clients must already have found and roamed to another access point with a stronger signal before the minimum RSSI value is reached. You can configure the minimum RSSI range from –80 through –90 dBm and the default is –85 dBm. • Hysteresis—Indicates how much greater the signal strength of a neighboring access point must be in order for the client to roam to it. This parameter is intended to reduce the amount of roaming between access points if the client is physically located on or near the border between two access points. You can configure the hysteresis range from 3 through 20 dB and the default is 3 dB. • Scan Threshold—Indicates a minimum RSSI that is allowed before the client should roam to a better access point. When the RSSI drops below the specified value, the client must be able to roam to a better access point within the specified transition time. This parameter also provides a power-save method to minimize the time that the client spends in active or passive scanning. For example, the client can scan slowly when the RSSI is above the threshold and scan more rapidly when the RSSI is below the threshold. You can configure the RSSI range from –70 through –77 dBm and the default value is –72 dBm. • Transition Time—Indicates the maximum time allowed for the client to detect a suitable neighboring access point to roam to and to complete the roam, whenever the RSSI from the client's associated access point is below the scan threshold. The Scan Threshold and Transition Time parameters guarantee a minimum level of client roaming performance. Together with the highest expected client speed and roaming hysteresis, these parameters make it possible to design a wireless LAN network that supports roaming simply by ensuring a certain minimum overlap distance between access points.

	Command or Action	Purpose
		You can configure the time period in the range from 1 through 10 seconds and the default time is 5 seconds.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring Mobility Oracle

SUMMARY STEPS

1. **configure terminal**
2. **wireless mobility oracle**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**wireless mobility oracle** **Example:** `Controller(config)# wireless mobility oracle`	Enables mobility oracle on the controller.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring Mobility Controller

SUMMARY STEPS

1. **configure terminal**
2. **wireless mobility controller**
3. **wireless mobility controller peer-group** *switch-peer-group-name*
4. **wireless mobility controller peer-group** *switch-peer-group-name* **member ip** *ip-address* {**public-ip** *public-ip-address*}
5. **wireless mobility controller peer-group** *switch-peer-group-name* **multicast**
6. **wireless mobility controller peer-group** *switch-peer-group-name* **multicast ip** *peer-group-multicast-ip-addr*
7. **wireless mobility controller peer-group***switch-peer-group-name* **bridge-domain-id** *id*
8. **wireless mobility group member ip** *ip-address* [**public-ip** *public-ip-address*] [**group** *group-name*]
9. **wireless mobility dscp** *value*
10. **wireless mobility group keepalive** {*count* | *interval*}
11. **wireless mobility group name** *name*
12. **wireless mobility oracle ip***mo-ip-address*
13. **wireless management interface** *interface-name*
14. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**wireless mobility controller** **Example:** `Controller(config)# wireless mobility controller`	Enables wireless mobility controller.
Step 3	**wireless mobility controller peer-group** *switch-peer-group-name* **Example:** `Controller(config)# wireless mobility controller peer-group SPG1`	Configures a switch peer group name. You can enter up to 31 case-sensitive ASCII printable characters for the group name. Spaces are not allowed in mobility group. **Note** The **No** form of the command deletes the switch peer group.
Step 4	**wireless mobility controller peer-group** *switch-peer-group-name* **member ip** *ip-address* {**public-ip** *public-ip-address*}	Adds a mobility group member to a switch peer group. **Note** The **No** form of the command deletes the member from the switch peer group.

	Command or Action	Purpose
	Example: `Controller(config)# wireless mobility controller peer-group SPG1 member ip 10.0.0.1`	
Step 5	**wireless mobility controller peer-group** *switch-peer-group-name* **multicast** **Example:** `Controller(config)# wireless mobility controller peer-group SPG1 multicast`	Configures the multicast mode within a switch peer group.
Step 6	**wireless mobility controller peer-group** *switch-peer-group-name* **multicast ip** *peer-group-multicast-ip-addr* **Example:** `Controller(config)# wireless mobility controller peer-group SPG1 multicast ip 10.0.0.4`	Configures the multicast IP address for a switch peer group. **Note** The **No** form of the command deletes the multicast IP for the switch peer group.
Step 7	**wireless mobility controller peer-group***switch-peer-group-name* **bridge-domain-id** *id* **Example:** `Controller(config)# wireless mobility controller peer-group SPG bridge-domain-id 10.0.0.5`	Configures the bridge domain ID for a switch peer group. The default is zero. **Note** The **No** form of command sets the bridge domain ID to the default value.
Step 8	**wireless mobility group member ip** *ip-address* [**public-ip** *public-ip-address*] [**group** *group-name*] **Example:** `Controller(config)# wireless mobility group member ip 10.0.0.1`	Adds a mobility group member. **Note** The **No** form of the command removes the member from the group. The default group name is the group name of MC.
Step 9	**wireless mobility dscp** *value* **Example:** `Controller(config)# wireless mobility dscp 46`	Sets the DSCP value for mobility control packet. You can configure the DSCP value in a range from 0 through 63. The default value is 46.
Step 10	**wireless mobility group keepalive** {*count* | *interval*} **Example:** `Controller(config)# wireless mobility group keepalive count`	Configures the wireless mobility group keepalive count which is the number of keepalive retries before a member status is termed DOWN and keepalive interval which is interval between two keepalives.
Step 11	**wireless mobility group name** *name* **Example:** `Controller(config)# wireless mobility group name group1`	Specifies the case sensitive wireless mobility group name which can be ASCII printable string up to 31 characters.
Step 12	**wireless mobility oracle ip***mo-ip-address*	Configures the mobility oracle IP address.

	Command or Action	Purpose
	Example: `Controller(config)# wireless mobility oracle ip 10.0.0.5`	
Step 13	**wireless management interface** *interface-name* **Example:** `Controller(config)# wireless management interface Vlan21`	Configures the wireless management interface.
Step 14	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring Mobility Agent

SUMMARY STEPS

1. **configure terminal**
2. **wireless mobility controller ip** *ip-address*
3. **wireless mobility load-balance**
4. **wireless mobility load-balance threshold** *threshold -value*
5. **wireless management interface** *interface-name*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**wireless mobility controller ip** *ip-address* **Example:** `Controller(config)# wireless mobility controller ip 10.10.10.20`	Sets the IP address of the mobility controller.
Step 3	**wireless mobility load-balance** **Example:** `Controller(config)# wireless mobility load-balance`	Configures wireless mobility load balancing.

	Command or Action	Purpose
Step 4	**wireless mobility load-balance threshold** *threshold -value* **Example:** `Controller(config)# wireless mobility load-balance threshold 100`	Configures the number of clients that can be local or anchored on the MA. You can configure the threshold value in a range from 100 to 2000. The default value is 1000.
Step 5	**wireless management interface** *interface-name* **Example:** `Controller(config)# wireless management interface Vlan21`	Configures wireless management interface for the mobility agent.
Step 6	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Monitoring Client Roaming Parameters

This section describes the new commands for the client parameters.

The following commands can be used to monitor the client roaming parameters on the controller.

Table 8: Monitoring Client Roaming Parameters Commands

Command	Purpose
show ap dot11 {5ghz | 24ghz} l2roam rf-param	Displays the current RF parameters configured for client roaming for the 802.11a or 802.11b/g network.
show ap dot11 {5ghz | 24ghz} l2roam statistics	Displays the CCX Layer 2 client roaming statistics for the 802.11a or 802.11b/g network.
show ap dot11 {5ghz | 24ghz} l2roam mac-address *mac-address* **statistics**	Displays the CCX Layer 2 client roaming statistics for a particular access point.

Monitoring Mobility Configurations

This section describes the new commands for monitoring mobility configurations.

The following command can be used to monitor mobility configurations on the Mobility Oracle, Mobility Controller, and Mobility Agent.

Table 9: Monitoring Mobility Configuration Commands on the Mobility Controller and Mobility Agent

Command	Purpose
show wireless mobility summary	Displays the summary information for the Mobility Controller and Mobility Agent.
show wireless mobility statistics	Displays mobility statistics.
show wireless mobility dtls connections	Displays established DTLS connections.

Table 10: Monitoring Mobility Configuration Commands on the Mobility Oracle

Command	Purpose
show wireless mobility oracle summary	Displays the status of the Mobility Controllers known to the Mobility Oracle.
show wireless mobility oracle client summary	Displays the information of a list of clients in the Mobility Oracle database.
show wireless mobility oracle client detail *client -mac-address*	Displays the detailed information of a particular client in the Mobility Oracle database.
show wireless mobility oracle *mc-ip*	Displays the information of a list of clients in the Mobility Oracle database that are anchored or associated to a specified Mobility Controller.

Table 11: Monitoring Mobility Configuration Commands on the Mobility Controller

Command	Purpose
show wireless mobility controller client summary	Displays a list of clients in the subdomain.
show wireless mobility controller client *mac-address* **detail**	Displays detailed information for a client in a sub-domain.
show wireless mobility agent *ma-ip* **client summary**	Displays a list of clients anchored or associated to a specified Mobility Agent.
show wireless mobility ap-list	Displays the list of Cisco APs known to the mobility group.

Table 12: Monitoring Mobility Configuration Commands on the Mobility Agent

Command	Purpose
show wireless mobility load-balance summary	Displays the summary of mobility load-balance properties.

Additional References for Configuring Client Roaming

Related Documents

Related Topic	Document Title
Mobility configuration	*Mobility Configuration Guide, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*
Mobility-related commands	*Mobility Command Reference Guide, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

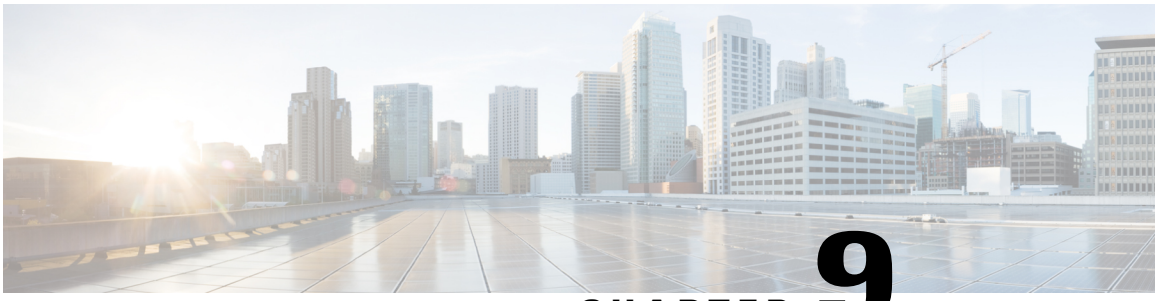
MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

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 Client Roaming Configuration

Release	Modification
Cisco IOS XE Release 3.2SE	This feature was introduced.



CHAPTER 9

Configuring Voice and Video Parameters

This module contains the following topics:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Voice and Video Parameters

You can check on the following points before configuring voice and video parameters:

- Ensure that the controller has access points connected to it.
- Configure SSID.

Restrictions for Voice and Video Parameters

The following are the restrictions that you should keep in mind while configuring voice and video parameters:

- SIP CAC can be used for the 9971 Cisco phones that support TSPEC-based admission control. You can also use the phones that support Status code 17.
- SIP snooping is supported for providing voice priority to the non-TSPEC SIP phones.
- TSPEC for video CAC is not supported.

Information About Configuring Voice and Video Parameters

Three parameters on the controller affect voice and/or video quality:

- Call Admission Control
- Expedited bandwidth requests
- Unscheduled automatic power save delivery

CAC and UAPSD are supported on Cisco Compatible Extensions (CCX) v4 and v5, however, these parameters are also supported even without CCX but on any device implementing WMM (that supports 802.1e). Expedited bandwidth requests is supported only on CCXv5.

Traffic stream metrics (TSM) can be used to monitor and report issues with voice quality.

Call Admission Control

Call Admission Control (CAC) enables an access point to maintain controlled quality of service (QoS) when the wireless LAN is experiencing congestion. The WMM protocol deployed in CCXv4 maintains QoS under differing network loads.

Two types of Over The Air (OTA) CAC are available: static-based CAC and load-based CAC.

The controller supports the following QoS policies:

- User-defined policies: You can define your own QoS policies. You can have more control over these policies than the existing metal policies.
- System-defined precious metal policies: To support backward compatibility.
    - Platinum: Used for VoIP clients.
    - Gold: Used for video clients.
    - Silver: Used for best effort traffic.
    - Bronze: Used for NRT traffic.

Static-Based CAC

Voice over WLAN applications supporting WMM and TSPEC can specify how much bandwidth or shared medium time is required to initiate a call. Bandwidth-based, or static, CAC enables the access point to determine

whether it is capable of accommodating a particular call. The access point rejects the call if necessary in order to maintain the maximum allowed number of calls with acceptable quality.

The QoS setting for a WLAN determines the level of bandwidth-based CAC support. To use bandwidth-based CAC with voice applications, the WLAN must be configured for Platinum QoS. With bandwidth-based CAC, the access point bandwidth availability is determined based on the amount of bandwidth currently used by the access point clients, to which the bandwidth requested by the Voice over WLAN applications is added. If this total exceeds a configured bandwidth threshold, the new call is rejected.

Note You must enable admission control (ACM) for CCXv4 clients that have WMM enabled. Otherwise, bandwidth-based CAC does not operate properly for these CCXv4 clients.

Load-Based CAC

Load-based CAC incorporates a measurement scheme that takes into account the bandwidth consumed by all traffic types (including that from clients), co-channel access point loads, and co-allocated channel interference, for voice and video applications. Load-based CAC also covers the additional bandwidth consumption resulting from PHY and channel impairment.

In load-based CAC, the access point continuously measures and updates the utilization of the RF channel (that is, the mean time of bandwidth that has been exhausted), channel interference, and the additional calls that the access point can admit. The access point admits a new call only if the channel has enough unused bandwidth to support that call. By doing so, load-based CAC prevents oversubscription of the channel and maintains QoS under all conditions of WLAN loading and interference.

Note If you disable load-based CAC, the access points start using bandwidth-based CAC.

IOSd Call Admission Control

IOSd Call Admission Control (CAC) controls bandwidth availability from controller to access point.

You can configure class-based, unconditional packet marking features on your switch for CAC.

CAC is a concept that applies to voice and video traffic only—not data traffic. If an influx of data traffic oversubscribes a particular link in the network, queueing, buffering, and packet drop decisions resolve the congestion. The extra traffic is simply delayed until the interface becomes available to send the traffic, or, if traffic is dropped, the protocol or the end user initiates a timeout and requests a retransmission of the information.

Network congestion cannot be resolved in this manner when real-time traffic, sensitive to both latency and packet loss, is present, without jeopardizing the quality of service (QoS) expected by the users of that traffic. For real-time delay-sensitive traffic such as voice, it is better to deny network access under congestion conditions than to allow traffic onto the network to be dropped and delayed, causing intermittent impaired QoS and resulting in customer dissatisfaction.

CAC is therefore a deterministic and informed decision that is made before a voice call is established and is based on whether the required network resources are available to provide suitable QoS for the new call.

Based on the admit CAC CLI configuration in addition to the existing CAC algorithm, controller allows either voice or video with TSPEC or SIP snooping. The **admit cac** CLI is mandatory for the voice call to pass through.

If the BSSID policer is configured for the voice or video traffic, then additional checks are performed on the packets.

Expedited Bandwidth Requests

The expedited bandwidth request feature enables CCXv5 clients to indicate the urgency of a WMM traffic specifications (TSPEC) request (for example, an e911 call) to the WLAN. When the controller receives this request, it attempts to facilitate the urgency of the call in any way possible without potentially altering the quality of other TSPEC calls that are in progress.

You can apply expedited bandwidth requests to both bandwidth-based and load-based CAC. Expedited bandwidth requests are disabled by default. When this feature is disabled, the controller ignores all expedited requests and processes TSPEC requests as normal TSPEC requests.

The table below lists examples of TSPEC request handling for normal TSPEC requests and expedited bandwidth requests.

Table 13: TSPEC Request Handling Examples

CAC Mode	Reserved bandwidth for voice calls[1]	Usage[2]	Normal TSPEC Request	TSPEC with Expedited Bandwidth Request
Bandwidth-based CAC	75% (default setting)	Less than 75%	Admitted	Admitted
		Between 75% and 90% (reserved bandwidth for voice calls exhausted)	Rejected	Admitted
		More than 90%	Rejected	Rejected
Load-based CAC		Less than 75%	Admitted	Admitted
		Between 75% and 85% (reserved bandwidth for voice calls exhausted)	Rejected	Admitted
		More than 85%	Rejected	Rejected

1 For bandwidth-based CAC, the voice call bandwidth usage is per access point radio and does not take into account co-channel access points. For load-based CAC, the voice call bandwidth usage is measured for the entire channel.

2 Bandwidth-based CAC (consumed voice and video bandwidth) or load-based CAC (channel utilization [Pb]).

Note Admission control for TSPEC G711-20ms and G711-40 ms codec types are supported.

U-APSD

Unscheduled automatic power save delivery (U-APSD) is a QoS facility defined in IEEE 802.11e that extends the battery life of mobile clients. In addition to extending battery life, this feature reduces the latency of traffic flow delivered over the wireless media. Because U-APSD does not require the client to poll each individual packet buffered at the access point, it allows delivery of multiple downlink packets by sending a single uplink trigger packet. U-APSD is enabled automatically when WMM is enabled.

Traffic Stream Metrics

In a voice-over-wireless LAN (VoWLAN) deployment, traffic stream metrics (TSM) can be used to monitor voice-related metrics on the client-access point air interface. It reports both packet latency and packet loss. You can isolate poor voice quality issues by studying these reports.

The metrics consist of a collection of uplink (client side) and downlink (access point side) statistics between an access point and a client device that supports CCX v4 or later releases. If the client is not CCX v4 or CCXv5 compliant, only downlink statistics are captured. The client and access point measure these metrics. The access point also collects the measurements every 5 seconds, prepares 90-second reports, and then sends the reports to the controller. The controller organizes the uplink measurements on a client basis and the downlink measurements on an access point basis and maintains an hour's worth of historical data. To store this data, the controller requires 32 MB of additional memory for uplink metrics and 4.8 MB for downlink metrics.

TSM can be configured through either the GUI or the CLI on a per radio-band basis (for example, all 802.11a radios). The controller saves the configuration in flash memory so that it persists across reboots. After an access point receives the configuration from the controller, it enables TSM on the specified radio band.

This table shows the upper limit for TSM entries in different controller series.

TSM Entries	5700
MAX AP TSM entries	100
MAX Client TSM entries	250
MAX TSM entries	100*250=25000

Note Once the upper limit is reached, additional TSM entries cannot be stored and sent to WCS or NCS. If client TSM entries are full and AP TSM entries are available, then only the AP entries are stored, and viceversa. This leads to partial output. TSM cleanup occurs every one hour. Entries are removed only for those APs and clients that are not in the system.

Information About Configuring Voice Prioritization Using Preferred Call Numbers

You can configure a controller to provide support for SIP calls from VoWLAN clients that do not support TSPEC-based calls. This feature is known as SIP CAC support. If bandwidth is available in the configured voice pool, the SIP call uses the normal flow and the controller allocates the bandwidth to those calls.

You can also prioritize up to six preferred call numbers. When a call comes to one of the configured preferred numbers, the controller does not check the configured maximum voice bandwidth. The controller allocates the bandwidth needed for the call, even if it exceeds the maximum bandwidth for voice configured for voice CAC. The preferred call will be rejected if bandwitdh allocation exceeds 85% of the radio bandwidth. The bandwidth allocation is 85 percent of the entire bandwidth pool, not just from the maximum configured voice pool. The bandwidth allocation is the same even for roaming calls.

You must configure the following parameters before configuring voice prioritization:

- Set WLAN QoS to allow voice calls to pass through.
- Enable ACM for the radio.
- Enable SIP call snooping on the WLAN.

Information About EDCA Parameters

Enhanced distributed channel access (EDCA) parameters are designed to provide preferential wireless channel access for voice, video, and other quality-of-service (QoS) traffic.

How to Configure Voice and Video Parameters

Configuring Voice Parameters (CLI)

Before You Begin

Ensure that you have configured SIP-based CAC.

You should have created a class map for CAC before beginning this procedure.

SUMMARY STEPS

1. **show wlan summary**
2. **show wlan** *wlan_id*
3. **configure terminal**
4. **policy-map** *policy-map name*
5. **class** {*class-name* | **class-default**}
6. **admit cac wmm-tspec**
7. **service-policy** *policy-map name*
8. **end**
9. **wlan** *wlan_profile_name wlan_ID SSID_network_name* **wlan shutdown**
10. **wlan** *wlan_profile_name wlan_ID SSID_network_name*
11. **wlan** *wlan_name* **call-snoop**
12. **wlan** *wlan_name* **service-policy input** *input_policy_name*
13. **wlan** *wlan_name* **service-policy output** *ouput_policy_name*
14. **wlan** *wlan_name* **service-policy input** *ingress_policy_name*
15. **wlan** *wlan_name* **service-policy output** *egress_policy_name*
16. **ap dot11 {5ghz | 24ghz} shutdown**
17. **ap dot11 {5ghz | 24ghz} cac voice sip**
18. **ap dot11 {5ghz | 24ghz} cac voice acm**
19. **ap dot11 {5ghz | 24ghz} cac voice max-bandwidth** *bandwidth*
20. **ap dot11 {5ghz | 24ghz} cac voice roam-bandwidth** *bandwidth*
21. **no wlan shutdown**
22. **no ap dot11 {5ghz | 24ghz} shutdown**
23. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show wlan summary** **Example:** `Controller# show wlan summary`	Specifies all the WLANs configured on the controller.
Step 2	**show wlan** *wlan_id* **Example:** `Controller# show wlan 25`	Specifies the WLAN that you plan to modify. For voice over WLAN, ensure that the WLAN is configured for WMM and the QoS level is set to Platinum.
Step 3	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 4	**policy-map** *policy-map name*	Enters policy map configuration mode.

	Command or Action	Purpose

	Example:
	`Controller(config)# policy-map test_2000`
	`Controller(config-pmap)#`
		Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy.
		In WLAN, you need to configure service-policy for these commands to take effect.

Step 5	**class** {*class-name* | **class-default**}
	Example:
	`Controller(config-pmap)# class test_1000`
	`Controller(config-pmap-c)#`
		Enters policy class map configuration mode. Specifies the name of the class whose policy you want to create or change.
		Specifies the name of the class whose policy you want to create or change.
		You can also create a system default class for unclassified packets.

Step 6	**admit cac wmm-tspec**
	Example:
	`Controller(config-pmap-c)# admit cac wmm-tspec`
	`Controller(config-pmap-c)#`
		(Optional) Admits the request for Call Admission Control (CAC) for policy map.

Step 7	**service-policy** *policy-map name*
	Example:
	`Controller(config-pmap-c)# service-policy test_2000`
	`Controller(config-pmap-c)#`
		Configures the QoS service policy.

Step 8	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Step 9	**wlan** *wlan_profile_name wlan_ID SSID_network_name*
	wlan shutdown
	Example:
	`Controller(config)# wlan wlan1`
	`Controller(config-wlan)# wlan shutdown`
		Disables all WLANs with WMM enabled prior to changing the video parameters.

Step 10	**wlan** *wlan_profile_name wlan_ID SSID_network_name*
	Example:
	`Controller(config)# wlan wlan1`
	`Controller(config-wlan)# wlan shutdown`
		Disables all WLANs with WMM enabled prior to changing the voice parameters.

Step 11	**wlan** *wlan_name* **call-snoop** **Example:** `Controller(config)# wlan wlan1 call-snoop`	Enables the call-snooping on a particular WLAN.
Step 12	**wlan** *wlan_name* **service-policy input** *input_policy_name*	Configures input SSID policy on a particular WLAN to voice.

	Command or Action	Purpose

	Example:
	`Controller(config)# wlan wlan1`
	`Controller(config-wlan)# service-policy input platinum-up`	

Step 13	**wlan** *wlan_name* **service-policy output** *ouput_policy_name*
	Example:
	`Controller(config)# wlan wlan1`
	`Controller(config-wlan)# service-policy output platinum`
		Configures output SSID policy on a particular WLAN to voice.

Step 14	**wlan** *wlan_name* **service-policy input** *ingress_policy_name*
	Example:
	`Controller(config)# wlan wlan1`
	`Controller(config-wlan)# service-policy input policy1`
		Configures ingress SSID policy on a particular WLAN as user-defined policy.

Step 15	**wlan** *wlan_name* **service-policy output** *egress_policy_name*
	Example:
	`Controller(config)# wlan wlan1`
	`Controller(config-wlan)# service-policy output policy2`
		Configures egress SSID policy on a particular WLAN as user-defined policy.

Step 16	**ap dot11 {5ghz | 24ghz} shutdown** **Example:**	Disables the radio network. `Controller(config)# ap dot11 5ghz shutdown`
Step 17	**ap dot11 {5ghz | 24ghz} cac voice sip** **Example:** `Controller(config)# ap dot11 5ghz cac voice sip`	Enables or disables SIP IOSd CAC for the 802.11a or 802.11b/g network.
Step 18	**ap dot11 {5ghz | 24ghz} cac voice acm** **Example:** `Controller(config)# ap dot11 5ghz cac voice acm`	Enables or disables bandwidth-based voice CAC for the 802.11a or 802.11b/g network.
Step 19	**ap dot11 {5ghz | 24ghz} cac voice max-bandwidth** *bandwidth* **Example:** `Controller(config)# ap dot11 5ghz cac voice max-bandwidth 85`	Sets the percentage of maximum bandwidth allocated to clients for voice applications on the 802.11a or 802.11b/g network. The bandwidth range is 5 to 85%, and the default value is 75%. Once the client reaches the value specified, the access point rejects new videos on this network.
Step 20	**ap dot11 {5ghz | 24ghz} cac voice roam-bandwidth** *bandwidth*	Sets the percentage of maximum allocated bandwidth reserved for roaming voice clients.

	Command or Action	Purpose
	Example: `Controller(config)# ap dot11 5ghz cac voice roam-bandwidth 10`	The bandwidth range is 0 to 25%, and the default value is 6%. The controller reserves this much bandwidth from the maximum allocated bandwidth for roaming voice clients.
Step 21	**no wlan shutdown** **Example:** `Controller(config-wlan)# no wlan shutdown`	Reenables all WLANs with WMM enabled.
Step 22	**no ap dot11 {5ghz | 24ghz} shutdown** **Example:** `Controller(config)# no ap dot11 5ghz shutdown`	Reenables the radio network.
Step 23	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring Video Parameters (CLI)

SUMMARY STEPS

1. **show wlan summary**
2. **show wlan** *wlan_id*
3. **configure terminal**
4. **policy-map** *policy-map name*
5. **class** {*class-name* | **class-default**}
6. **admit cac wmm-tspec**
7. **service-policy** *policy-map name*
8. **end**
9. **wlan***wlan_profile_name*
10. **ap dot11 {5ghz | 24ghz} shutdown**
11. **ap dot11 {5ghz | 24ghz} cac video acm**
12. **ap dot11 {5ghz | 24ghz} cac video load-based**
13. **ap dot11 {5ghz | 24ghz} cac video max-bandwidth** *bandwidth*
14. **ap dot11 {5ghz | 24ghz} cac video roam-bandwidth** *bandwidth*
15. **no wlan shutdown** *wlan_id*
16. **no ap dot11 {5ghz | 24ghz} shutdown**
17. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show wlan summary** **Example:** `Controller# show wlan summary`	Specifies all the WLANs configured on the controller.
Step 2	**show wlan** *wlan_id* **Example:** `Controller# show wlan 25`	Specifies the WLAN that you plan to modify.
Step 3	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.

Step 4	**policy-map** *policy-map name*
	
	Example:
	`Controller(config)# policy-map test_2000`
	`Controller(config-pmap)#`
		Enters policy map configuration mode.
		Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy.
		In WLAN, you need to configure service-policy for these commands to take effect.

Step 5	**class** {*class-name* | **class-default**}
	
	Example:
	`Controller(config-pmap)# class test_1000`
	`Controller(config-pmap-c)#`
		Enters policy class map configuration mode. Specifies the name of the class whose policy you want to create or change.
		Specifies the name of the class whose policy you want to create or change.
		You can also create a system default class for unclassified packets.

Step 6	**admit cac wmm-tspec**
	
	Example:
	`Controller(config-pmap-c)# admit cac wmm-tspec`
	`Controller(config-pmap-c)#`
		(Optional) Admits the request for Call Admission Control (CAC) for policy map.

Step 7	**service-policy** *policy-map name*
	
	Example:
	`Controller(config-pmap-c)# service-policy test_2000`
	`Controller(config-pmap-c)#`
		Configures the QoS service policy.

Step 8	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

	Command or Action	Purpose

Step 9	**wlan***wlan_profile_name*
	
	Example:
	`Controller(config)# wlan wlan1`
	`Controller(config-wlan)# wlan shutdown`
		Disables all WLANs with WMM enabled prior to changing the video parameters.

Step 10	**ap dot11 {5ghz | 24ghz} shutdown** **Example:** `Controller(config)# ap dot11 5ghz shutdown`	Disables the radio network.
Step 11	**ap dot11 {5ghz | 24ghz} cac video acm** **Example:** `Controller(config)# ap dot11 5ghz cac video acm`	Enables or disables bandwidth-based video CAC for the 802.11a or 802.11b/g network.
Step 12	**ap dot11 {5ghz | 24ghz} cac video load-based** **Example:** `Controller(config)# ap dot11 5ghz cac video load-based`	Configures the load-based CAC method. If you do not enter this command, then the default static CAC is applied.
Step 13	**ap dot11 {5ghz | 24ghz} cac video max-bandwidth** *bandwidth* **Example:** `Controller(config)# ap dot11 5ghz cac video max-bandwidth 20`	Sets the percentage of maximum bandwidth allocated to clients for video applications on the 802.11a or 802.11b/g network. The bandwidth range is 5 to 85%, and the default value is 75%. The default value is 0, which means no bandwidth request control. The sum of the voice bandwidth and video bandwidth should not exceed 85% or configured maximum media bandwidth.
Step 14	**ap dot11 {5ghz | 24ghz} cac video roam-bandwidth** *bandwidth* **Example:** `Controller(config)# ap dot11 5ghz cac video roam-bandwidth 9`	Sets the percentage of maximum allocated bandwidth reserved for roaming clients for video. The bandwidth range is 0 to 25%, and the default value is 0%.
Step 15	**no wlan shutdown** *wlan_id* **Example:** `Controller(config-wlan)# no wlan shutdown 25`	Reenables all WLANs with WMM enabled.
Step 16	**no ap dot11 {5ghz | 24ghz} shutdown** **Example:** `Controller(config)# no ap dot11 5ghz shutdown`	Reenables the radio network.
Step 17	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring SIP-Based CAC (CLI)

SIP CAC controls the total number of SIP calls that can be made.

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan-name*
3. **call-snoop**
4. **service-policy** [**client**] **input** *policy-map name*
5. **service-policy** [**client**] **output** *policy-map name*
6. **end**
7. **show wlan** {*wlan-id* | *wlan-name*}
8. **configure terminal**
9. **ap dot11 {5ghz | 24ghz} cac {voice | video} acm**
10. **ap dot11 {5ghz | 24ghz} cac voice sip**
11. **end**

DETAILED STEPS

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

Step 2	**wlan** *wlan-name*
	
	Example:
	`Controller(config)# wlan qos-wlan`
	`Controller(config-wlan)#`
		Enters the WLAN configuration submode.

Step 3	**call-snoop** **Example:** `Controller(config-wlan)# call-snoop`	Enables the call-snooping feature for a particular WLAN.
Step 4	**service-policy** [**client**] **input** *policy-map name* **Example:** `Controller(config-wlan)# service-policy input platinum-up`	Assigns a policy map to WLAN input traffic. Ensure that you provide QoS policy to voice for input traffic.

	Command or Action	Purpose
Step 5	**service-policy** [**client**] **output** *policy-map name* **Example:** `Controller(config-wlan)# service-policy output platinum`	Assigns policy map to WLAN output traffic. Ensure that you provide QoS policy to voice for output traffic.
Step 6	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.
Step 7	**show wlan** {*wlan-id* | *wlan-name*} **Example:** `Controller# show wlan qos-wlan`	Verifies the configured QoS policy on the WLAN.
Step 8	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 9	**ap dot11 {5ghz | 24ghz} cac {voice | video} acm** **Example:** `Controller(config)# ap dot11 5ghz cac voice acm`	Enables the ACM static on the radio. When enabling SIP snooping, use the static CAC, not the load-based CAC.
Step 10	**ap dot11 {5ghz | 24ghz} cac voice sip** **Example:** `Controller(config)# ap dot11 5ghz cac voice sip`	Configures SIP-based CAC.
Step 11	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring a Preferred Call Number (CLI)

Before You Begin

You must set the following parameters before configuring a preferred call number.

- Set WLAN QoS to voice.
- Enable ACM for the radio.
- Enable SIP call snooping on the WLAN.
- Enable SIP-based CAC.

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan-name* **qos platinum**
3. **ap dot11 {5ghz | 24ghz} cac {voice | video} acm**
4. **wlan** *wlan-name*
5. **wireless sip preferred-call-no** *call_index call_number*
6. **no wireless sip preferred-call-no** *call_index*
7. **end**

DETAILED STEPS

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

Step 2	**wlan** *wlan-name* **qos platinum**
	
	Example:
	`Controller(config)# wlan wlan1`
	`Controller(config-wlan)# qos platinum`
		Sets QoS to voice on a particular WLAN.

Step 3	**ap dot11 {5ghz | 24ghz} cac {voice | video} acm** **Example:** `Controller(config)# ap dot11 5ghz cac voice acm`	Enables the static ACM on the radio. When enabling SIP snooping, use the static CAC, not the load-based CAC.

Step 4	**wlan** *wlan-name*
	
	Example:
	`Controller(config)# wlan wlan1`
	`Controller(config-wlan)# call-snoop`
		Enables the call-snooping feature for a particular WLAN.

Step 5	**wireless sip preferred-call-no** *call_index call_number* **Example:** `Controller(config)# wireless sip preferred-call-no 1 555333`	Adds a new preferred call.
Step 6	**no wireless sip preferred-call-no** *call_index* **Example:** `Controller(config)# no wireless sip preferred-call-no 1`	Removes a preferred call.
Step 7	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring EDCA Parameters (CLI)

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 {5ghz | 24ghz } shutdown**
3. **ap dot11 {5ghz | 24ghz} edca-parameters {custom-voice | optimized-video-voice | optimized-voice | svp-voice | wmm-default}**
4. **show ap dot11 {5ghz | 24ghz} network**
5. **no ap dot11 {5ghz | 24ghz} shutdown**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 {5ghz | 24ghz } shutdown** **Example:** `Controller(config)# ap dot11 5ghz shutdown`	Disables the radio network.
Step 3	**ap dot11 {5ghz | 24ghz} edca-parameters {custom-voice | optimized-video-voice | optimized-voice | svp-voice | wmm-default}** **Example:** `Controller(config)# ap dot11 5ghz edca-parameters optimized-voice`	Enables a specific EDCA parameters for the 802.11a or 802.11b/g network. • custom-voice—Enables custom voice parameters for the 802.11a or 802.11b/g network. • optimized-video-voice—Enables EDCA voice- and video-optimized parameters for the 802.11a or 802.11b/g network. Choose this option when both voice and video services are deployed on your network. • optimized-voice—Enables non-SpectraLink voice-optimized profile parameters for the 802.11a or 802.11b/g network. Choose this option when voice services other than SpectraLink are deployed on your network. • svp-voice—Enables SpectraLink voice priority parameters for the 802.11a or 802.11b/g network. Choose this option if SpectraLink phones are deployed on your network to improve the quality of calls.

	Command or Action	Purpose
		• wmm-default—Enables the Wi-Fi Multimedia (WMM) default parameters for the 802.11a or 802.11b/g network. This is the default value. Choose this option when voice or video services are not deployed on your network.
Step 4	**show ap dot11 {5ghz | 24ghz} network** **Example:** `Controller(config)# show ap dot11 5ghz network`	Displays the current status of MAC optimization for voice.
Step 5	**no ap dot11 {5ghz | 24ghz} shutdown** **Example:** `Controller(config)# no ap dot11 5ghz shutdown`	Reenables the radio network
Step 6	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Monitoring Voice and Video Parameters

This section describes the new commands for the voice and video parameters.

The following command can be used to monitor voice and video parameters.

Table 14: Monitoring Voice Parameters Commands

Command	Purpose
show ap dot11 {5ghz | 24ghz} network	Displays the radio-based statistics for voice.
show ap name *ap_name* **dot11 24ghz tsm all**	Displays the TSM voice metrics and current status of MAC optimization for voice.
show ap name *apname* **cac voice**	Displays the information about CAC for a particular access point.
show client detail *client_mac*	Displays the U-APSD status for a particular client.
show policy-map interface wireless client	Displays the video client policy details.
show access-list	Displays the video client dynamic access-list from the controller.

show wireless client voice diag status	Displays information about whether voice diagnostics are enabled or disabled. If enabled, this also displays information about the clients in the watch list and the time remaining for the diagnostics of the voice call. **Note** To work on voice diagnostics CLIs, you need to enter the following command: **debug voice-diagnostic mac-addr** *client_mac_01 client_mac_02*
show wireless client voice diag tspec	Displays the TSPEC information sent from the clients that are enabled for voice diagnostics.
show wireless client voice diag qos-map	Displays information about the QoS/DSCP mapping and packet statistics in each of the four queues: VO, VI, BE, BK. The different DSCP values are also displayed.
show wireless client voice diag rssi	Display the client's RSSI values in the last 5 seconds when voice diagnostics is enabled.
show client voice-diag roam-history	Displays information about the last three roaming calls. The output contains the timestamp, access point associated with roaming, roaming reason, and if there is a roaming failure, reason for roaming-failure.
show policy-map interface wireless mac *mac-address*	Displays information about the voice and video data packet statistics.
show wireless media-stream client summary	Displays a summary of the media stream and video client information.
show controllers d0 | b queue	Displays which queue the packets are going through on an access point.
show platform qos queue stats *interface*	Displays which queue packets are going through from the controller.

You can monitor the video parameters using the following commands.

Table 15: Monitoring Video Parameters Commands

Command	Purpose
show ap join stats summary *ap_mac*	Displays the last join error detail for a specific access point.
show ip igmp snooping wireless mgid	Displays the TSM voice metrics and current status of MAC optimization for voice.

show wireless media-stream multicast-direct state	Displays the media stream multicast-direct parameters.
show wireless media-stream group summary	Displays the summary of the media stream and client information.
show wireless media-stream group detail *group_name*	Displays the details of a specific media-stream group.
show wireless media-stream client summary	Displays the details for a set of media-stream clients.
show wireless media-stream client detail *group_name*	Displays the details for a set of media-stream clients.
show ap dot11 {5ghz | 24ghz) media-stream rrc	Display the details of media stream.
show wireless media-stream message details	Displays information about the message configuration.
show ap name *ap-name* **auto-rf dot11 5ghz | i Util**	Displays the details of channel utilization.
show controllers d0 | b queue	Displays which queue the packets are going through on an access point based on 2.4- and 5-GHz bands.
show controllers d1 | b queue	Displays which queue the packets are going through on an access point based on 2.4- and 5-GHz bands.
show cont d1 | b Media	Displays the video metric details on the band A or B.
show capwap mcast mgid all	Displays information about all the multicast groups and their corresponding multicast group identifications (MGIDs) associated to the access point.
show capwap mcast mgid id *id*	Displays information about all the video clients joined to the multicast group in a specific MGID.

Additional References for Voice and Video Parameters

Related Documents

Related Topic	Document Title
Multicast configuration	*Multicast Configuration Guide, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*
VideoStream configuration	*VideoStream Configuration Guide, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

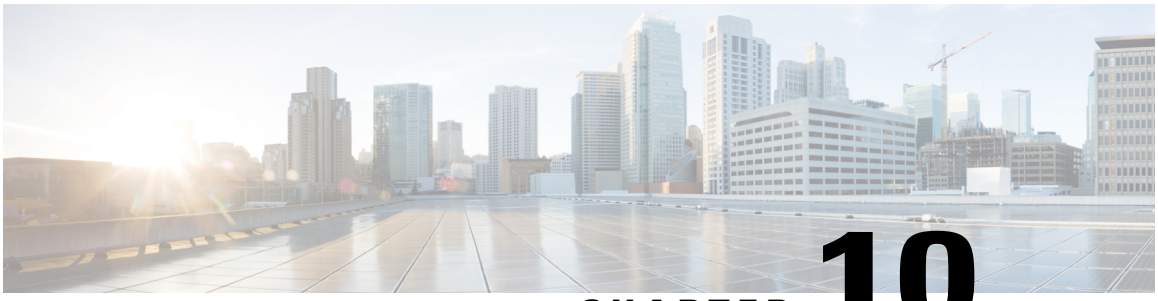
MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

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 Voice and Video Parameters Configuration

Release	Modification
Cisco IOS XE Release 3.2SE	This feature was introduced.



CHAPTER 10

Configuring RFID Tag Tracking

This module contains the following topics:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Configuring RFID Tag Tracking

The Controller enables you to configure radio-frequency identification (RFID) tag tracking. RFID tags are small wireless devices that are affixed to assets for real-time location tracking. They operate by advertising their location using special 802.11 packets, which are processed by access points, the controller, and the location appliance.

How to Configure RFID Tag Tracking

Configuring RFID Tag Tracking (CLI)

SUMMARY STEPS

1. **location rfid status**
2. (Optional) **no location rfid status**
3. **location rfid timeout** *seconds*
4. **location rfid mobility vendor-name** *name*
5. (Optional) **no location rfid mobility** *name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**location rfid status** **Example:** `Controller(config)# location rfid status`	Enables RFID tag tracking. By default, RFID tag tracking is enabled.
Step 2	(Optional) **no location rfid status** **Example:** `Controller(config)# no location rfid status`	Disables RFID tag tracking.
Step 3	**location rfid timeout** *seconds* **Example:** `Controller(config)# location rfid timeout 1500`	Specifies a static timeout value (between 60 and 7200 seconds). The static timeout value is the amount of time that the controller maintains tags before expiring them. For example, if a tag is configured to beacon every 30 seconds, we recommend that you set the timeout value to 90 seconds (approximately three times the beacon value). The default value is 1200 seconds.
Step 4	**location rfid mobility vendor-name** *name* **Example:** `Controller(config)# location rfid mobility vendor-name Aerosct`	Enables RFID tag mobility for specific tags. When you enter the **location rfid mobility vendor-name** command, tags are unable to obtain a DHCP address for client mode when attempting to select and/or download a configuration. **Note** These commands can be used only for Pango tags. Therefore, the only valid entry for **vendor_name** is “pango” in all lowercase letters.
Step 5	(Optional) **no location rfid mobility** *name* **Example:** `Controller(config)# no location rfid mobility test`	Disables RFID tag mobility for specific tags. When you enter the **no location rfid mobility** command , tags can obtain a DHCP address. If a tag roams from one subnet to another, it obtains a new address rather than retaining the anchor state.

Monitoring RFID Tag Tracking Information

This section describes the new commands for the RFID Tag Tracking Information.

The following command can be used to monitor the RFID Tag Tracking Information on the controller.

Table 16: Monitoring RFID Tag Tracking Information Commands

Command	Purpose
show location rfid config	Displays the current configuration for RFID tag tracking.
show location rfid detail *mac_address*	Displays the detailed information for a specific RFID tag.
show location rfid summary	Displays a list of all RFID tags currently connected to the controller.
show location rfid client	Displays a list of RFID tags that are associated to the controller as clients.

Additional References RFID Tag Tracking

Related Documents

Related Topic	Document Title
System management commands	*System Management Command Reference, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

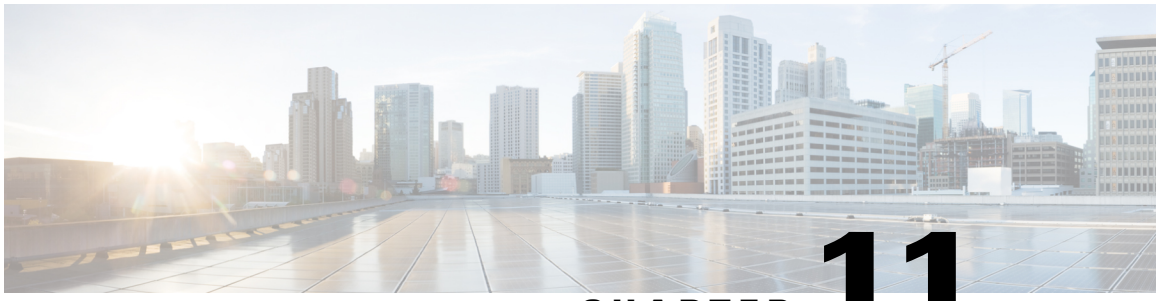
MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

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 RFID Tag Tracking Configuration

Release	Modification
Cisco IOS XE Release 3.2SE	This feature was introduced.



CHAPTER 11

Configuring Location Settings

This module contains the following topics:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Configuring Location Settings

The controller determines the location of client devices by gathering received signal strength indication (RSSI) measurements from access points all around the client of interest. The controller can obtain location reports from up to 16 access points for clients, RFID tags, and rogue access points.

You can configure the path loss measurement (S60) request for normal clients or calibrating clients to improve location accuracy.

How to Configure Location Settings

Configuring Location Settings (CLI)

SUMMARY STEPS

1. **configure terminal**
2. **location plm** {**calibrating** [**multiband** | **uniband**] | **client** *burst_interval*
3. **location rssi-half-life** {**calibrating-client** | **client** | **rogue-aps** | **tags** } *seconds*
4. **location expiry** {**calibrating-client** | **client** | **rogue-aps** | **tags** } *timeout*
5. **location algorithm** {**rssi-average** | **simple**}
6. **location admin-tag** *string*
7. **location civic-location identifier** {*identifier* | **host**}
8. **location custom-location identifier** {*identifier* | **host**}
9. **location geo-location identifier** {*identifier* | **host**}
10. **location prefer** {**cdp** | **lldp-med** | **static**} **weight** *priority_value*
11. **location rfid** {**status** | **timeout** | **vendor-name**}
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**location plm** {**calibrating** [**multiband** | **uniband**] | **client** *burst_interval* **Example:** `Controller(config)# location plm client 100`	Configures the path loss measurement (S60) request for calibrating clients or non-calibrating. The path loss measurement request improves the location accuracy. You can configure the **burst_interval** parameter for the normal, noncalibrating client from zero through 3600 seconds, and the default value is 60 seconds. You can configure the path loss measurement request for calibrating clients on the associated 802.11a or 802.11b/g radio or on the associated 802.11a/b/g radio. If a client does not send probes often or sends them only on a few channels, its location cannot be updated or cannot be updated accurately. The **location plm** command forces clients to send more packets on all channels. When a CCXv4 (or higher) client associates, the Controller sends it a path loss measurement request, which instructs the client to transmit on the bands and channels that the access points are on (typically, channels 1, 6, and 11 for 2.4-GHz-only access points) at a configurable interval (such as 60 seconds) indefinitely.

	Command or Action	Purpose
Step 3	**location rssi-half-life** {**calibrating-client** | **client** | **rogue-aps** | **tags** } *seconds* **Example:** `Controller(config)# location rssi-half-life calibrating-client 60`	Configures the RSSI half life for the clients, calibrating clients, RFID tags, and rogue access points. You can enter the **location rssi-half-life** parameter value for the clients, calibrating clients, RFID tags, and rogue access points as 0, 1, 2, 5, 10, 20, 30, 60, 90, 120, 180, or 300 seconds, and the default value is 0 seconds. Some client devices transmit at reduced power immediately after changing channels, and RF is variable, so RSSI values might vary considerably from packet to packet. The **location rssi-half-life** command increases accuracy by averaging nonuniformly arriving data using a configurable forget period (or half life). **Note** We recommend that you do not use or modify the **location rssi-half-life** command.
Step 4	**location expiry** {**calibrating-client** | **client** | **rogue-aps** | **tags** } *timeout* **Example:** `Controller(config)# location expiry calibrating-client 50`	Configures the RSSI timeout value for the clients, calibrating clients, RFID tags, and rogue access points. You can enter the RSSI timeout value for the clients, RFID tags, and rogue access points from 5 through 3600 seconds, and the default value is 5 seconds. For the calibrating clients, you can enter the RSSI timeout value from 0 through 3600 seconds, and the default value is 5 seconds. Ensuring that recent, strong RSSIs are retained by the CPU is critical to location accuracy. The **location expiry** command enables you to specify the length of time after which old RSSI averages expire. **Note** We recommend that you do not use or modify the **location expiry** command.
Step 5	**location algorithm** {**rssi-average** | **simple**} **Example:** `Controller(config)# location algorithm rssi-average`	Configures the algorithm used to average RSSI and signal-to-noise ratio (SNR) values. You can enter the **location algorithm rssi-average** command to specify a more accurate algorithm but requires more CPU overhead or the **location algorithm simple** command to specify a faster algorithm that requires low CPU overhead but provides less accuracy. **Note** We recommend that you do not use or modify the **location algorithm** command.
Step 6	**location admin-tag** *string* **Example:** `Controller(config)# location admin-tag`	Sets administrative tag or site information for the location of client devices.
Step 7	**location civic-location identifier** {*identifier* | **host**} **Example:** `Controller(config)# location civic-location identifier host`	Specifies civic location information. You can set the civic location identifier either as a string or host.

	Command or Action	Purpose
Step 8	**location custom-location identifier** {*identifier* | **host**} **Example:** `Controller(config)# location custom-location identifier host`	Specifies custom location information. You can set the custom location identifier either as a string or host.
Step 9	**location geo-location identifier** {*identifier* | **host**} **Example:** `Controller(config)# location geo-location identifier host`	Specifies geographical location information of the client devices. You can set the location identifier either as a string or host.
Step 10	**location prefer** {**cdp** | **lldp-med** | **static**} **weight** *priority_value* **Example:** `Controller(config)# location prefer weight cdp 50`	Sets location information source priority. You can enter the priority weight from zero through 255.
Step 11	**location rfid** {**status** | **timeout** | **vendor-name**} **Example:** `Controller(config)# location rfid timeout 100`	Configures RFID tag tracking options such as RFID tag status, RFID timeout value, and RFID tag vendor name. You can enter the RFID timeout value in a range from 60 and 7200 seconds.
Step 12	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Modifying the NMSP Notification Interval for Clients, RFID Tags, and Rogues (CLI)

The Network Mobility Services Protocol (NMSP) manages communication between the mobility services engine and the controller for incoming and outgoing traffic. If your application requires more frequent location updates, you can modify the NMSP notification interval (to a value between 1 and 180 seconds) for clients, active RFID tags, and rogue access points and clients.

Note The TCP port (16113) that the controller and mobility services engine communicate over must be open (not blocked) on any firewall that exists between the controller and the mobility services engine for NMSP to function.

SUMMARY STEPS

1. **configure terminal**
2. **nmsp notification interval** {**attachment** *seconds* | **location** *seconds* | **rssi** [**clients** *interval* | **rfid** *interval* | **rogues** [**ap** | **client**] *interval*]}
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**nmsp notification interval** {**attachment** *seconds* | **location** *seconds* | **rssi** [**clients** *interval* | **rfid** *interval* | **rogues** [**ap** | **client**] *interval*]} **Example:** `Controller(config)# nmsp notification interval rssi rfid 50`	Sets the NMSP notification interval value for clients, RFID tags, and rogue clients and access points. You can enter the NMSP notification interval value for RSSI measurement from 1 through 180 seconds.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Modifying the NMSP Notification threshold for Clients, RFID Tags, and Rogues (CLI)

SUMMARY STEPS

1. **configure terminal**
2. **location notify-threshold** {**clients** | **rogues ap** | **tags** } *threshold*
3. **end**

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	**location notify-threshold** {**clients** | **rogues ap** | **tags** } *threshold* **Example:** `Controller(config)# location notify-threshold clients 5`	Configures the NMSP notification threshold for clients, RFID tags, and rogue clients and access points. You can enter the RSSI threshold value from zero through 10 db.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Monitoring Location Settings and NMSP Settings

Monitoring Location Settings (CLI)

This section describes the new commands for location settings.

The following commands can be used to monitor location settings on the controller.

Table 17: Monitoring Location Settings Command

Command	Purpose
show location summary	Displays the current location configuration values.
show location statistics rfid	Displays the location-based RFID statistics.
show location detail *client_mac_addr*	Displays the RSSI table for a particular client.

Monitoring NMSP Settings (CLI)

This section describes the new commands for NMSP settings.

The following commands can be used to monitor NMSP settings on the controller.

Table 18: Monitoring NMSP Settings Command

Command	Purpose
show nmsp attachment suppress interfaces	Displays the attachment suppress interfaces.

show nmsp capability	Displays the NMSP capabilities.
show nmsp notification interval	Displays the NMSP notification intervals.
show nmsp statistics connection	Displays the connection-specific NMSP counters.
show nmsp statistics summary	Displays the common NMSP counters.
show nmsp status	Displays the status of active NMSP connections.
show nmsp subscription detail	Displays all of the mobility services to which the controller is subscribed.
show nmsp subscription detail *ip_addr*	Displays details only for the mobility services subscribed to by a specific IP address.
show nmsp subscription summary	Displays details for all of the mobility services to which the controller is subscribed.

Examples: Location Settings Configuration

This example shows how to configure the path loss measurement (S60) request for calibrating client on the associated 802.11a or 802.11b/g radio:

```
Controller# configure terminal
Controller(config)# location plm calibrating uniband
Controller(config)# end
Controller# show location summary
```

This example shows how to configure the RSSI half life for a rouge access point:

```
Controller# configure terminal
Controller(config)# location rssi-half-life rogue-aps 20
Controller(config)# end
Controller# show location summary
```

Examples: NMSP Settings Configuration

This example shows how to configure the NMSP notification interval for RFID tags:

```
Controller# configure terminal
Controller(config)# nmsp notification interval rssi rfid 50
Controller(config)# end
Controller# show nmsp notification interval
```

This example shows how to configure the NMSP notification threshold for clients:

```
Controller# configure terminal
Controller(config)# nmsp notify-threshold 5
Controller(config)# end
Controller# show nmsp statistics summary
```

Additional References for Location Settings

Related Documents

Related Topic	Document Title
System management commands	*System Management Command Reference, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

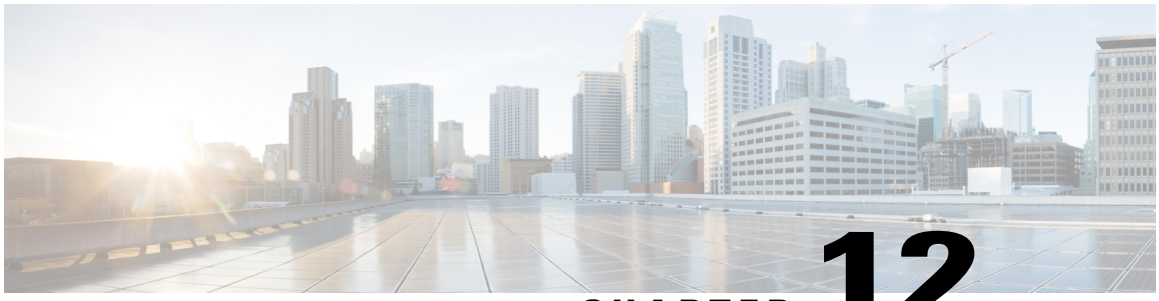
MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

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 Location Settings Configuration

Release	Modification
Cisco IOS XE Release 3.2SE	This feature was introduced.



CHAPTER 12

Configuring System Message Logs

This module contains the following topics:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Configuring System Message Logs

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 the **service sequence-numbers**, **service timestamps log datetime**, **service timestamps log datetime** [**localtime**] [**msec**] [**show-timezone**], or **service timestamps log uptime** global configuration command.

Table 19: 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.
hostname-n	Hostname of a stack member and its switch number in the stack. Though the is a stack member, it does *not* append its hostname to system messages.

Default System Message Logging Settings

Table 20: 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.

Feature	Default Setting
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.

How to Configure System Message 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:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**logging buffered** [*size*] **Example:** `Controller(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 . The range is 4096 to 2147483647 bytes. The default buffer size is 4096 bytes. If a standalone switch or the 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:** `Controller(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:** `Controller(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 . • *filename*—Enters the log message filename. • (Optional) **max-file-size** —Specifies the maximum logging file size. The range is 4096 to 2147483647. The default is 4096 bytes. • (Optional) *min-file-size*—Specifies the minimum logging file size. The range is 1024 to 2147483647. The default is 2048 bytes. • (Optional) *severity-level-number*	*type*—Specifies either the logging severity level or the logging type. The severity range is 0 to 7.

	Command or Action	Purpose
Step 5	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 6	**terminal monitor** **Example:** `Controller# 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:** `Controller# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**line** [**console** | **vty**] *line-number* [*ending-line-number*] **Example:** `Controller(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 Or you can change the setting of the single vty line being used for your current connection. For example, to change the setting for vty line 2, enter: line vty 2 When you enter this command, the mode changes to line configuration.
Step 3	**logging synchronous** [**level** [*severity-level* | **all**] | **limit** *number-of-buffers*] **Example:** `Controller(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:** `Controller(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 re-enable 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:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**no logging console** **Example:** `Controller(config)# no logging console`	Disables message logging.
Step 3	**end** **Example:** `Controller(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:** `Controller# 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:
	`Controller(config)# service timestamps log uptime`
	or
	`Controller(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:** `Controller(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:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**service sequence-numbers** **Example:** `Controller(config)# service sequence-numbers`	Enables sequence numbers.
Step 3	**end** **Example:** `Controller(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:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**logging console** *level* **Example:** `Controller(config)# logging console 3`	Limits messages logged to the console. By default, the console receives debugging messages and numerically lower levels.
Step 3	**logging monitor** *level* **Example:** `Controller(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:** `Controller(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:** `Controller(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:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**logging history** *level* **Example:** `Controller(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.
Step 3	**logging history size** *number* **Example:** `Controller(config)# logging history size 200`	Specifies the number of syslog messages that can be stored in the history table. The default is to store one message. The range is 0 to 500 messages.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

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. **Example:** `local7.debug /usr/adm/logs/`*cisco.log*	• **local7—**Specifies the logging facility. • **debug—**Specifies the syslog level. The file must already exist, and the syslog daemon must have permission to write to it.
Step 2	Enter these commands at the UNIX shell prompt. **Example:** `$ touch /var/log/`*cisco.log* `$ chmod 666 /var/log/`*cisco.log*	Creates the log file. The syslog daemon sends messages at this level or at a more severe level to this file.
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.

Monitoring and Maintaining System Message 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.

Configuration Examples for System Message Logs

Stacking System Message: Example

This example shows a partial switch system message for 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)
```

Switch System Message: Example

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)
```

Additional References for System Message Logs

Related Documents

Related Topic	Document Title
System message log commands	
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

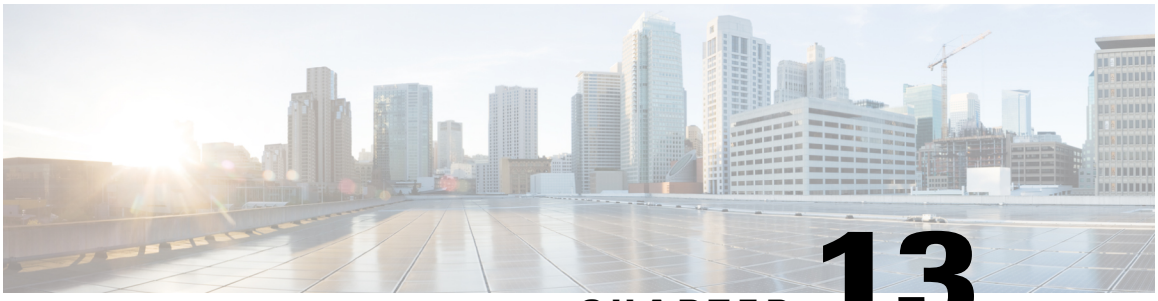
MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

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
	This feature was introduced.



CHAPTER 13

Configuring Online Diagnostics

This module contains the following topics:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 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** | **complete** | **minimal** | **non-disruptive** | **per-port**}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**diagnostic start switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all** | **basic** | **complete** | **minimal** | **non-disruptive** | **per-port**} **Example:** `Controller# 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 4. 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. • **complete**—Starts the complete test suite.

	Command or Action	Purpose
		• **minimal—**Starts the minimal bootup test suite. • **non-disruptive—**Starts the non-disruptive test suite. • **per-port—**Starts the per-port 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** | **complete** | **minimal** | **non-disruptive** | **per-port**} {**daily** | **on** *mm dd yyyy hh:mm* | **port** *inter-port-number port-number-list* | **weekly** *day-of-week hh:mm*}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**diagnostic schedule switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all** | **basic** | **complete** | **minimal** | **non-disruptive** | **per-port**} {**daily** | **on** *mm dd yyyy hh:mm* | **port** *inter-port-number port-number-list* | **weekly** *day-of-week hh:mm*} **Example:** `Controller(config)# diagnostic schedule switch 3 test 1,2,4-6 on november 3 2006 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 4. 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.

	Command or Action	Purpose
		• **all—**All test IDs. • **basic—**Starts the basic on-demand diagnostic tests. • **complete—**Starts the complete test suite. • **minimal—**Starts the minimal bootup test suite. • **non-disruptive—**Starts the non-disruptive test suite. • **per-port—**Starts the per-port 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.

By default, health monitoring is disabled, but the switch generates a syslog message when a test fails.

SUMMARY STEPS

1. **configure terminal**
2. **diagnostic monitor interval switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all**} *hh:mm:ss milliseconds day*
3. **diagnostic monitor syslog**
4. **diagnostic monitor threshold switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all**} **failure count** *count*
5. **diagnostic monitor switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all**}
6. **end**

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	**diagnostic monitor interval switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all**} *hh:mm:ss milliseconds day* **Example:** `Controller(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. The range is from 1 to 4. 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: • *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 3	**diagnostic monitor syslog** **Example:** `Controller(config)# diagnostic monitor syslog`	(Optional) Configures the switch to generate a syslog message when a health-monitoring test fails.
Step 4	**diagnostic monitor threshold switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all**} **failure count** *count* **Example:** `Controller(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 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. The range for the failure threshold *count* is 0 to 99.

	Command or Action	Purpose
Step 5	**diagnostic monitor switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all**} **Example:** `Controller(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.
Step 6	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

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 ro switch stack and check the test results by using the privileged EXEC **show** commands in this table:

Table 21: 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

Start Diagnostic Tests: Examples

This example shows how to start a diagnostic test by using the test name:

```
Controller# diagnostic start switch 2 test TestInlinePwrCtlr
```

This example shows how to start all of the basic diagnostic tests:

```
Controller# diagnostic start switch 1 test all
```

Configure a Health Monitoring Test: Example

This example shows how to configure a health-monitoring test:

```
Controller(config)# diagnostic monitor threshold switch 3 test 1 failure count 50
Controller(config)# diagnostic monitor interval switch 3 test TestPortAsicRingLoopback
```

Schedule Diagnostic Test: Examples

This example shows how to schedule diagnostic testing for a specific day and time on a nonstacking switch:

```
Controller(config)# diagnostic schedule test TestPortAsicCam on december 3 2006 22:25
```

This example shows how to schedule diagnostic testing to occur weekly at a specific time on member switch 6 when this command is entered on switch:

```
Controller(config)# diagnostic schedule switch 6 test 1-4,7 weekly saturday 10:30
```

Displaying Online Diagnostics: Examples

This example shows how to display on demand diagnostic settings:

```
Controller# show diagnostic ondemand settings

Test iterations = 1
Action on test failure = continue
```

This example shows how to display diagnostic events for errors:

```
Controller# show diagnostic events event-type error

Diagnostic events (storage for 500 events, 0 events recorded)
Number of events matching above criteria = 0

No diagnostic log entry exists.
```

This example shows how to display the description for a diagnostic test:

```
Controller# show diagnostic description switch 1 test all

DiagGoldPktTest :
        The GOLD packet Loopback test verifies the MAC level loopback
        functionality. In this test, a GOLD packet, for which doppler
        provides the support in hardware, is sent. The packet loops back
        at MAC level and is matched against the stored packet. It is a non
        -disruptive test.

DiagThermalTest :
        This test verifies the temperature reading from the sensor is below the yellow
        temperature threshold. It is a non-disruptive test and can be run as a health
monitoring test.

DiagFanTest :
        This test verifies all fan modules have been inserted and working properly on the
board
        It is a non-disruptive test and can be run as a health monitoring test.

DiagPhyLoopbackTest :
        The PHY Loopback test verifies the PHY level loopback
        functionality. In this test, a packet is sent which loops back
        at PHY level and is matched against the stored packet. It is a
        disruptive test and cannot be run as a health monitoring test.

DiagScratchRegisterTest :
        The Scratch Register test monitors the health of application-specific
        integrated circuits (ASICs) by writing values into registers and reading
        back the values from these registers. It is a non-disruptive test and can
        be run as a health monitoring test.

DiagPoETest :
        This test checks the PoE controller functionality. This is a disruptive test
        and should not be performed during normal switch operation.

DiagStackCableTest :
        This test verifies the stack ring loopback functionality
        in the stacking environment. It is a disruptive test and
        cannot be run as a health monitoring test.

DiagMemoryTest :
        This test runs the exhaustive ASIC memory test during normal switch operation
        NG3K utilizes mbist for this test. Memory test is very disruptive
        in nature and requires switch reboot after the test.

Controller#
```

This example shows how to display the boot up level:

```
Controller# show diagnostic bootup level
Current bootup diagnostic level: minimal
Controller#
```

Additional References for Online Diagnostics

Related Documents

Related Topic	Document Title
Online diagnostics commands	
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

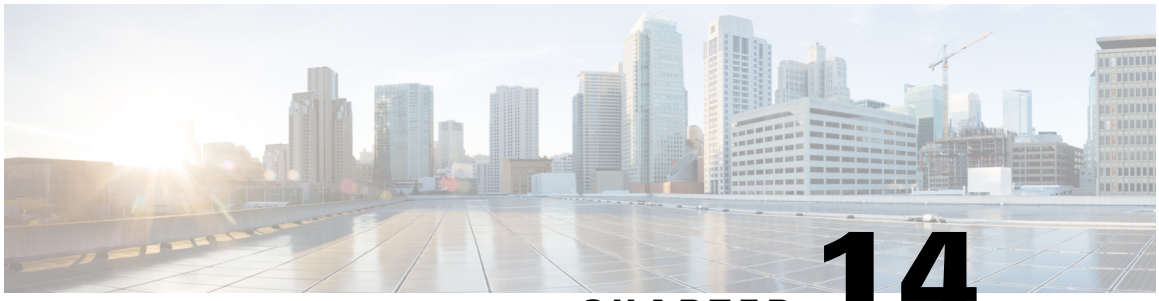
MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

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
	This feature was introduced.



CHAPTER 14

Predownloading an Image to Access Points

This module contains the following topics:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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.

Predownloading an Image to an Access Point

To minimize network outages, you can now download an upgrade image to the access point from the controller without resetting the access point or losing network connectivity. Previously, you would download an upgrade image to the controller and reset it, which causes the access point to go into discovery mode. After the access point discovers the controller with the new image, the access point downloads the new image, resets, goes into discovery mode, and rejoins the controller.

You can now download the upgrade image to the controller and then download the image to the access point while the network is still up. When both devices are up, the access point discovers and rejoins the controller.

Restrictions for Predownloading an Image to an Access Point

The following are the restrictions for predownloading an image to an access point:

- The maximum number of concurrent predownloads is limited to half the number of concurrent normal image downloads. This limitation allows new access points to join the controller during image downloading.

  If you reach the predownload limit, then the access points that cannot get an image sleep for a time between 180 to 600 seconds and then reattempt the predownload.

- Access points with 16-MB total available memory may not have enough free memory to download an upgrade image and may automatically delete crash info files, radio files, and any backup images to free up space. However, this limitation does not affect the predownload process because the predownload image replaces any backup image on the access point.
- All the primary, secondary, and tertiary controllers should run the same images or the feature will not be effective.
- At the time of reset, you have to make sure that all the access points should have downloaded the image.
- The access point can store only two software images.

How to predownload an Image to an Access Point

Predownloading an Image to Access Points (CLI)

Before You Begin

There are some prerequisites you must keep in mind while predownloading an image to access point:

- Predownloading can be done only when the controller is booted in the install mode.
- You can copy the new image either from the TFTP server, flash image, or USB.
- Before predownloading the new image, you must install the new software using the **software install** command and select **no** to the **reload** option.

SUMMARY STEPS

1. **ap image predownload** or **ap** *ap-name* **predownload**
2. **show ap image**
3. **ap image swap**
4. **ap image reset**
5. **reload**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**ap image predownload** or **ap** *ap-name* **predownload**
	
	Example:
	`Controller# ap image predownload`
	`Controller# ap ap1 predownload`
		Downloads new image to all access points or a specific access point connected to the controller.

Step 2	**show ap image** **Example:** `Controller# show ap image`	Verifies the access points predownload process.
Step 3	**ap image swap** **Example:** `Controller# ap image swap`	Specifies to swap the backup image to primary image.
Step 4	**ap image reset** **Example:** `Controller# ap image reset`	Resets the access points.
Step 5	**reload** **Example:** `Controller# reload`	Resets the system.

Monitoring Access Point Predownload Process

This section describes the commands for monitoring the access point predownload process.

The following command can be used to monitor the access point predownload process.

Table 22: Monitoring Access Point Predownload Process Commands

Command	Purpose
show ap image	Verifies the access points predownload process.

Displaying Access Point Predownload Status

While downloading the access point predownload image, you can simultaneously enter the **show ap image** command to verify the predownload progress on the access points:

```
Controller# show ap image
Total number of APs  : 1

Number of APs
```

```
       Initiated                 : 1
       Predownloading            : 1
       Completed predownloading  : 0
       Not Supported             : 0
       Failed to Predownload     : 0

AP Name                          Primary Image   Backup Image    Predownload Status
Predownload Ver...  Next Retry Time    Retry Count
-----------------------------------------------------------------------------------
AP1                              10.0.1.66       10.0.1.66       Predownloading
10.0.1.67           NA                          0


Controller# show ap image

Total number of APs  : 1

Number of APs
       Initiated                 : 1
       Predownloading            : 0
       Completed predownloading  : 1
       Not Supported             : 0
       Failed to Predownload     : 0

AP Name                          Primary Image   Backup Image    Predownload Status
Predownload Ver...  Next Retry Time    Retry Count
-----------------------------------------------------------------------------------
AP1                              10.0.1.66       10.0.1.67       Complete
10.0.1.67           NA                          0
```

Examples: Access Point Predownload Process

This example shows how to predownload an image to an access point AP1:

```
Controller# ap image predownload
Controller# show ap image
Controller# ap image swap
Controller# show ap image
Controller# ap image reset
Controller# reload
```

Additional References for Predownloading an Image to an Access Point

Related Documents

Related Topic	Document Title
Lightweight access points configuration	*Lightweight Access Point Configuration Guide, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*
Lightweight Access Point commands	*Lightweight Access Point Command Reference, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

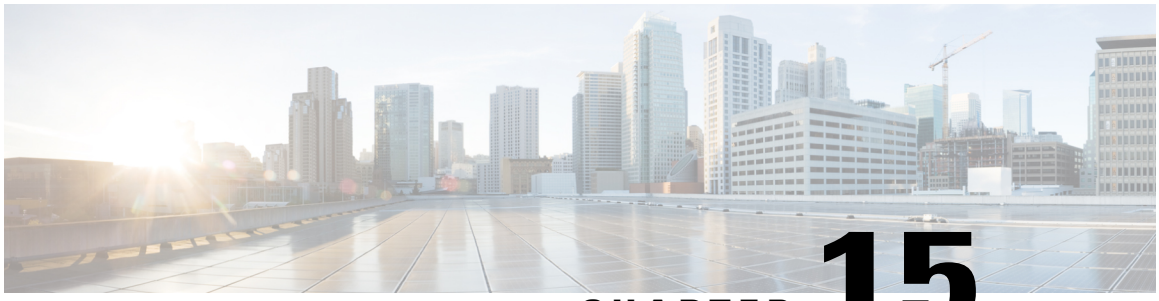
MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

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 Predownloading an Image to an Access Point

Release	Modification
Cisco IOS XE Release 3.2SE	This feature was introduced.



CHAPTER 15

Troubleshooting the Software Configuration

This module 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), the device manager, or Network Assistant to identify and solve problems.

Additional troubleshooting information, such as LED descriptions, is provided in the hardware installation guide.

This module contains the following topics:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Troubleshooting the Software Configuration

Review the topics in this section.

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.

Related Topics

Recovering from a Software Failure, on page 191

Lost or Forgotten Password on a Controller

The default configuration for the controller allows an end user with physical access to the controller 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 controller.

Note On these controllers, 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.

Related Topics

Recovering from a Lost or Forgotten Password, on page 193

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:

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

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 sense the real-time power consumption of the device by monitoring and policing the power usage.

Related Topics

Scenarios to Troubleshoot Power over Ethernet (PoE), on page 206

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 is 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.

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.

Related Topics

Executing Ping, on page 200

Pinging an IP Host: Example, on page 208

Layer 2 Traceroute

The Layer 2 traceroute feature allows the switch to identify the physical path that a packet takes from a source device to a destination device. Layer 2 traceroute supports only unicast source and destination MAC addresses. It finds the path by using the MAC address tables of the switches in the path. When the switch detects a device in the path that does not support Layer 2 traceroute, the switch continues to send Layer 2 trace queries and lets them time out.

The switch can only identify the path from the source device to the destination device. It cannot identify the path that a packet takes from source host to the source device or from the destination device to the destination host.

Layer 2 Traceroute 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 switches in the physical path must be reachable from each other.
- The maximum number of hops identified in the path is ten.
- You can enter the **traceroute mac** or the **traceroute mac ip** privileged EXEC command on a switch that is not in the physical path from the source device to the destination device. All switches in the path must be reachable from this switch.
- The **traceroute mac** command output shows the Layer 2 path only when the specified source and destination MAC addresses belong to the same VLAN. If you specify source and destination MAC addresses that belong to different VLANs, the Layer 2 path is not identified, and an error message appears.
- If you specify a multicast source or destination MAC address, the path is not identified, and an error message appears.
- If the source or destination MAC address belongs to multiple VLANs, you must specify the VLAN to which both the source and destination MAC addresses belong. If the VLAN is not specified, the path is not identified, and an error message appears.
- The **traceroute mac ip** command output shows the Layer 2 path when the specified source and destination IP addresses belong to the same subnet. When you specify the IP addresses, the switch uses the Address Resolution Protocol (ARP) to associate the IP addresses with the corresponding MAC addresses and the VLAN IDs.
  - If an ARP entry exists for the specified IP address, the switch uses the associated MAC address and identifies the physical path.
  - If an ARP entry does not exist, the switch sends an ARP query and tries to resolve the IP address. If the IP address is not resolved, the path is not identified, and an error message appears.
- When multiple devices are attached to one port through hubs (for example, multiple CDP neighbors are detected on a port), the Layer 2 traceroute feature is not supported. When more than one CDP neighbor is detected on a port, the Layer 2 path is not identified, and an error message appears.
- 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 switches can participate as the source or destination of the **traceroute** privileged EXEC command and might or might not appear as a hop in the **traceroute** command output. If the switch is the destination of the traceroute, it is displayed as the final destination in the traceroute output. Intermediate switches do not show

up in the traceroute output if they are only bridging the packet from one port to another within the same VLAN. However, if the intermediate switch is a multilayer switch that is routing a particular packet, this switch shows up as a hop in the traceroute output.

The **traceroute** privileged EXEC command uses the Time To Live (TTL) field in the IP header to cause routers and servers to generate specific return messages. Traceroute starts by sending a User Datagram Protocol (UDP) datagram to the destination host with the TTL field set to 1. If a router finds a TTL value of 1 or 0, it drops the datagram and sends an Internet Control Message Protocol (ICMP) time-to-live-exceeded message to the sender. Traceroute finds the address of the first hop by examining the source address field of the ICMP time-to-live-exceeded message.

To identify the next hop, traceroute sends a UDP packet with a TTL value of 2. The first router decrements the TTL field by 1 and sends the datagram to the next router. The second router sees a TTL value of 1, discards the datagram, and returns the time-to-live-exceeded message to the source. This process continues until the TTL is incremented to a value large enough for the datagram to reach the destination host (or until the maximum TTL is reached).

To learn when a datagram reaches its destination, traceroute sets the UDP destination port number in the datagram to a very large value that the destination host is unlikely to be using. When a host receives a datagram destined to itself containing a destination port number that is unused locally, it sends an ICMP *port-unreachable* error to the source. Because all errors except port-unreachable errors come from intermediate hops, the receipt of a port-unreachable error means that this message was sent by the destination port.

Related Topics

Executing IP Traceroute, on page 201

Performing a Traceroute to an IP Host: Example, on page 209

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.

Related Topics

Redirecting Debug and Error Message Output, on page 202

Enabling All System Diagnostics: Example, on page 210

Crashinfo Files

The crashinfo files save information that helps Cisco technical support representatives to debug problems that caused the Cisco IOS image to fail (crash). The switch generates two files at the time of the failure: full core and crashinfo.

The information in the crashinfo file includes the Cisco IOS image name and version that failed, a list of the processor registers, and a stack trace. You can provide this information to the Cisco technical support representative by using the **show tech-support** privileged EXEC command.

The file names have the following format:

```
[fullcore | crashinfo]_[process that crashed]_[date]-[timestamp]-UTC
```

From IOS, you can view the crashinfo files on each switch by using the following command:

```
Controller#  dir crashinfo?
crashinfo-1: crashinfo-2: crashinfo-3: crashinfo:
Controller#
```

For example, to access the crashinfo directory for switch 1, enter

```
Controller dir crashinfo-1
```

From the ROMMON prompt, you can view the crashinfo files by using the **dir** command:

```
Controller: dir sda1
```

The following is sample output of a crashinfo file

```
Controller# dir crashinfo:

Directory of crashinfo:/

   12  -rwx        2768  Dec 31 1969 16:00:15 -08:00  koops.dat
   15  -rwx           0  Jan 12 2000 22:53:40 -08:00  deleted_crash_files
   16  -rwx     4246576  Jan 12 2000 22:53:40 -08:00  crashinfo_stack-mgr_20000113-065250-UTC

   17  -rwx          50   Oct 2 2012 03:18:42 -08:00  last_crashinfo
   26  -rwx          39  Jan 22 2013 14:14:14 -08:00  last_systemreport
   18  -rwx     2866565  Jan 12 2000 22:53:41 -08:00  fullcore_stack-mgr_20000113-065250-UTC

   20  -rwx     4391796   Feb 1 2000 17:50:44 -08:00  crashinfo_stack-mgr_20000202-014954-UTC

   21  -rwx     2920325   Feb 1 2000 17:50:45 -08:00  fullcore_stack-mgr_20000202-014954-UTC
34817  -rw-     1050209  Jan 10 2013 20:26:23 -08:00  system-report_1_20130111-042535-UTC.gz
18434  -rw-     1016913  Jan 11 2013 10:35:28 -08:00  system-report_1_20130111-183440-UTC.gz
18435  -rw-     1136167  Jan 22 2013 14:14:11 -08:00  system-report_1_20130122-221322-UTC.gz
34821  -rw-     1094631   Jan 2 2013 17:59:23 -08:00  system-report_1_20130103-015835-UTC.gz

 6147  -rw-      967429   Jan 3 2013 10:32:44 -08:00  system-report_1_20130103-183156-UTC.gz
34824  -rwx          50  Jan 22 2013 14:14:14 -08:00  deleted_sysreport_files
6155  -rwx         373  Jan 22 2013 14:14:13 -08:00  last_systemreport_log

145898496 bytes total (18569216 bytes free)
stack3#

The file name of the most recent crashinfo file is stored in last_crashinfo.
The file name of the most recent system report is stored in last_systemreport.


Controller#
```

System Reports

When a controller reloads or crashes, a system report is automatically generated for each switch in the switch stack. The system report file captures all the trace buffers, and other system-wide logs found on the switch. System reports are located in the crashinfo directory in the following format:
`system-report_[switch number]_[date]-[timestamp]-UTC.gz`

After a switch reload or crash, you should check if a system report file was generated. The name of the most recently generated system report file is stored in the last_systemreport file under the crashinfo directory. The system report and crashinfo files assist TAC when troubleshooting your issue.

Onboard Failure Logging on the Switch

You can use the on-board-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 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 stack member
- Power over Ethernet (PoE)—Record of the power consumption of PoE ports on a standalone switch or a stack member
- Temperature—Temperature of a standalone switch or a stack member
- Uptime data—Time when a standalone switch or a 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 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.

Related Topics

Configuring OBFL, on page 203

Displaying OBFL Information, on page 203

Fan Failures

By default, the feature is disabled. When more than one of the fans in a field-replaceable unit (FRU) or in a power supply fails, the switch does not shut down, and this error message appears:

```
Multiple fan(FRU/PS) failure detected. System may get overheated. Change fan quickly.
```

The switch might overheat and shut down.

To enable the fan failures feature, enter the **system env fan-fail-action shut** privileged EXEC command. If more than one fan in the switch fails, the switch automatically shuts down, and this error message appears:

```
Faulty (FRU/PS) fans detected, shutting down system!
```

After the first fan shuts down, if the switch detects a second fan failure, the switch waits for 20 seconds before it shuts down.

To restart the switch, it must be power cycled.

Possible Symptoms of High CPU Utilization

Excessive CPU utilization might result in these symptoms, but the symptoms could also result from other causes.

- Spanning tree topology changes
- EtherChannel links brought down due to loss of communication
- Failure to respond to management requests (ICMP ping, SNMP timeouts, slow Telnet or SSH sessions)

- UDLD flapping
- IP SLAs failures because of SLAs responses beyond an acceptable threshold
- DHCP or IEEE 802.1x failures if the switch does not forward or respond to requests

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

Before You Begin

This recovery procedure requires that you have physical access to the switch.

This procedure uses boot loader commands and TFTP to recover from a corrupted or incorrect image file.

SUMMARY STEPS

1. From your PC, download the software image file (*image.bin*) from Cisco.com.
2. Load the software image to your TFTP server.
3. Connect your PC to the switch Ethernet management port.
4. Unplug the switch power cord.
5. Press the **Mode** button, and at the same time, reconnect the power cord to the switch.
6. From the bootloader (ROMMON) prompt, ensure that you can ping your TFTP server.
7. Verify that you have a recovery image in your recovery partition (sda9:).
8. From the bootloader (ROMMON) prompt, initiate the emergency-install feature that assists you in recovering the software image on your switch.

DETAILED STEPS

Step 1 From your PC, download the software image file (*image.bin*) from Cisco.com.

Step 2 Load the software image to your TFTP server.

Step 3 Connect your PC to the switch Ethernet management port.

Step 4 Unplug the switch power cord.

Step 5 Press the **Mode** button, and at the same time, reconnect the power cord to the switch.

Step 6 From the bootloader (ROMMON) prompt, ensure that you can ping your TFTP server.

a) Set the IP address **switch: set IP_ADDR** *ip_address subnet_mask*

Example:

```
switch: set IP_ADDR 192.0.2.123/255.255.255.0
```

b) Set the default router IP address **switch: set DEFAULT_ROUTER** *ip_address*

Example:

```
switch: set DEFAULT_ROUTER 192.0.2.1
```

c) Verify that you can ping the TFTP server **switch: ping** *ip_address_of_TFTP_server*

Example:

```
switch: ping 192.0.2.15
ping 192.0.2.1 with 32 bytes of data...
Host 192.0.2.1 is alive.
switch:
```

Step 7 Verify that you have a recovery image in your recovery partition (sda9:).
This recovery image is required for recovery using the emergency-install feature.

Example:

```
switch: dir sda9:
Directory of sda9:/

    2  drwx  1024       .
    2  drwx  1024       ..
   11  -rw-  18923068   c3850-recovery.bin

36939776 bytes available (20830208 bytes used)
switch:
```

Step 8 From the bootloader (ROMMON) prompt, initiate the emergency-install feature that assists you in recovering the software image on your switch.
WARNING: The emergency install command will erase your entire boot flash!

Example:

```
switch: emergency-install tftp://172.20.249.254/katana/ct5760.renum.bin

The bootflash will be erased during install operation, continue (y/n)?y
Starting emergency recovery (tftp://172.20.249.254/katana/ct5760.renum.bin)...
Loading "sda9:ct5760-recovery.bin"...
Reading full image into memory.....................done
Verifying image digital signature.
Nova Bundle Image
-------------------------------------
Kernel Address    : 0x8b35b598
Kernel Size       : 0x367550/3568976
Initramfs Address : 0x8b6c2ae8
Initramfs Size    : 0xbfe484/12575876
Compression Format: unknown


File "sda9:ct5760-recovery.bin" uncompressed and installed, entry point: 0x8b35b598
Image validated
\ufffd

Initiating Emergency Installation of bundle tftp://172.20.249.254/katana/ct5760.renum.bin
```

```
Downloading bundle tftp://172.20.249.254/katana/ct5760.renum.bin...

Validating bundle tftp://172.20.249.254/katana/ct5760.renum.bin...
Installing bundle tftp://172.20.249.254/katana/ct5760.renum.bin...
Verifying bundle tftp://172.20.249.254/katana/ct5760.renum.bin...
Package ct5760-base.SPA.03.02.00.pkg is Digitally Signed
Package ct5760-drivers.SPA.03.02.00.SE.pkg is Digitally Signed
Package ct5760-infra.SPA.03.02.00.pkg is Digitally Signed
Package ct5760-iosd-ipservicesk9.SPA.150-1.EX.pkg is Digitally Signed
Package ct5760-platform.SPA.03.02.00.SE.pkg is Digitally Signed
Package ct5760-wcm.SPA.10.0.10.48.pkg is Digitally Signed
Preparing flash...
Syncing device...
Emergency Install successful... Rebooting
Restarting system.
Xmodem file system is available.
Base ethernet MAC Address: 20:37:06:4d:64:00
Verifying bootloader digital signature.

The system is not configured to boot automatically.  The
following command will finish loading the operating system
software:

    boot
```

Related Topics

Software Failure on a Switch, on page 184

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.

SUMMARY STEPS

1. Connect a terminal or PC to the switch.
2. Set the line speed on the emulation software to 9600 baud.
3. On a switch, power off the standalone switch or the entire switch stack. On a switch, power off the switch.
4. Reconnect the power cord to the or the . 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.
5. After recovering the password, reload the switch or the .
6. Power on the remaining switches in the stack.

DETAILED STEPS

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. If you are recovering the password for a switch stack, connect to the console port of the or
- Connect a PC to the Ethernet management port. If you are recovering the password for a switch stack, connect to the Ethernet management port of a stack member .

Step 2 Set the line speed on the emulation software to 9600 baud.

Step 3 On a switch, power off the standalone switch or the entire switch stack. On a switch, power off the switch.

Step 4 Reconnect the power cord to the or the . 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:

  ```
  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:

  ```
  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 or the .
On a switch:

```
Switch> reload
Proceed with reload? [confirm] y
```

Step 6 Power on the remaining switches in the stack.

Related Topics

Lost or Forgotten Password on a Controller, on page 184

Procedure with Password Recovery Enabled

If the password-recovery mechanism 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
```

SUMMARY STEPS

1. Initialize the flash file system.
2. If disabled, enable switch password recovery with the following command.
3. Ignore the startup configuration with the following command.
4. Boot the switch with the *packages.conf* file from flash.
5. Terminate the initial configuration dialog by answering **No**.
6. At the switch prompt, enter privileged EXEC mode.
7. Copy the startup configuration to running configuration.
8. Enter global configuration mode and change the **enable** password.
9. Write the running configuration to the startup configuration file.
10. Confirm that manual boot mode is enabled.
11. Reload the controller.
12. Return the Bootloader parameters (previously changed in Steps 2 and 3) to their original values.
13. Boot the controller with the *packages.conf* file from flash.
14. After the controller boots up, disable manual boot on the controller.

DETAILED STEPS

Step 1 Initialize the flash file system.

```
switch: flash_init
```

Step 2 If disabled, enable switch password recovery with the following command.

```
switch: SWITCH_DISABLE_PASSWORD_RECOVERY=0
```

Step 3 Ignore the startup configuration with the following command.

```
switch: SWITCH_IGNORE_STARTUP_CFG=1
```

Step 4 Boot the switch with the *packages.conf* file from flash.

```
switch: boot flash:packages.conf
```

Step 5 Terminate the initial configuration dialog by answering **No**.

```
Would you like to enter the initial configuration dialog? [yes/no]: No
```

Step 6 At the switch prompt, enter privileged EXEC mode.

```
Switch> enable
Switch#
```

Step 7 Copy the startup configuration to running configuration.

```
Controller# copy startup-config running-config 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 8 Enter global configuration mode and change the **enable** password.

```
Controller# configure terminal
Controller(config)#
```

Step 9 Write the running configuration to the startup configuration file.

```
Controller# copy running-config startup-config
```

Step 10 Confirm that manual boot mode is enabled.

```
Controller# show boot

 BOOT variable = flash:packages.conf;
 Manual Boot = yes
 Enable Break = yes
```

Step 11 Reload the controller.

```
Controller# reload
```

Step 12 Return the Bootloader parameters (previously changed in Steps 2 and 3) to their original values.

```
switch: SWITCH_DISABLE_PASSWORD_RECOVERY=1
switch: switch: SWITCH_IGNORE_STARTUP_CFG=0
```

Step 13 Boot the controller with the *packages.conf* file from flash.

```
switch: boot flash:packages.conf
```

Step 14 After the controller boots up, disable manual boot on the controller.

```
Controller(config)# no boot manual
```

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.

SUMMARY STEPS

1. Elect to continue with password recovery and lose the existing configuration:
2. Load any helper files:
3. Display the contents of flash memory:
4. Boot up the system:
5. At the switch prompt, enter privileged EXEC mode:
6. Enter global configuration mode:
7. Change the password:
8. Return to privileged EXEC mode:
9. Write the running configuration to the startup configuration file:
10. You must now reconfigure the switch. If the system administrator has the backup switch and VLAN configuration files available, you should use those.

DETAILED STEPS

Step 1 Elect to continue with password recovery and lose the existing configuration:

```
Would you like to reset the system back to the default configuration (y/n)? Y
```

Step 2 Load any helper files:

```
Switch: load_helper
```

Step 3 Display the contents of flash memory:

```
switch: dir flash:
```

The switch file system appears.

```
Directory of flash:
```

```
13  drwx         192   Mar 01 1993 22:30:48 switch_image
16128000 bytes total (10003456 bytes free)
```

Step 4 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 5 At the switch prompt, enter privileged EXEC mode:

```
Switch> enable
```

Step 6 Enter global configuration mode:

```
Controller# configure terminal
```

Step 7 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 8 Return to privileged EXEC mode:

```
Switch (config)# exit
Controller#
```

Note Before continuing to Step 9, power on any connected stack members and wait until they have completely initialized.

Step 9 Write the running configuration to the startup configuration file:

```
Controller# 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 re-enable 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 10 You must now reconfigure the switch. If the system administrator has the backup switch and VLAN configuration files available, you should use those.

Preventing Switch Stack Problems

Note

- Make sure that the switches 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.
- After adding or removing stack members, make sure that the switch stack is operating at full bandwidth (32 Gb/s). 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 . 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 switches in the stack can make it easier to remotely troubleshoot the switch stack. However, you need to remember that the switches have manually assigned numbers if you add, remove, or rearrange switches 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:

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 switches.

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.

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 re-insert 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.

Executing Ping

If you attempt to ping a host in a different IP subnetwork, you must define a static route to the network or have IP routing configured to route between those subnets.

IP routing is disabled by default on all switches.

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* `Controller# ping 172.20.52.3`	Pings a remote host through IP or by supplying the hostname or network address.

Related Topics

Ping, on page 185
Pinging an IP Host: Example, on page 208

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.

Monitoring the Physical Path

You can monitor the physical path that a packet takes from a source device to a destination device.

Table 23: Monitoring the Physical Path

Command	Purpose
tracetroute 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.
tracetroute 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* `Controller# traceroute ip 192.51.100.1`	Traces the path that packets take through the network.

Related Topics

IP Traceroute , on page 186
Performing a Traceroute to an IP Host: Example, on page 209

Running TDR and Displaying the Results

When you run TDR on an interface, you can run it on the or a stack member.

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. Logging messages to the console produces very high overhead, whereas logging messages to a virtual terminal produces less overhead. Logging messages to a syslog server produces even less, and logging to an internal buffer produces the least overhead of any method.

Related Topics

Debug Commands, on page 188

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. On switches, the switch number is always 1. 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 .

Related Topics

Onboard Failure Logging on the Switch, on page 189
Displaying OBFL Information, on page 203

Verifying Troubleshooting Software Configuration

Displaying OBFL Information

Table 24: Commands for Displaying OBFL Information

Command	Purpose
show onboard switch *switch-number* **clilog** `Controller# show onboard switch 1 clilog`	Displays the OBFL CLI commands that were entered on a standalone switch or the specified stack members.
show onboard switch *switch-number* **environment** `Controller# show onboard switch 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.

Command	Purpose
show onboard switch *switch-number* **message** `Controller# show onboard switch 1 message`	Displays the hardware-related messages generated by a standalone switch or the specified stack members.
show onboard switch *switch-number* **counter** `Controller# show onboard switch 1 counter`	Displays the counter information on a standalone switch or the specified stack members.
show onboard switch *switch-number* **temperature** `Controller# show onboard switch 1 temperature`	Displays the temperature of a standalone switch or the specified switch stack members.
show onboard switch *switch-number* **uptime** `Controller# show onboard switch 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 specified stack members have been running since they last restarted.
show onboard switch *switch-number* **voltage** `Controller# show onboard switch 1 voltage`	Displays the system voltages of a standalone switch or the specified stack members.
show onboard switch *switch-number* **status** `Controller# show onboard switch 1 status`	Displays the status of a standalone switch or the specified stack members.

Related Topics

Onboard Failure Logging on the Switch, on page 189
Configuring OBFL, on page 203

Verifying the Problem and Cause for High CPU Utilization: Example

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.

```
Controller# 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 25: 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 26: Power Over Ethernet Troubleshooting Scenarios

Symptom or problem	Possible cause and solution
No PoE on only one port. 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**, **show interface status**, or **show power inline detail** 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** and **show inline power detail** commands 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 re-enable 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.

Related Topics

Power over Ethernet Ports, on page 184

Configuration Examples for Troubleshooting Software

Pinging an IP Host: Example

This example shows how to ping an IP host:

```
Controller# 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
Controller#
```

Table 27: 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.

Related Topics

Ping, on page 185
Executing Ping, on page 200

Performing a Traceroute to an IP Host: Example

This example shows how to perform a **traceroute** to an IP host:

```
Controller# 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 28: 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.
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.

Related Topics

IP Traceroute , on page 186
Executing IP Traceroute, on page 201

Enabling All System Diagnostics: Example

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:

```
Controller# 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.

Related Topics

Debug Commands, on page 188

Additional References for Troubleshooting Software Configuration

Related Documents

Related Topic	Document Title
Platform-independent command reference	*Cisco IOS Configuration Fundamentals Command Reference, Cisco IOS XE Release 3S (Catalyst 3850 Switches)*
Platform_independent configuration information	*Configuration Fundamentals Configuration Guide, Cisco IOS XE Release 3S (Catalyst 3850 Switches)*

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

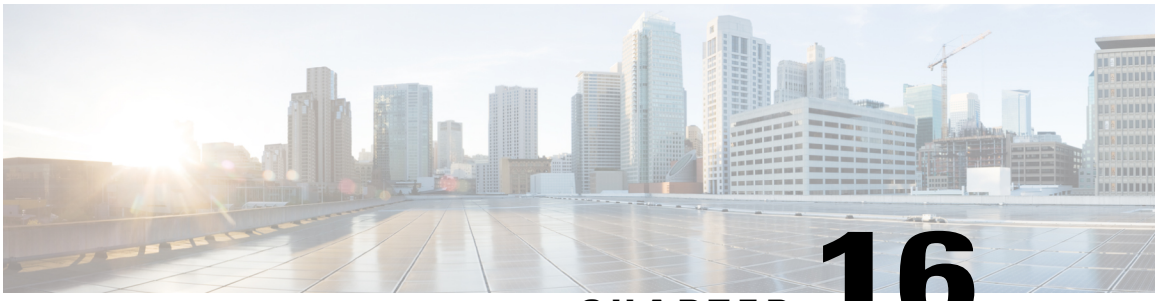
Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, 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
	This feature was introduced.

PART II

QoS



CHAPTER 16

Configuring QoS

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 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 wired or wireless network.
- Traffic characteristics and needs of your wired or wireless 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 wired or wireless network.
- Location of congestion points in the wired or wireless network.

Restrictions for QoS on Wired Targets

A target is an entity where a policy is applied. You can apply a policy to either a wired or wireless target. A wired target can be either a port, client, or VLAN. A wireless target can be either a port, SSID, client, or radio. Wireless QoS policies for port, SSID, client, and radio are applied in the downstream direction. That is, when traffic is flowing from the controller to wireless client.

Note For information about the restrictions for QoS on wireless targets, see Restrictions for Wireless QoS, on page 296.

The following are restrictions for applying QoS features on the controller the wired target:

- A maximum of 8 queuing classes are supported on the controller port for the wired target.
- A maximum of 63 policers are supported per policy on the wired port for the wired target.
- No more than two levels are supported in a QoS hierarchy.
- In a hierarchical policy, overlapping actions between parent and child are not allowed, except for the case where a policy has the port-shaper in the parent and queueing features in the child policy.
- Policing in both the parent and child is not supported in a QoS hierarchy.
- Marking in both the parent and child is not supported in a QoS hierarchy.
- A mixture of queue-limit and queue-buffer in the same policy is not supported.

  Note The queue-limit percent is not supported on the controller because the queue-buffer command handles this functionality. Queue-limit is only supported with the DSCP and CoS extensions.

- The classification sequence for all wired queuing-based policies should be the same across all wired upstream ports (10-Gigabit Ethernet), and the same for all downstream wired ports (1-Gigabit Ethernet).
- Empty classes are not supported.
- The **match vlan** class-map configuration command supports either a VLAN range or a single VLAN.
- The actions under a policer within a policy-map have the following restrictions:
  - The conform action must be transmit.
  - The exceed/violate action for markdown type can only be cos2cos, prec2prec, dscp2dscp.
  - The markdown types must be the same within a policy.
- Table maps have the following specific restrictions:
  - Only one table map per direction per target is supported.
  - Table maps must be configured under the class-default, table maps are unsupported for a user-defined class.

- Hierarchical policies are required for the following:
  - Port-shapers
  - Aggregate policers
  - PV policy
  - Parent queueing with child marking

Note Any "on-the-fly" change (modifying a policy while it is attached to a target) is not permitted for wired hierarchical policies.

- For ports with wired targets, these are the only supported hierarchical policies:
  - Police chaining in the same policy is unsupported, except for wireless client.
  - Hierarchical queueing is unsupported in the same policy (port shaper is the exception).
  - In a parent class, all filters must have the same type. The child filter type must match the parent filter type with the following exceptions:
    - If the parent class is configured to match IP, then the child class can be configured to match the ACL.
    - If the parent class is configured to match CoS, then the child class can be configured to match the ACL.
- Modification to a policy *name* on a wired port is not recommended, please remove and reapply the policies if any unexpected behavior is seen.

The following are restrictions for applying QoS features on the VLAN to the wired target:

- For a flat or nonhierarchical policy, only marking or a table map is supported.

The following are restrictions for applying QoS features on the EtherChannel member links to the wired target:

- QoS is unsupported.

Related Topics

Restrictions for Wireless QoS, on page 296

Information About QoS

QoS Overview

By configuring the quality of service (QoS), you can provide preferential treatment to specific types of traffic at the expense of other traffic types. Without QoS, the controller offers best-effort service to each packet, regardless of the packet contents or size. The controller sends the packets without any assurance of reliability, delay bounds, or throughput.

The following are specific features provided by QoS:

- Low latency
- Bandwidth guarantee
- Buffering capabilities and dropping disciplines
- Traffic policing
- Enables the changing of the attribute of the frame or packet header
- Relative services

Modular QoS Command-Line Interface

With the controller, QoS features are enabled through the Modular QoS command-line interface (MQC). The MQC is a command-line interface (CLI) structure that allows you to create traffic policies and attach these policies to interfaces. A traffic policy contains a traffic class and one or more QoS features. A traffic class is used to classify traffic, whereas the QoS features in the traffic policy determine how to treat the classified traffic. One of the main goals of MQC is to provide a platform-independent interface for configuring QoS across Cisco platforms.

QoS and IPv6

The controller supports IPv6-based QoS policies. You can configure IPv6 ACLs as a match criteria and then set the traffic class of the packet or police the stream. You can also match on DSCP values (traffic class for IPv6) and perform the action. Unless a policy has an IPv4 ACL or specifically has following match criterion, it also matches IPv6 traffic.

Note The class default always matches IPv6 traffic, in addition to IPv4 and MAC.

Wired Access Features for QoS

The following table describes the supported QoS features for wired access.

Table 29: QoS Wired Access Features

Feature	Description
Supported targets	• Gigabit Ethernet • 10-Gigabit Ethernet • VLAN
Configuration sequence	QoS policy installed using the **service-policy** command.

Feature	Description
Supported number of queues at port level	Up to 8 queues supported on a port. No Approximate Fair Dropping or Discard (AFD) support for wired targets.
Supported classification mechanism	• DSCP • IP precedence • CoS • QoS-group • ACL membership including: ◦ IPv4 ACLs ◦ IPv6 ACLS ◦ MAC ACLs

Hierarchical QoS

The controller supports hierarchical QoS (HQoS). HQoS allows you to perform:

- Hierarchical classification— Traffic classification is based upon other classes.
- Hierarchical policing—The process of having the policing configuration at multiple levels in a hierarchical policy.
- Hierarchical shaping—Shaping can also be configured at multiple levels in the hierarchy.

Note Hierarchical shaping is only supported for the port shaper, where for the parent you only have a configuration for the class default, and the only action for the class default is shaping.

Related Topics

Examples: Hierarchical Classification, on page 284

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

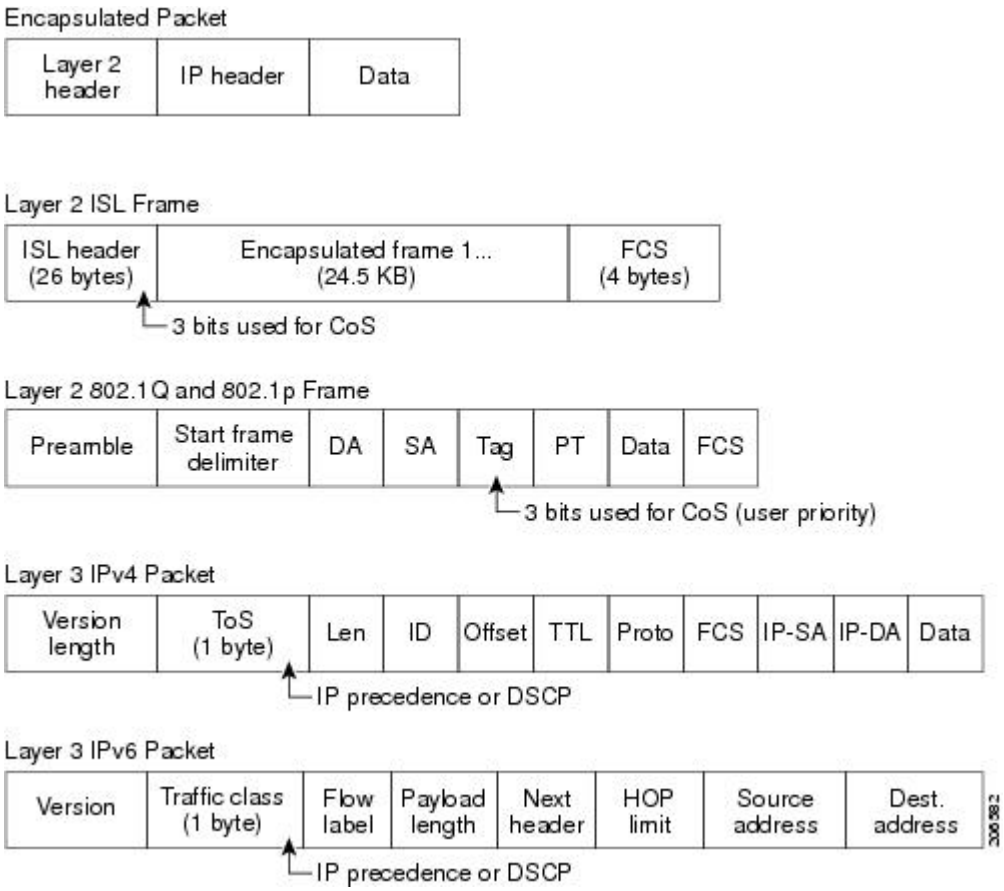
preferential treatment. Implementing QoS in your network makes network performance more predictable and bandwidth utilization more effective.

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.

The special bits in the Layer 2 frame or a Layer 3 packet are shown in the following figure:

Figure 3: QoS Classification Layers in Frames and Packets



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.

Packet Classification

Packet classification is the process of identifying a packet as belonging to one of several classes in a defined policy, based on certain criteria. The Modular QoS CLI (MQC) is a policy-class based language. The policy class language is used to define the following:

- Class-map template with one or several match criteria
- Policy-map template with one or several classes associated to the policy map

The policy map template is then associated to one or several interfaces on the controller.

Packet classification is the process of identifying a packet as belonging to one of the classes defined in the policy map. The process of classification will exit when the packet being processed matches a specific filter in a class. This is referred to as first-match exit. In other words, if a packet matches multiple classes in a policy, irrespective of the order of classes in the policy map, it would still exit the classification process after matching the first class.

If a packet does not match any of the classes in the policy, it would be classified into the default class in the policy. Every policy map has a default class, which is a system defined class to match packets that do not match any of the user-defined classes.

Packet classification can be categorized into the following types:

- Classification based on information that is propagated with the packet
- Classification based on information that is controller specific

- Hierarchical classification

Classification Based on Information That is Propagated with the Packet

Classification that is based on information that is part of the packet and propagated either end-to-end or between hops, typically includes the following:

- Classification based on Layer 3 or 4 headers
- Classification based on Layer 2 information

Classification Based on Layer 3 or Layer 4 Header

This is the most common deployment scenario. Numerous fields in the Layer 3 and Layer 4 headers can be used for packet classification.

At the most granular level, this classification methodology can be used to match an entire flow. For this deployment type, an access control list (ACLs) can be used. ACLs can also be used to match based on various subsets of the flow (for example, source IP address only, or destination IP address only, or a combination of both).

Classification can also be done based on the precedence or DSCP values in the IP header. The IP precedence field is used to indicate the relative priority with which a particular packet needs to be handled. It is made up of three bits in the IP header's type of service (ToS) byte.

The following table shows the different IP Precedence bit values and their names:

Note IP Precedence is not supported for wireless QoS.

Table 30: IP Precedence Values and Names

IP Precedence Value	IP Precedence Bits	IP Precedence Names
0	000	Routine
1	001	Priority
2	010	Immediate
3	011	Flash
4	100	Flash Override
5	101	Critical
6	110	Internetwork control
7	111	Network control

Note All routing control traffic in the network uses IP Precedence value 6 by default. IP Precedence value 7 also is reserved for network control traffic. Therefore, the use of IP Precedence values 6 and 7 is not recommended for user traffic.

The DSCP field is made up of 6 bits in the IP header and is being standardized by the Internet Engineering Task Force (IETF) Differentiated Services Working Group. The original ToS byte contained the DSCP bits has been renamed the DSCP byte. The DSCP field is part of the IP header, similar to IP Precedence. In fact, the DSCP field is a super set of the IP Precedence field. Therefore, the DSCP field is used and is set in ways similar to what was described with respect to IP Precedence.

Note The DSCP field definition is backward-compatible with the IP Precedence values.

Classification Based on Layer 2 Header

A variety of methods can be used to perform classification based on the Layer 2 header information. The most common methods include the following:

- MAC address-based classification (only for access groups)—Classification is based upon the source MAC address (for policies in the input direction) and destination MAC address (for policies in the output direction).
- Class-of-Service—Classification is based on the 3 bits in the Layer 2 header based on the IEEE 802.1p standard. This usually maps to the ToS byte in the IP header.
- VLAN ID—Classification is based on the VLAN ID of the packet.

Note Some of these fields in the Layer 2 header can also be set using a policy.

Classification Based on Information that is Device Specific (QoS Groups)

The controller also provides classification mechanisms that are available where classification is not based on information in the packet header or payload.

At times there may be a requirement to aggregate traffic coming from multiple input interfaces into a specific class in the output interface. For example, there could be multiple customer edge routers going into the same access controller on different interfaces. The service provider might want to police all the aggregate voice traffic going into the core to a specific rate. However, the voice traffic coming in from the different customers could have a different ToS settings. QoS group-based classification is a feature that is useful in these scenarios.

Policies configured on the input interfaces would set the QoS group to a specific value, which can then be used to classify packets in the policy enabled on output interface.

The QoS group is a field in the packet data structure internal to the controller. It is important to note that a QoS group is an internal label to the controller and is not part of the packet header.

Hierarchical Classification

The controller permits you to perform a classification based on other classes. Typically, this action may be required when there is a need to combine the classification mechanisms (that is, filters) from two or more classes into a single class map.

QoS Wired Model

To implement QoS, the controller must perform the following tasks:

- Traffic classification—Distinguishes packets or flows from one another.
- Traffic marking and policing—Assigns a label to indicate the given quality of service as the packets move through the controller, and then make the packets comply with the configured resource usage limits.
- Queuing and scheduling—Provides different treatment in all situations where resource contention exists.
- Shaping—Ensures that traffic sent from the controller meets a specific traffic profile.

Ingress Port Activity

The following activities occur at the ingress port of the controller:

- Classification—Classifying a distinct path for a packet by associating it with a QoS label. For example, the controller maps the CoS or DSCP in the packet to a QoS label to distinguish one type of traffic from another. The QoS label that is generated identifies all future QoS actions to be performed on this packet.
- Policing—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—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, mark down the QoS label in the packet, or drop the packet).

Note Applying polices on the wireless ingress port is not supported on the controller.

Egress Port Activity

The following activities occur at the egress port of the controller:

- Policing—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—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, mark down the QoS label in the packet, or drop the packet).
- Queueing—Queueing evaluates the QoS packet label and the corresponding DSCP or CoS value before selecting which of the egress queues to use. Because congestion can occur when multiple ingress ports simultaneously send data to an egress port, Weighted Tail Drop (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.

Classification

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 enabled on the controller. By default, QoS is enabled on the controller.

During classification, the controller 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.

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.
- 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 controller 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.

Class Maps

A class map is a mechanism that you use to name a specific traffic flow (or class) and 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.

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 controller 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 create a default class by using the **class class-default** policy-map configuration command. The default class is system-defined and cannot be configured. Unclassified traffic (traffic that does not meet the match criteria specified in the traffic classes) is treated as default traffic.

Related Topics

Creating a Traffic Class, on page 241

Examples: Classification by Access Control Lists, on page 283

Policy Maps

A policy map specifies which traffic class to act on. Actions can include the following:

- Setting a specific DSCP or IP precedence value in the traffic class
- Setting a CoS value in the traffic class
- Setting a QoS group
- Setting a wireless LAN (WLAN) value in the traffic class
- 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 and name a policy map using the **policy-map** global configuration command. When you enter this command, the controller enters the policy-map configuration mode. In this mode, you specify the actions to take on a specific traffic class by using the **class** or **set** policy-map configuration and policy-map class configuration commands.

The policy map can also be configured using the **police** and **bandwidth** 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. In addition, the policy-map can further be configured using the **priority** policy-map class configuration command, to schedule priority for the class or the queueing policy-map class configuration commands, **queue-buffers** and **queue-limit**.

To enable the policy map, you attach it to a port by using the **service-policy** interface configuration command.

Related Topics

Creating a Traffic Policy, on page 244

Policy Map on Physical Port

You can configure a nonhierarchical policy map on a physical port that specifies which traffic class to act on. Actions can include setting a specific DSCP or IP precedence value in the traffic class, specifying the traffic bandwidth limitations for each matched traffic class (policer), and taking action 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.

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**).

- A separate policy-map class can exist for each type of traffic received through a port.

Related Topics

Attaching a Traffic Policy to an Interface, on page 253

Policy Map on VLANs

The controller supports a VLAN QoS feature that allows the user to perform QoS treatment at the VLAN level (classification and QoS actions) using the incoming frame's VLAN information. In VLAN-based QoS, a service policy is applied to an SVI interface. All physical interfaces belonging to a VLAN policy map then need to be programmed to refer to the VLAN-based policy maps instead of the port-based policy map.

Although the policy map is applied to the VLAN SVI, any policing (rate-limiting) action can only be performed on a per-port basis. You cannot configure the policer to take account of the sum of traffic from a number of physical ports. Each port needs to have a separate policer governing the traffic coming into that port.

Related Topics

Classifying, Policing, and Marking Traffic on SVIs by Using Policy Maps, on page 258

Examples: Policer VLAN Configuration, on page 289

Policing

After a packet is classified and has a DSCP-based, CoS-based, or QoS-group 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 or CoS value of the packet and allowing the packet to pass through.

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 on a physical port or an SVI.

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.

Related Topics

Configuring Police, on page 269

Examples: Policing Action Configuration, on page 288

Token-Bucket Algorithm

Policing uses a token-bucket algorithm. As each frame is received by the controller, 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 controller 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. You configure how fast (the average rate) that the tokens are removed from the bucket by using the rate option of the **police** policy-map class configuration command.

Related Topics

Configuring Police, on page 269

Examples: Policing Units, on page 289

Marking

Marking is used to convey specific information to a downstream device in the network, or to carry information from one interface in a controller to another.

Marking can be used to set certain field/bits in the packet headers, or marking can also be used to set certain fields in the packet structure that is internal to the controller. Additionally, the marking feature can be used to define mapping between fields. The following marking methods are available for QoS:

- Packet header
- Device (controller) specific information
- Table maps

Packet Header Marking

Marking on fields in the packet header can be classified into two general categories:

- IPv4/v6 header bit marking
- Layer 2 header bit marking

The marking feature at the IP level is used to set the precedence or the DSCP in the IP header to a specific value to get a specific per-hop behavior at the downstream device (switch or router), or it can also be used to aggregate traffic from different input interfaces into a single class in the output interface. The functionality is currently supported on both the IPv4 and IPv6 headers.

Marking in the Layer 2 headers is typically used to influence dropping behavior in the downstream devices (switch or router). It works in tandem with the match on the Layer 2 headers. The bits in the Layer 2 header that can be set using a policy map are class of service.

Switch Specific Information Marking

This form of marking includes marking of fields in the packet data structure that are not part of the packets header, so that the marking can be used later in the data path. This is not propagated between the switches. Marking of QoS-group falls into this category. This form of marking is only supported in policies that are enabled on the input interfaces. The corresponding matching mechanism can be enabled on the output interfaces on the same switch and an appropriate QoS action can be applied.

Table Map Marking

Table map marking enables the mapping and conversion from one field to another using a conversion table. This conversion table is called a table map.

Depending upon the table map attached to an interface, CoS, DSCP, and UP values (UP specific to wireless packets) of the packet are rewritten. The controller allows configuring both ingress table map policies and egress table map policies.

As an example, a table map can be used to map the Layer 2 CoS setting to a precedence value in Layer 3. This feature enables combining multiple **set** commands into a single table, which indicates the method to perform the mapping. This table can be referenced in multiple policies, or multiple times in the same policy.

The following table shows the currently supported forms of mapping:

Table 31: Packet-Marking Types for Which a To-From Relationship Can Be Established

The "To" Packet-Marking Type	The "From" Packet-Marking Type
Precedence	CoS
Precedence	QoS Group
DSCP	CoS
DSCP	QoS Group
CoS	Precedence
CoS	DSCP
QoS Group	Precedence
QoS Group	DSCP

A table map-based policy supports the following capabilities:

- Mutation—You can have a table map that maps from one DSCP value set to another DSCP value set and this can be attached to an egress port.

- Rewrite—Packets coming in are rewritten depending upon the configured table map.
- Mapping—Table map based policies can be used instead of set policies.

The following steps are required for table map marking:

1. Define the table map—Use the **table-map** global configuration command to map the values. The table does not know of the policies or classes within which it will be used. The default command in the table map is used to indicate the value to be copied into the 'to' field when there is no matching 'from' field.
2. Define the policy map—You must define the policy map where the table map will be used.
3. Associate the policy to an interface.

Note A table map policy on an input port changes the trust setting of that port to "from" type of qos-marking.

Related Topics

Configuring Table Maps, on page 261

Examples: Table Map Marking Configuration, on page 291

Traffic Conditioning

To support QoS in a network, traffic entering the service provider network needs to be policed on the network boundary routers to ensure that the traffic rate stays within the service limit. Even if a few routers at the network boundary start sending more traffic than what the network core is provisioned to handle, the increased traffic load leads to network congestion. The degraded performance in the network makes it difficult to deliver QoS for all the network traffic.

Traffic policing functions (using the police feature) and shaping functions (using the traffic shaping feature) manage the traffic rate, but differ in how they treat traffic when tokens are exhausted. The concept of tokens comes from the token bucket scheme, a traffic metering function.

Note When running QoS tests on network traffic, you may see different results for the shaper and policing data. Network traffic data from shaping provides more accurate results.

This table compares the policing and shaping functions.

Table 32: Comparison Between Policing and Shaping Functions

Policing Function	Shaping Function
Sends conforming traffic up to the line rate and allows bursts.	Smooths traffic and sends it out at a constant rate.
When tokens are exhausted, action is taken immediately.	When tokens are exhausted, it buffers packets and sends them out later, when tokens are available. A class with shaping has a queue associated with it which will be used to buffer the packets.

Policing Function	Shaping Function
Policing has multiple units of configuration – in bits per second, packets per second and cells per second.	Shaping has only one unit of configuration - in bits per second.
Policing has multiple possible actions associated with an event, marking and dropping being example of such actions.	Shaping does not have the provision to mark packets that do not meet the profile.
Works for both input and output traffic.	Implemented for output traffic only.
Transmission Control Protocol (TCP) detects the line at line speed but adapts to the configured rate when a packet drop occurs by lowering its window size.	TCP can detect that it has a lower speed line and adapt its retransmission timer accordingly. This results in less scope of retransmissions and is TCP-friendly.

Policing

The QoS policing feature is used to impose a maximum rate on a traffic class. The QoS policing feature can also be used with the priority feature to restrict priority traffic. If the rate is exceeded, then a specific action is taken as soon as the event occurs. The rate (committed information rate [CIR] and peak information rate [PIR]) and the burst parameters (conformed burst size [B_c] and extended burst size [B_e]) are all configured in bytes per second.

The following policing forms or policers are supported for QoS:

- Single-rate two-color policing
- Dual-rate three-color policing

Single-Rate Two-Color Policing

Single-rate two-color policer is the mode in which you configure only a CIR and a B_c.

The B_c is an optional parameter, and if it is not specified it is computed by default. In this mode, when an incoming packet has enough tokens available, the packet is considered to be conforming. If at the time of packet arrival, enough tokens are not available within the bounds of B_c, the packet is considered to have exceeded the configured rate.

Note For information about the token-bucket algorithm, see Token-Bucket Algorithm, on page 228.

Related Topics

Configuring Police, on page 269

Examples: Single-Rate Two-Color Policing Configuration, on page 290

Dual-Rate Three-Color Policing

Within the dual rate policer, there are two possible modes:

- Color-blind mode
- Color-aware mode

In both modes, you configure a committed information rate (CIR) and a peak information rate (PIR). As the name suggests, there are two token buckets in this case, one for the peak rate, and one for the conformed rate.

Note For information about the token-bucket algorithm, see Token-Bucket Algorithm, on page 228.

In the color-blind mode, the incoming packet is first checked against the peak rate bucket. If there are not enough tokens available, the packets is said to violate the rate. If there are enough tokens available, then the tokens in the conformed rate buckets are checked to determine if there are enough tokens available. The tokens in the peak rate bucket are decremented by the size of the packet. If it does not have enough tokens available, the packet is said to have exceeded the configured rate. If there are enough tokens available, then the packet is said to conform, and the tokens in both the buckets are decremented by the size of the packet.

In the color-aware mode, as already mentioned, the incoming packets have already colored into conform, exceed, and violate classes. The algorithm used by the color aware policer as described below for a packet of size B.

1. If the packet belongs to the violate-class, then apply the violate action.
2. Compare the packet size against the number of tokens in the peak rate bucket:
   - If the packet size is more than the number of tokens, then apply the violate action.
   - If the packet size is less than the number of tokens, then go to step 3.
3. If the packet is marked as exceeded, decrement the token in the peak rate bucket by the size of the packet, then apply the exceed action.
4. Compare the packet size against the number of tokens in the conform rate bucket.
   - If the packet size is more than the number of tokens, decrement the token in the peak rate bucket by the size of the packet, then apply the exceed action.
   - If the packet size is less than the number of tokens, then go to step 5.
5. Decrement the number of tokens in the conform bucket by the size of the packet and then apply the conform action.

The rate at which tokens are replenished depends on the packet arrival. Assume that a packet comes in at time T1 and the next one comes in at time T2. The time interval between T1 and T2 determines the number of tokens that need to be added to the token bucket. This is calculated as:

Time interval between packets (T2-T1) * CIR)/8 bytes

Related Topics

Configuring Police, on page 269

Examples: Dual-Rate Three-Color Policing Configuration, on page 290

Shaping

Shaping is the process of imposing a maximum rate of traffic, while regulating the traffic rate in such a way that the downstream switches and routers are not subjected to congestion. Shaping in the most common form is used to limit the traffic sent from a physical or logical interface.

Shaping has a buffer associated with it that ensures that packets which do not have enough tokens are buffered as opposed to being immediately dropped. The number of buffers available to the subset of traffic being shaped is limited and is computed based on a variety of factors. The number of buffers available can also be tuned using specific QoS commands. Packets are buffered as buffers are available, beyond which they are dropped.

Class-based Traffic Shaping

The controller uses class-based traffic shaping. This shaping feature is enabled on a class in a policy that is associated to an interface. A class that has shaping configured is allocated a number of buffers to hold the packets that do not have tokens. The buffered packets are sent out from the class using FIFO. In the most common form of usage, class based shaping is used to impose a maximum rate for an physical interface or logical interface as a whole. The following shaping forms are supported in a class:

- Average rate shaping
- Hierarchical shaping

Shaping is implemented using a token bucket. The values of CIR, B_c and B_e determine the rate at which the packets are sent out and the rate at which the tokens are replenished.

Note For information about the token-bucket algorithm, see Token-Bucket Algorithm, on page 228.

Average Rate Shaping

You use the **shape average** policy-map class command to configure average rate shaping.

This command configures a maximum bandwidth for a particular class. The queue bandwidth is restricted to this value even though the port has more bandwidth available. The controller supports configuring shape average by either a percentage or by a target bit rate value.

Related Topics

Configuring Shaping, on page 279

Examples: Average Rate Shaping Configuration, on page 286

Hierarchical Shaping

Shaping can also be configured at multiple levels in a hierarchy. This is accomplished by creating a parent policy with shaping configured, and then attaching child policies with additional shaping configurations to the parent policy.

There are two supported types of hierarchical shaping:

- Port shaper
- User-configured shaping

The port shaper uses the class default and the only action permitted in the parent is shaping. The queueing action is in the child with the port shaper. With the user configured shaping, you cannot have queueing action in the child.

Related Topics

Configuring Shaping, on page 279

Queueing and Scheduling

The controller uses both queueing and scheduling to help prevent traffic congestion. The controller supports the following queueing and scheduling features:

- Bandwidth
- Weighted Tail Drop
- Priority queues
- Queue buffers

Bandwidth

The controller supports the following bandwidth configurations:

- Bandwidth percent
- Bandwidth remaining ratio

Related Topics

Configuring Bandwidth, on page 267

Bandwidth Percent

You can use the **bandwidth percent** policy-map class command to allocate a minimum bandwidth to a particular class. The total sum cannot exceed 100 percent and in case the total sum is less than 100 percent, then the rest of the bandwidth is divided equally among all bandwidth queues.

Note A queue can oversubscribe bandwidth in case the other queues do not utilize the entire port bandwidth.

You cannot mix bandwidth types on a policy map. For example, you cannot configure bandwidth in a single policy map using both a bandwidth percent and in kilobits per second.

Bandwidth Remaining Ratio

You use the **bandwidth remaining ratio** policy-map class command, to create a ratio for sharing unused bandwidth in specified queues. Any unused bandwidth will be used by these specific queues in the ratio that is specified by the configuration. Use this command when the **priority** command is also used for certain queues in the policy.

When you assign ratios, the queues will be assigned certain weights which are inline with these ratios.

You can specify ratios using a range from 0 to 100. For example you can configure a bandwidth remaining ration of 2 on one class, and another queue with a bandwidth remaining ratio of 4 on another class. The bandwidth remaining ratio of 2 will be scheduled twice as often as the bandwidth remaining ratio of 4.

The total bandwidth ratio allocation for the policy can exceed 100. For example, you can configure a queue with a bandwidth remaining ratio of 50, and another queue with a bandwidth remaining ratio of 100.

Weighted Tail Drop

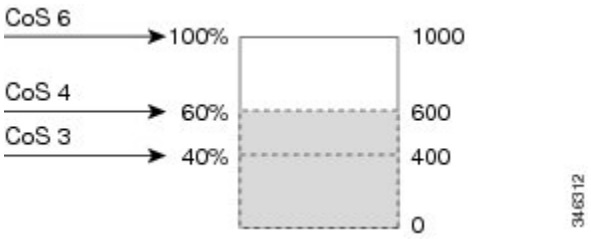
The controller egress queues use an enhanced version of the tail-drop congestion-avoidance mechanism called weighted tail drop (WTD). WTD is implemented on queues to manage the queue lengths and to provide drop precedences for different traffic classifications.

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 controller drops the frame.

Each queue has three configurable threshold values. The QoS label determines which of the three threshold values is subjected to the frame.

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.

Figure 4: WTD and Queue Operation



In the example, CoS value 6 has a greater importance than the other CoS values, and is assigned to the 100-percent drop threshold (queue-full state). CoS values 4 is assigned to the 60-percent threshold, and CoS values 3 is assigned to the 40-percent threshold. All of these threshold values are assigned using the **queue-limit cos** command.

Assuming the queue is already filled with 600 frames, and a new frame arrives. It contains CoS value 4 and is subjected to the 60-percent threshold. If this frame is added to the queue, the threshold will be exceeded, so the controller drops it.

Related Topics

Configuring Queue Limits, on page 276

Examples: Queue-limit Configuration, on page 287

Weighted Tail Drop Default Values

The following are the Weighted Tail Drop (WTD) default values and the rules for configuring WTD threshold values.

- If you configure less than three queue-limit percentages for WTD, then WTD default values are assigned to these thresholds.

  The following are the WTD threshold default values:

Table 33: WTD Threshold Default Values

Threshold	Default Value Percentage
0	80
1	90
2	400

- If 3 different WTD thresholds are configured, then the queues are programmed as configured.
- If 2 WTD thresholds are configured, then the maximum value percentage will be 400.
- If a WTD single threshold is configured as x, then the maximum value percentage will be 400.
  - If the value of x is less than 90, then threshold1=90 and threshold 0= x.
  - If the value of x equals 90, then threshold1=90, threshold 0=80.
  - If the value x is greater than 90, then threshold1=x, threshold 0=80.

Priority Queues

Each port supports eight egress queues, of which two can be given a priority.

You use the **priority level** policy class-map command to configure the priority for two classes. One of the classes has to be configured with a priority queue level 1, and the other class has to be configured with a priority queue level 2. Packets on these two queues are subjected to less latency with respect to other queues.

Related Topics

Configuring Priority, on page 272

Queue Buffer

Each 1-Gigabit port on the controller is allocated 168 buffers. Each 10-Gigabit port is allocated 1800 buffers. At boot time, when there is no policy map enabled on the wired port, there are two queues created by default. Wired ports can have a maximum of 8 queues configured using MQC-based policies. The following table shows which packets go into which one of the queues:

Table 34: DSCP, Precedence, and CoS - Queue Threshold Mapping Table

DSCP, Precedence or CoS	Queue	Threshold
Control Packets	0	1
Rest of Packets	1	2

Note You can guarantee the availability of buffers, set drop thresholds, and configure the maximum memory allocation for a queue. You use the **queue-buffers** policy-map class command to configure the queue buffers. You use the **queue-limit** policy-map class command to configure the maximum thresholds.

There are two types of buffer allocations: hard buffers, which are explicitly reserved for the queue, and soft buffers, which are available for other ports when unused by a given port. By default, Queue 0 will be given 40 percent of the buffers that are available for the interface as hard buffers, that is 67 buffers are allocated for Queue 0 in the context of 1-Gigabit ports, and 720 buffers in the context of 10-Gigabit ports. The soft maximum for this queue is set to 268 (calculated as 67 * 400/100) for 1-Gigabit ports and 2880 for 10-Gigabit ports, where 400 is the default maximum threshold that is configured for any queue.

Queue 1 does not have any hard buffers allocated. The default soft buffer limit is set to 400 (which the maximum threshold). The threshold would determine the maximum number of soft buffers that can be borrowed from the common pool.

Queue Buffer Allocation

The buffer allocation to any queue can be tuned using the **queue-buffers ratio** policy-map class configuration command.

Related Topics

Configuring Queue Buffers, on page 274

Examples: Queue Buffers Configuration, on page 288

Dynamic Threshold and Scaling

Traditionally, reserved buffers are statically allocated for each queue. No matter whether the queue is active or not, its buffers are held up by the queue. In addition, as the number of queues increases, the portion of the reserved buffers allocated for each queue can become smaller and smaller. Eventually, a situation may occur where there are not enough reserved buffers to support a jumbo frame for all queues.

The controller supports Dynamic Thresholding and Scaling (DTS), which is a feature that provides a fair and efficient allocation of buffer resources. When congestion occurs, this DTS mechanism provides an elastic buffer allocation for the incoming data based on the occupancy of the global/port resources. Conceptually, DTS scales down the queue buffer allocation gradually as the resources are used up to leave room for other queues, and vice versa. This flexible method allows the buffers to be more efficiently and fairly utilized.

As mentioned in the previous sections, there are two limits configured on a queue—a hard limit and a soft limit.

Hard limits are not part of DTS. These buffers are available only for that queue. The sum of the hard limits should be less than the globally set up hard maximum limit. The global hard limit configured for egress queuing is currently set to 5705. In the default scenario when there are no MQC policies, configured, the 24 1-Gigabit ports would take up 24 * 67 = 1608, and the 4 10-Gigabit ports would take up 4 * 720 = 2880, for a total of 4488 buffers, allowing room for more hard buffers to be allocated based upon the configuration.

Soft limit buffers participate in the DTS process. Additionally, some of the soft buffer allocations can exceed the global soft limit allocation. The global soft limit allocation for egress queuing is currently set to 7607. The sum of the hard and soft limits add up to 13312, which in turn translates to 3.4 MB. Because the sum of the soft buffer allocations can exceed the global limit, it allows a specific queue to use a large number of buffers when the system is lightly loaded. The DTS process dynamically adjusts the per-queue allocation as the system becomes more heavily loaded.

Trust Behavior

Trust Behavior for Wired Ports

For wired ports that are connected to the controller (end points such as IP phones, laptops, cameras, telepresence units, or other devices), their DSCP, precedence, or CoS values coming in from these end points are trusted by the controller and therefore are retained in the absence of any explicit policy configuration.

The packets are enqueued to the appropriate queue per the default initial configuration.

In scenarios where the incoming packet type differs from the outgoing packet type, the trust behavior and the queuing behavior are explained in the following table. Note that the default trust mode for a wired port is DSCP based. The trust mode 'falls back' to CoS if the incoming packet is a pure Layer 2 packet. You can also change the trust setting from DSCP to CoS. This is accomplished by using an MQC policy that has a class default with a 'set cos cos table default default-cos' action, where default-cos is the name of the table map created (which only performs a default copy).

Table 35: Trust and Queueing Behavior

Incoming Packet	Outgoing Packet	Trust Behavior	Queuing Behavior
Layer 3	Layer 3	Preserve DSCP/Precedence	Based on DSCP
Layer 2	Layer 2	Not applicable	Based on CoS
Tagged	Tagged	Preserve DSCP and CoS	Based on DSCP (trust DSCP takes precedence)
Layer 3	Tagged	Preserve DSCP, CoS is set to 0	Based on DSCP

Port Security on a Trusted Boundary for Cisco IP Phones

In a typical network, you connect a Cisco IP Phone to a controller 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 controller 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 controller should be trusted to ensure that voice traffic is properly prioritized over other types of traffic in the network. By using the **trust device** interface configuration command, you configure the controller port to which the telephone is connected to trust the 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 controller. Without trusted boundary, the CoS labels generated by the PC are trusted by the controller (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 controller port. If the telephone is not detected, the trusted boundary feature disables the trusted setting on the controller 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 controller.

Related Topics

Configuring Trust Behavior for the Device Type, on page 264

Standard QoS Default Settings

Default Wired QoS Configuration

There are two queues configured by default on each wired interface on the controller. All control traffic traverses and is processed through queue 0. All other traffic traverses and is processed through queue 1.

DSCP Maps

All the DSCP maps, except the DSCP-to-DSCP-mutation map, are globally defined and are applied to all ports.

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 36: 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 37: 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 38: 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

Configuring Class, Policy, and Table Maps

Creating a Traffic Class

To create a traffic class containing match criteria, use the **class-map** command to specify the traffic class name, and then use the following **match** commands in class-map configuration mode, as needed.

Before You Begin

All match commands specified in this configuration task are considered optional, but you must configure at least one match criterion for a class.

SUMMARY STEPS

1. **configure terminal**
2. **class-map** {*class-map name* | **match-any** }
3. **match access-group** {*index number* | *name* }
4. **match class-map** *class-map name*
5. **match cos** *cos value*
6. **match dscp** *dscp value*
7. **match ip** {**dscp** *dscp value* | **precedence** *precedence value* }
8. **match non-client-nrt**
9. **match qos-group** *qos group value*
10. **match vlan** *vlan value*
11. **match wlan user-priority** *wlan value*
12. **end**

DETAILED STEPS

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

Step 2	**class-map** {*class-map name*	**match-any** }
		
		Example:
		
		`Controller(config)# class-map test_1000`
		`Controller(config-cmap)#`
			Enters class map configuration mode.
			• Creates a class map to be used for matching packets to the class whose name you specify.
			• If you specify **match-any**, one of the match criteria must be met for traffic entering the traffic class to be classified as part of the traffic class. This is the default.

	Command or Action	Purpose

Step 3	**match access-group** {*index number* | *name* }
	
	Example:
	
	`Controller(config-cmap)# match access-group 100`
	`Controller(config-cmap)#`
		The following parameters are available for this command:
		• access-group
		• class-map
		• cos
		• dscp
		• ip
		• non-client-nrt
		• precedence
		• qos-group
		• vlan
		• wlan user priority
		
		(Optional) For this example, enter the access-group ID:
		• Access list index (value from 1 to 2799)
		• Named access list

Step 4	**match class-map** *class-map name*
	
	Example:
	
	`Controller(config-cmap)# match class-map test_2000`
	`Controller(config-cmap)#`
		(Optional) Matches to another class-map name.

Step 5	**match cos** *cos value*
	
	Example:
	
	`Controller(config-cmap)# match cos 2 3 4 5`
	`Controller(config-cmap)#`
		(Optional) Matches IEEE 802.1Q or ISL class of service (user) priority values:
		• Enters up to 4 CoS values separated by spaces (0 to 7).

Step 6	**match dscp** *dscp value*
	
	Example:
	
	`Controller(config-cmap)# match dscp af11 af12`
	`Controller(config-cmap)#`
		(Optional) Matches the DSCP values in IPv4 and IPv6 packets.

Step 7	**match ip** {**dscp** *dscp value* | **precedence** *precedence value* }	(Optional) Matches IP values including the following: • **dscp—**Matches IP DSCP (DiffServ codepoints).

	Command or Action	Purpose

	Example:
	`Controller(config-cmap)# match ip dscp af11 af12`
	`Controller(config-cmap)#`
		• **precedence**—Matches IP precedence (0 to 7).

Step 8	**match non-client-nrt**
	Example:
	`Controller(config-cmap)# match non-client-nrt`
	`Controller(config-cmap)#`
		(Optional) Matches non-client NRT (Non-Real-Time).
		Note This match is applicable only for policies on a wireless port. It carries all the multi-destination and AP (non-client) bound traffic.

Step 9	**match qos-group** *qos group value*
	Example:
	`Controller(config-cmap)# match qos-group 10`
	`Controller(config-cmap)#`
		(Optional) Matches QoS group value (from 0 to 31).

Step 10	**match vlan** *vlan value*
	Example:
	`Controller(config-cmap)# match vlan 210`
	`Controller(config-cmap)#`
		(Optional) Matches a VLAN ID (from 1 to 4095).

Step 11	**match wlan user-priority** *wlan value*
	Example:
	`Controller(config-cmap)# match wlan user priority 7`
	`Controller(config-cmap)#`
		(Optional) Matches 802.11 specific values. Enter the user priority 802.11 TID user priority (0 to 7).

Step 12	**end** **Example:** `Controller(config-cmap)# end`	Saves the configuration changes.

What to Do Next

Configure the policy map.

Related Topics

Class Maps, on page 225

Examples: Classification by Access Control Lists, on page 283

Creating a Traffic Policy

To create a traffic policy, use the **policy-map** global configuration command to specify the traffic policy name.

The traffic class is associated with the traffic policy when the **class** command is used. The **class** command must be issued after you enter the policy map configuration mode. After entering the **class** command, the controller is automatically in policy map class configuration mode, which is where the QoS policies for the traffic policy are defined.

The following policy map class-actions are supported:

- admit—Admits the request for Call Admission Control (CAC).
- bandwidth—Bandwidth configuration options.
- exit—Exits from the QoS class action configuration mode.
- no—Negates or sets default values for the command.
- police—Policer configuration options.
- priority—Strict scheduling priority configuration options for this class.
- queue-buffers—Queue buffer configuration options.
- queue-limit—Queue maximum threshold for Weighted Tail Drop (WTD) configuration options.
- service-policy—Configures the QoS service policy.
- set—Sets QoS values using the following options:
  - CoS values
  - DSCP values
  - Precedence values
  - QoS group values
  - WLAN values
- shape—Traffic shaping configuration options.

Before You Begin

You should have first created a class-map.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map** *policy-map name*
3. **class** {*class-name* | **class-default** }
4. **admit**
5. **bandwidth** {**kb/s** *kb/s value* | **percent** *percentage* | **remaining** {*percent* | *ratio* }}
6. **exit**
7. **no**
8. **police** {*target_bit_rate* | **cir** | **rate** }
9. **priority** {*kb/s* | **level** *level value* | **percent** *percentage value* }
10. **queue-buffers ratio** *ratio limit*
11. **queue-limit** {*packets* | **cos** | **dscp** | **percent** }
12. **service-policy** *policy-map name*
13. **set** {**cos** | **dscp** | **ip** | **precedence** | **qos-group** | **wlan**}
14. **shape average** {*target _bit_rate* | **percent** }
15. **end**

DETAILED STEPS

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

Step 2	**policy-map** *policy-map name*
	
	Example:
	
	`Controller(config)# policy-map test_2000`
	`Controller(config-pmap)#`
		Enters policy map configuration mode.
		
		Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy.

Step 3	**class** {*class-name*	**class-default** }
		
		Example:
		
		`Controller(config-pmap)# class test_1000`
		`Controller(config-pmap-c)#`
			Specifies the name of the class whose policy you want to create or change.
			
			You can also create a system default class for unclassified packets.

Step 4	**admit** **Example:** `Controller(config-pmap-c)# admit cac`	(Optional) Admits the request for Call Admission Control (CAC). For a more detailed example of this command and its usage, see Configuring Call Admission Control, on page 266. **Note** This command only configures CAC for wireless QoS.

	Command or Action	Purpose
	wmm-tspec Controller(config-pmap-c)#	
Step 5	**bandwidth** {**kb/s** *kb/s value* | **percent** *percentage* | **remaining** {*percent* | *ratio* }} **Example:** Controller(config-pmap-c)# **bandwidth 50** Controller(config-pmap-c)#	(Optional) Sets the bandwidth using one of the following: • **kb/s—**Kilobits per second, enter a value between 20000 and 10000000 for Kb/s. • **percent—**Enter the percentage of the total bandwidth to be used for this policy map. • **remaining—**Enter the percentage ratio of the remaining bandwidth. For a more detailed example of this command and its usage, see Configuring Bandwidth, on page 267.
Step 6	**exit** **Example:** Controller(config-pmap-c)# **exit** Controller(config-pmap-c)#	(Optional) Exits from QoS class action configuration mode.
Step 7	**no** **Example:** Controller(config-pmap-c)# **no** Controller(config-pmap-c)#	(Optional) Negates the command.
Step 8	**police** {*target_bit_rate* | **cir** | **rate** } **Example:** Controller(config-pmap-c)# **police 100000** Controller(config-pmap-c)#	(Optional) Configures the policer: • *target_bit_rate***—**Enter the bit rate per second, enter a value between 8000 and 10000000000. • **cir—**Committed Information Rate • **rate—**Specify police rate, PCR for hierarchical policies or SCR for single-level ATM 4.0 policer policies. For a more detailed example of this command and its usage, see Configuring Police, on page 269.
Step 9	**priority** {*kb/s* | **level** *level value* | **percent** *percentage value* } **Example:** Controller(config-pmap-c)# **priority percent 50** Controller(config-pmap-c)#	(Optional) Sets the strict scheduling priority for this class. Command options include: • *kb/s***—**Kilobits per second, enter a value between 1 and 2000000. • **level—**Establishes a multi-level priority queue. Enter a value (1 or 2). • **percent—**Enter a percent of the total bandwidth for this priority.

	Command or Action	Purpose
		For a more detailed example of this command and its usage, see Configuring Priority, on page 272.

Step 10	**queue-buffers ratio** *ratio limit*
	
	Example:
	
	`Controller(config-pmap-c)# queue-buffers ratio 10`
	`Controller(config-pmap-c)#`
		(Optional) Configures the queue buffer for the class. Enter the queue buffers ratio limit (0 to 100).
		
		For a more detailed example of this command and its usage, see Configuring Queue Buffers, on page 274.

Step 11	**queue-limit** {*packets* | **cos** | **dscp** | **percent** }
	
	Example:
	
	`Controller(config-pmap-c)# queue-limit cos 7 percent 50`
	`Controller(config-pmap-c)#`
		(Optional) Specifies the queue maximum threshold for the tail drop:
		
		• *packets*—Packets by default, enter a value between 1 to 2000000.
		• **cos**—Enter the parameters for each COS value.
		• **dscp**—Enter the parameters for each DSCP value.
		• **percent**—Enter the percentage for the threshold.
		
		For a more detailed example of this command and its usage, see Configuring Queue Limits, on page 276.

Step 12	**service-policy** *policy-map name*
	
	Example:
	
	`Controller(config-pmap-c)# service-policy test_2000`
	`Controller(config-pmap-c)#`
		(Optional) Configures the QoS service policy.

Step 13	**set** {**cos** | **dscp** | **ip** | **precedence** | **qos-group** | **wlan**}
	
	Example:
	
	`Controller(config-pmap-c)# set cos 7`
	`Controller(config-pmap-c)#`
		(Optional) Sets the QoS values. Possible QoS configuration values include:
		
		• **cos**—Sets the IEEE 802.1Q/ISL class of service/user priority.
		• **dscp**—Sets DSCP in IP(v4) and IPv6 packets.
		• **ip**—Sets IP specific values.
		• **precedence**—Sets precedence in IP(v4) and IPv6 packet.
		• **qos-group**—Sets the QoS Group.
		• **wlan**—Sets the WLAN user-priority.

Step 14	**shape average** {*target _bit_rate* | **percent** }
	
	Example:
	
	`Controller(config-pmap-c) #shape average percent 50`
	`Controller(config-pmap-c) #`
		(Optional) Sets the traffic shaping. Command parameters include:
		
		• *target_bit_rate*—Target bit rate.
		• **percent**—Percentage of interface bandwidth for Committed Information Rate.

	Command or Action	Purpose
		For a more detailed example of this command and its usage, see Configuring Shaping, on page 279.

Step 15	**end**
	
	Example:
	
	`Controller(config-pmap-c) #end`
	`Controller(config-pmap-c) #`
		Saves the configuration changes.

What to Do Next

Configure the interface.

Related Topics

Policy Maps, on page 226

Configuring Class-Based Packet Marking

This procedure explains how to configure the following class-based packet marking features on your controller:

- CoS value
- DSCP value
- IP value
- Precedence value
- QoS group value
- WLAN value

Before You Begin

You should have created a class map and a policy map before beginning this procedure.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map** *policy name*
3. **class** *class name*
4. **set cos** { *cos value* | **cos table** *table-map name* | **dscp table** *table-map name* | **precedence table** *table-map name* | **qos-group table** *table-map name* | **wlan user-priority table** *table-map name* }
5. **set dscp** { *dscp value* | **default** | **dscp table** *table-map name* | **ef** | **precedence table** *table-map name* | **qos-group table** *table-map name* | **wlan user-priority table** *table-map name* }
6. **set ip** { **dscp** | **precedence** }
7. **set precedence** { *precedence value* | **cos table** *table-map name* | **dscp table** *table-map name* | **precedence table** *table-map name* | **qos-group table** *table-map name* }
8. **set qos-group** { *qos-group value* | **dscp table** *table-map name* | **precedence table** *table-map name* }
9. **set wlan user-priority** { *wlan user-priority value* | **cos table** *table-map name* | **dscp table** *table-map name* | **qos-group table** *table-map name* | **wlan table** *table-map name* }
10. **end**
11. **show policy-map**

DETAILED STEPS

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

Step 2	**policy-map** *policy name*
	
	Example:
	
	`Controller(config)# policy-map policy1`
	`Controller(config-pmap)#`
		Enters policy map configuration mode.
		
		Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy.

Step 3	**class** *class name*
	
	Example:
	
	`Controller(config-pmap)# class class1`
	`Controller(config-pmap-c)#`
		Enters policy class map configuration mode. Specifies the name of the class whose policy you want to create or change.
		
		Command options for policy class map configuration mode include the following:
		
		• **admit—**Admits the request for Call Admission Control (CAC).
		
		• **bandwidth—**Bandwidth configuration options.
		
		• **exit—**Exits from the QoS class action configuration mode.
		
		• **no—**Negates or sets default values for the command.
		
		• **police—**Policer configuration options.

	Command or Action	Purpose
		• **priority—**Strict scheduling priority configuration options for this class. • **queue-buffers—**Queue buffer configuration options. • **queue-limit—**Queue maximum threshold for Weighted Tail Drop (WTD) configuration options. • **service-policy—**Configures the QoS service policy. • **set—**Sets QoS values using the following options: ◦ CoS values ◦ DSCP values ◦ Precedence values ◦ QoS group values ◦ WLAN values • **shape—**Traffic shaping configuration options. **Note** This procedure describes the available configurations using **set** command options. The other command options (**admit**, **bandwidth**, etc.) are described in other sections of this guide. Although this task lists all of the possible **set** commands, only one **set** command is supported per class.

Step 4	**set cos** { *cos value* | **cos table** *table-map name* | **dscp table** *table-map name* | **precedence table** *table-map name* | **qos-group table** *table-map name* | **wlan user-priority table** *table-map name* }
	Example:
	`Controller(config-pmap)# set cos 5`
	`Controller(config-pmap)#`
		(Optional) Sets the specific IEEE 802.1Q Layer 2 CoS value of an outgoing packet. Values are from 0 to7.
		You can also set the following values using the **set cos** command:
		• **cos table—**Sets the CoS value based on a table map.
		• **dscp table—**Sets the code point value based on a table map.
		• **precedence table—**Sets the code point value based on a table map.
		• **qos-group table—**Sets the CoS value from QoS group based on a table map.
		• **wlan user-priority table—**Sets the CoS value from the WLAN user priority based on a table map.

Step 5	**set dscp** { *dscp value* | **default** | **dscp table** *table-map name* | **ef** | **precedence table** *table-map name* | **qos-group table** *table-map name* | **wlan user-priority table** *table-map name* } **Example:** `Controller(config-pmap)# set dscp`	(Optional) Sets the DSCP value. In addition to setting specific DSCP values, you can also set the following using the **set dscp** command: • **default—**Matches packets with default DSCP value (000000). • **dscp table—**Sets the packet DSCP value from DSCP based on a table map. • **ef—**Matches packets with EF DSCP value (101110).

	Command or Action	Purpose
	af11 Controller(config-pmap)#	• **precedence table—**Sets the packet DSCP value from precedence based on a table map. • **qos-group table—**Sets the packet DSCP value from a QoS group based upon a table map. • **wlan user-priority table—**Sets the packet DSCP value based upon a WLAN user-priority based upon a table map.
Step 6	**set ip** { **dscp** | **precedence** } **Example:** Controller(config-pmap)# **set ip dscp c3** Controller(config-pmap)#	(Optional) Sets IP specific values. These values are either IP DSCP or IP precedence values. You can set the following values using the **set ip dscp** command: • *dscp value***—**Sets a specific dscp value. • **default—**Matches packets with default DSCP value (000000). • **dscp table—**Sets the packet DSCP value from DSCP based on a table map. • **ef—**Matches packets with EF DSCP value (101110). • **precedence table—**Sets the packet DSCP value from precedence based on a table map. • **qos-group table—**Sets the packet DSCP value from a QoS group based upon a table map. • **wlan user-priority table—**Sets the packet DSCP value based upon a WLAN user-priority based upon a table map. You can set the following values using the **set ip precedence** command: • *precedence value***—**Sets the precedence value (from 0 to 7) . • **cos table—**Sets the packet precedence value from Layer 2 CoS based on a table map. • **dscp table—**Sets the packet precedence from DSCP value based on a table map. • **precedence table—**Sets the precedence value from precedence based on a table map • **qos-group table—**Sets the precedence value from a QoS group based upon a table map.
Step 7	**set precedence** { *precedence value* | **cos table** *table-map name* | **dscp table** *table-map name* | **precedence table** *table-map name* | **qos-group table** *table-map name* }	(Optional) Sets precedence values in IPv4 and IPv6 packets. You can set the following values using the **set precedence** command: • *precedence value***—**Sets the precedence value (from 0 to 7) . • **cos table—**Sets the packet precedence value from Layer 2 CoS on a table map.

	Command or Action	Purpose

	Example:
	`Controller(config-pmap)# set precedence 5`
	`Controller(config-pmap)#`
		• **dscp table—**Sets the packet precedence from DSCP value on a table map.
		• **precedence table—**Sets the precedence value from precedence based on a table map.
		• **qos-group table—**Sets the precedence value from a QoS group based upon a table map.

Step 8	**set qos-group** { *qos-group value* | **dscp table** *table-map name* | **precedence table** *table-map name* }
	Example:
	`Controller(config-pmap)# set qos-group 10`
	`Controller(config-pmap)#`
		(Optional) Sets QoS group values. You can set the following values using this command:
		• *qos-group value—*A number from 1 to 31.
		• **dscp table—**Sets the code point value from DSCP based on a table map.
		• **precedence table—**Sets the code point value from precedence based on a table map.

Step 9	**set wlan user-priority** { *wlan user-priority value* | **cos table** *table-map name* | **dscp table** *table-map name* | **qos-group table** *table-map name* | **wlan table** *table-map name* }
	Example:
	`Controller(config-pmap)# set wlan user-priority 1`
	`Controller(config-pmap)#`
		(Optional) Sets the WLAN user priority value. You can set the following values using this command:
		• *wlan user-priority value—*A value between 0 to 7.
		• **cos table—**Sets the WLAN user priority value from CoS based on a table map.
		• **dscp table—**Sets the WLAN user priority value from DSCP based on a table map.
		• **qos-group table—**Sets the WLAN user priority value from QoS group based on a table map.
		• **wlan table—**Sets the WLAN user priority value from the WLAN user priority based on a table map.

Step 10	**end**
	Example:
	`Controller(config-pmap)# end`
	`Controller#`
		Saves configuration changes.

Step 11	**show policy-map** **Example:** `Controller# show policy-map`	(Optional) Displays policy configuration information for all classes configured for all service policies.

What to Do Next

Attach the traffic policy to an interface using the **service-policy** command.

Attaching a Traffic Policy to an Interface

After the traffic class and traffic policy are created, you must use the **service-policy** interface configuration command to attach a traffic policy to an interface, and to specify the direction in which the policy should be applied (either on packets coming into the interface or packets leaving the interface).

Before You Begin

A traffic class and traffic policy must be created before attaching a traffic policy to an interface.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *type*
3. **service-policy** {**input** *policy-map* | **output** *policy-map* }
4. **end**
5. **show policy map**

DETAILED STEPS

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

Step 2	**interface** *type*
	
	Example:
	
	`Controller(config)# interface GigabitEthernet1/0/1`
	`Controller(config-if)#`
		Enters interface configuration mode and configures an interface.
		Command parameters for the interface configuration include:
		• **Auto Template—** Auto-template interface
		• **Capwap—**Capwap tunnel interface
		• **GigabitEthernet—**Gigabit Ethernet IEEE 802
		• **GroupVI—**Group virtual interface
		• **Internal Interface—** Internal interface
		• **Loopback—**Loopback interface
		• **Null—**Null interface
		• **Port-channel—**Ethernet Channel of interface
		• **TenGigabitEthernet—**10-Gigabit Ethernet
		• **Tunnel—**Tunnel interface

	Command or Action	Purpose
		• **Vlan—**Catalyst VLANs • **Range—**interface range

Step 3	**service-policy** {**input** *policy-map* | **output** *policy-map* }
	Example:
	`Controller(config-if)# service-policy output policy_map_01`
	`Controller(config-if)#`
		Attaches a policy map to an input or output interface. This policy map is then used as the service policy for that interface.
		In this example, the traffic policy evaluates all traffic leaving that interface.

Step 4	**end**
	Example:
	`Controller(config-if)# end`
	`Controller#`
		Saves configuration changes.

Step 5	**show policy map** **Example:** `Controller# show policy map`	(Optional) Displays statistics for the policy on the specified interface.

What to Do Next

Proceed to attach any other traffic policy to an interface, and to specify the direction in which the policy should be applied.

Related Topics

Policy Map on Physical Port, on page 226

Classifying, Policing, and Marking Traffic on Physical Ports by Using Policy Maps

You can configure a nonhierarchical policy map on a physical port that specifies which traffic class to act on. Actions supported are remarking and policing.

Before You Begin

You should have already decided upon the classification, policing, and marking of your network traffic by policy maps prior to beginning this procedure.

SUMMARY STEPS

1. **configure terminal**
2. **class-map** {*class-map name* | **match-any** }
3. **match access-group** { *access list index* | *access list name* }
4. **policy-map** *policy-map-name*
5. **class** {*class-map-name* | **class-default**}
6. **set** {**cos** | **dscp** | **ip** | **precedence** | **qos-group** | **wlan user-priority**}
7. **police** {*target_bit_rate* | **cir** | **rate** }
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:** `Controller# configure terminal`	Enters the global configuration mode.

Step 2	**class-map** {*class-map name*	**match-any** }
		
		Example:
		
		`Controller(config)# class-map ipclass1`
		`Controller(config-cmap)# exit`
		`Controller(config)#`
			Enters class map configuration mode.
			• Creates a class map to be used for matching packets to the class whose name you specify.
			• If you specify **match-any**, one of the match criteria must be met for traffic entering the traffic class to be classified as part of the traffic class. This is the default.

Step 3	**match access-group** { *access list index*	*access list name* }
		
		Example:
		
		`Controller(config-cmap)# match access-group 1000`
		`Controller(config-cmap)# exit`
		`Controller(config)#`
			Specifies the classification criteria to match to the class map. You can match on the following criteria:
			• **access-group—**Matches to access group.
			• **class-map—**Matches to another class map.
			• **cos—**Matches to a CoS value.
			• **dscp—**Matches to a DSCP value.
			• **ip—**Matches to a specific IP value.
			• **non-client-nrt—**Matches non-client NRT.

	Command or Action	Purpose
		• **precedence—**Matches precedence in IPv4 and IPv6 packets. • **qos-group—**Matches to a QoS group. • **vlan—**Matches to a VLAN. • **wlan—**Matches to a wireless LAN.

Step 4	**policy-map** *policy-map-name*
	Example:
	`Controller(config)# policy-map flowit`
	`Controller(config-pmap)#`
		Creates a policy map by entering the policy map name, and enters policy-map configuration mode.
		By default, no policy maps are defined.

Step 5	**class** {*class-map-name* | **class-default**}
	Example:
	`Controller(config-pmap)# class ipclass1`
	`Controller(config-pmap-c)#`
		Defines a traffic classification, and enter 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 predefined 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 6	**set** {**cos** | **dscp** | **ip** | **precedence** | **qos-group** | **wlan user-priority**}
	Example:
	`Controller(config-pmap-c)# set dscp 45`
	`Controller(config-pmap-c)#`
		(Optional) Sets the QoS values. Possible QoS configuration values include:
		• **cos—**Sets the IEEE 802.1Q/ISL class of service/user priority.
		• **dscp—**Sets DSCP in IP(v4) and IPv6 packets.
		• **ip—**Sets IP specific values.
		• **precedence—**Sets precedence in IP(v4) and IPv6 packet.
		• **qos-group—**Sets QoS group.
		• **wlan user-priority—**Sets WLAN user priority.
		In this example, the **set dscp** command classifies the IP traffic by setting a new DSCP value in the packet.

Step 7	**police** {*target_bit_rate* | **cir** | **rate** }
	Example:
	`Controller(config-pmap-c)# police 100000 conform-action transmit exceed-action drop`
	`Controller(config-pmap-c)#`
		(Optional) Configures the policer:
		• *target_bit_rate***—**Specifies the bit rate per second, enter a value between 8000 and 10000000000.
		• **cir—**Committed Information Rate.
		• **rate—**Specifies the police rate, PCR for hierarchical policies, or SCR for single-level ATM 4.0 policer policies.

	Command or Action	Purpose
		In this example, the **police** command adds a policer to the class where any traffic beyond the 100000 set target bit rate is dropped.
Step 8	**exit** **Example:** `Controller(config-pmap-c)# exit`	Returns to policy map configuration mode.
Step 9	**exit** **Example:** `Controller(config-pmap)# exit`	Returns to global configuration mode.
Step 10	**interface** *interface-id* **Example:** `Controller(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:** `Controller(config-if)# service-policy input flowit`	Specifies the policy-map name, and applies it to an ingress port. Only one policy map per ingress port is supported.
Step 12	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.
Step 13	**show policy-map** [*policy-map-name* [**class** *class-map-name*]] **Example:** `Controller# show policy-map`	(Optional) Verifies your entries.
Step 14	**copy running-config startup-config** **Example:** `Controller# copy-running-config startup-config`	(Optional) Saves your entries in the configuration file.

What to Do Next

If applicable to your QoS configuration, configure classification, policing, and marking of traffic on SVIs by using policy maps.

Classifying, Policing, and Marking Traffic on SVIs by Using Policy Maps

Before You Begin

You should have already decided upon the classification, policing, and marking of your network traffic by using policy maps prior to beginning this procedure.

SUMMARY STEPS

1. **configure terminal**
2. **class-map** {*class-map name* | **match-any** }
3. **match vlan** *vlan number*
4. **policy-map** *policy-map-name*
5. **description** *description*
6. **class** {*class-map-name* | **class-default**}
7. **set** {**cos** | **dscp** | **ip** | **precedence** | **qos-group** | **wlan user-priority**}
8. **police** {*target_bit_rate* | **cir** | **rate** }
9. **exit**
10. **exit**
11. **interface** *interface-id*
12. **service-policy input** *policy-map-name*
13. **end**
14. **show policy-map** [*policy-map-name* [**class** *class-map-name*]]
15. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**class-map** {*class-map name*	**match-any** } **Example:** `Controller(config)# class-map class_vlan100`	Enters class map configuration mode. • Creates a class map to be used for matching packets to the class whose name you specify. • If you specify **match-any**, one of the match criteria must be met for traffic entering the traffic class to be classified as part of the traffic class. This is the default.

	Command or Action	Purpose

Step 3	**match vlan** *vlan number*
	
	Example:
	
	`Controller(config-cmap)# match vlan 100`
	`Controller(config-cmap)# exit`
	`Controller(config)#`
		Specifies the VLAN to match to the class map.

Step 4	**policy-map** *policy-map-name*
	
	Example:
	
	`Controller(config)# policy-map policy_vlan100`
	`Controller(config-pmap)#`
		Creates a policy map by entering the policy map name, and enters policy-map configuration mode.
		
		By default, no policy maps are defined.

Step 5	**description** *description* **Example:** `Controller(config-pmap)# description vlan 100`	(Optional) Enters a description of the policy map.

Step 6	**class** {*class-map-name* | **class-default**}
	
	Example:
	
	`Controller(config-pmap)# class class_vlan100`
	`Controller(config-pmap-c)#`
		Defines a traffic classification, and enters the 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 predefined 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 7	**set** {**cos** | **dscp** | **ip** | **precedence** | **qos-group** | **wlan user-priority**}
	
	Example:
	
	`Controller(config-pmap-c)# set dscp af23`
	`Controller(config-pmap-c)#`
		(Optional) Sets the QoS values. Possible QoS configuration values include:
		
		• **cos—**Sets the IEEE 802.1Q/ISL class of service/user priority.
		• **dscp—**Sets DSCP in IP(v4) and IPv6 packets.
		• **ip—**Sets IP specific values.
		• **precedence—**Sets precedence in IP(v4) and IPv6 packet.
		• **qos-group—**Sets QoS group.
		• **wlan user-priority—**Sets WLAN user-priority.
		
		In this example, the **set dscp** command classifies the IP traffic by matching the packets with a DSCP value of AF23 (010010).

	Command or Action	Purpose

Step 8	**police** {*target_bit_rate* | **cir** | **rate** }
	
	Example:
	
	`Controller(config-pmap-c)# police 200000 conform-action transmit exceed-action drop`
	`Controller(config-pmap-c)#`
		(Optional) Configures the policer:
		• *target_bit_rate*—Specifies the bit rate per second. Enter a value between 8000 and 10000000000.
		• **cir**—Committed Information Rate.
		• **rate**—Specifies the police rate, PCR for hierarchical policies, or SCR for single-level ATM 4.0 policer policies.
		
		In this example, the **police** command adds a policer to the class where any traffic beyond the 200000 set target bit rate is dropped.

Step 9	**exit** **Example:** `Controller(config-pmap-c)# exit`	Returns to policy map configuration mode.
Step 10	**exit** **Example:** `Controller(config-pmap)# exit`	Returns to global configuration mode.
Step 11	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet 1/0/3`	Specifies the port to attach to the policy map, and enters interface configuration mode. Valid interfaces include physical ports.
Step 12	**service-policy input** *policy-map-name* **Example:** `Controller(config-if)# service-policy input policy_vlan100`	Specifies the policy-map name, and applies it to an ingress port. Only one policy map per ingress port is supported.
Step 13	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.
Step 14	**show policy-map** [*policy-map-name* [**class** *class-map-name*]] **Example:** `Controller# show policy-map`	(Optional) Verifies your entries.

	Command or Action	Purpose
Step 15	**copy running-config startup-config** **Example:** `Controller# copy-running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Policy Map on VLANs, on page 227

Examples: Policer VLAN Configuration, on page 289

Configuring Table Maps

Table maps are a form of marking, and also enable the mapping and conversion of one field to another using a table. For example, a table map can be used to map and convert a Layer 2 CoS setting to a precedence value in Layer 3.

Note A table map can be referenced in multiple policies or multiple times in the same policy.

SUMMARY STEPS

1. **configure terminal**
2. **table-map** *name* { **default** {*default value* | **copy** | **ignore**} | **exit** | **map** {**from** *from value* **to** *to value* } | **no** }
3. **map from** *value* **to** *value*
4. **exit**
5. **exit**
6. **show table-map**
7. **configure terminal**
8. **policy-map**
9. **class class-default**
10. **set cos dscp table** *table map name*
11. **end**

DETAILED STEPS

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

Step 2	**table-map** *name* { **default** {*default value* | **copy** | **ignore**} | **exit** | **map** {**from** *from value* **to** *to value* } | **no** }
	
	Example:
	
	`Controller(config)# table-map table01`
	`Controller(config-tablemap)#`
		Creates a table map and enters the table map configuration mode. In table map configuration mode, you can perform the following tasks:
		• **default—**Configures the table map default value, or sets the default behavior for a value not found in the table map to copy or ignore.
		• **exit—**Exits from the table map configuration mode.
		• **map—**Maps a *from* to a *to* value in the table map.
		• **no—**Negates or sets the default values of the command.

Step 3	**map from** *value* **to** *value*
	
	Example:
	
	`Controller(config-tablemap)# map from 0 to 2`
	`Controller(config-tablemap)# map from 1 to 4`
	`Controller(config-tablemap)# map from 24 to 3`
	`Controller(config-tablemap)# map from 40 to 6`
	`Controller(config-tablemap)# default 0`
	`Controller(config-tablemap)#`
		In this step, packets with DSCP values 0 are marked to the CoS value 2, DSCP value 1 to the CoS value 4, DSCP value 24 to the CoS value 3, DSCP value 40 to the CoS value 6 and all others to the CoS value 0.
		Note The mapping from CoS values to DSCP values in this example is configured by using the **set** policy map class configuration command as described in a later step in this procedure.

Step 4	**exit**
	
	Example:
	
	`Controller(config-tablemap)# exit`
	`Controller(config)#`
		Returns to global configuration mode.

Step 5	**exit**
	
	Example:
	
	`Controller(config) exit`
	`Controller#`
		Returns to privileged EXEC mode.

Step 6	**show table-map**
	
	Example:
	
	`Controller# show table-map`
	`Table Map table01`
		Displays the table map configuration.

	Command or Action	Purpose
	from 0 to 2 from 1 to 4 from 24 to 3 from 40 to 6 default 0	

Step 7	**configure terminal**
	
	Example:
	
	`Controller# configure terminal`
	`Controller(config)#`
		Enters global configuration mode.

Step 8	**policy-map**
	
	Example:
	
	`Controller(config)# policy-map table-policy`
	`Controller(config-pmap)#`
		Configures the policy map for the table map.

Step 9	**class class-default**
	
	Example:
	
	`Controller(config-pmap)# class class-default`
	`Controller(config-pmap-c)#`
		Matches the class to the system default.

Step 10	**set cos dscp table** *table map name*
	
	Example:
	
	`Controller(config-pmap-c)# set cos dscp table table01`
	`Controller(config-pmap-c)#`
		If this policy is applied on input port, that port will have trust dscp enabled on that port and marking will take place depending upon the specified table map.

Step 11	**end**
	
	Example:
	
	`Controller(config-pmap-c)# end`
	`Controller#`
		Returns to privileged EXEC mode.

What to Do Next

Configure any additional policy maps for QoS for your network. After creating your policy maps, attach the traffic policy or polices to an interface using the **service-policy** command.

Related Topics

Table Map Marking, on page 229

Examples: Table Map Marking Configuration, on page 291

Configuring Trust

Configuring Trust Behavior for the Device Type

This procedure explains how to configure trust for one or more device classes within your network configuration.

Before You Begin

There are two types of trust behavior supported on the controller:

- Trust QoS at the policy level. You can configure trust for individual packets by creating specific policy maps and applying them on an interface. If you do not configure a specific policy map, then the default is to trust DSCP.
- Trust devices at the interface level. You can configure trust for the device using the **trust device** interface configuration command.

Note The default mode on an interface is trusted and changes to untrusted only when an untrusted device is detected. In the untrusted mode, the DSCP, IP precedence, or CoS value is reset to 0.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *type*
3. **trust device** { **cisco-phone** | **cts** | **ip-camera** | **media-player** }
4. **end**
5. **show interface status**

DETAILED STEPS

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

Step 2	**interface** *type*
	
	Example:
	
	`Controller(config)# interface GigabitEthernet1/0/1`
	`Controller(config-if)#`
		Enters interface configuration mode and configures an interface.
		Command parameters for the interface configuration include:
		• **Auto Template—** Auto-Template interface
		• **Capwap—**Capwap tunnel interface

	Command or Action	Purpose
		• **GigabitEthernet**—Gigabit Ethernet IEEE 802 • **GroupVI**—Group virtual interface • **Internal Interface**—Internal interface • **Loopback**—Loopback interface • **Null**—Null interface • **Port-channel**—Ethernet Channel of interface • **TenGigabitEthernet**—10-Gigabit Ethernet • **Tunnel**—Tunnel interface • **Vlan**—Catalyst VLANs • **range**—interface range

Step 3	**trust device** { **cisco-phone** | **cts** | **ip-camera** | **media-player** }
	
	Example:
	
	`Controller(config-if)# trust device cisco-phone`
	`Controller(config-if)#`
		Configures the trust value for the interface. You can configure trust for the following supported devices:
		• **cisco-phone**—Cisco IP Phone
		• **cts**—Cisco TelePresence system
		• **ip-camera**—IPVSC
		• **media-player**—DMP

Step 4	**end**
	
	Example:
	
	`Controller(config-if)# end`
	`Controller#`
		Saves configuration changes.

Step 5	**show interface status**
	
	Example:
	
	`Controller# show interface status`
	`Controller#`
		(Optional) Displays the configured interface's status.

What to Do Next

Connect the trusted device to the appropriately configured trusted port on the controller.

Related Topics

Port Security on a Trusted Boundary for Cisco IP Phones, on page 238

Configuring QoS Features and Functionality

Configuring Call Admission Control

This task explains how to configure class-based, unconditional packet marking features on your controller for Call Admission Control (CAC).

CAC is a concept that applies to voice traffic only—not data traffic. If an influx of data traffic oversubscribes a particular link in the network, queueing, buffering, and packet drop decisions resolve the congestion. The extra traffic is simply delayed until the interface becomes available to send the traffic, or, if traffic is dropped, the protocol or the end user initiates a timeout and requests a retransmission of the information.

Network congestion cannot be resolved in this manner when real-time traffic, sensitive to both latency and packet loss, is present, without jeopardizing the quality of service (QoS) expected by the users of that traffic. For real-time delay-sensitive traffic such as voice, it is better to deny network access under congestion conditions than to allow traffic onto the network to be dropped and delayed, causing intermittent impaired QoS and resulting in customer dissatisfaction.

CAC is therefore a deterministic and informed decision that is made before a voice call is established and is based on whether the required network resources are available to provide suitable QoS for the new call.

Before You Begin

You should have created a class map for CAC before beginning this procedure.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map** *policy name*
3. **class** *class name*
4. **admit cac wmm-tspec**
5. **end**
6. **show policy-map**

DETAILED STEPS

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

Step 2	**policy-map** *policy name*
	
	Example:
	
	`Controller(config)# policy-map policy_CAC01`
	`Controller(config-pmap)#`
		Enters policy map configuration mode.
		
		Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy.

	Command or Action	Purpose

Step 3	**class** *class name*
	
	Example:
	
	`Controller(config-pmap)# class class_CAC01`
	`Controller(config-pmap-c)#`
		Enters policy class map configuration mode. Specifies the name of the class whose policy you want to create or change. Command options for policy class map configuration mode include the following:
		• *word*—Class map name.
		• **class-default**—System default class matching any otherwise unclassified packets.

Step 4	**admit cac wmm-tspec**
	
	Example:
	
	`Controller(config-pmap-c)# admit cac wmm-tspec`
	`Controller(config-pmap-c)#`
		Configures call admission control for the policy map.
		Note This command only configures CAC for wireless QoS.

Step 5	**end**
	
	Example:
	
	`Controller(config-pmap)# end`
	`Controller#`
		Saves configuration changes.

Step 6	**show policy-map** **Example:** `Controller# show policy-map`	(Optional) Displays policy configuration information for all classes configured for all service policies.

What to Do Next

Configure any additional policy maps for QoS for your network. After creating your policy maps, attach the traffic policy or polices to an interface using the **service-policy** command.

Configuring Bandwidth

This procedure explains how to configure bandwidth on your controller.

Before You Begin

You should have created a class map for bandwidth before beginning this procedure.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map** *policy name*
3. **class** *class name*
4. **bandwidth** { *Kb/s* | **percent** *percentage* | **remaining** { **ratio** *ratio* } }
5. **end**
6. **show policy-map**

DETAILED STEPS

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

Step 2	**policy-map** *policy name*
	
	Example:
	
	`Controller(config)# policy-map policy_bandwidth01`
	`Controller(config-pmap)#`
		Enters policy map configuration mode.
		
		Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy.

Step 3	**class** *class name*
	
	Example:
	
	`Controller(config-pmap)# class class_bandwidth01`
	`Controller(config-pmap-c)#`
		Enters policy class map configuration mode. Specifies the name of the class whose policy you want to create or change. Command options for policy class map configuration mode include the following:
		
		• *word*—Class map name.
		
		• **class-default**—System default class matching any otherwise unclassified packets.

Step 4	**bandwidth** { *Kb/s* | **percent** *percentage* | **remaining** { **ratio** *ratio* } }
	
	Example:
	
	`Controller(config-pmap-c)# bandwidth 200000`
	`Controller(config-pmap-c)#`
		Configures the bandwidth for the policy map. The parameters include:
		
		• *Kb/s*—Configures a specific value in kilobits per second (from 20000 to 10000000).
		
		• **percent-**—Allocates minimum bandwidth to a particular class based on a percentage. The queue can oversubscribe bandwidth in case other queues do not utilize the entire port bandwidth. The total sum cannot exceed 100 percent, and in case it is less than 100 percent, the rest of the bandwidth is equally divided along all bandwidth queues.
		
		• **remaining—** Allocates minimum bandwidth to a particular class. The queue can oversubscribe bandwidth in case other queues do not utilize entire port bandwidth. The total sum cannot exceed 100 percent. It is preferred to use this command when the **priority** command is used for

	Command or Action	Purpose
		certain queues in the policy. You can also assign ratios rather than percentages to each queue; the queues will be assigned certain weights which are inline with these ratios. Ratios can range from 0 to 100. Total bandwidth ratio allocation for the policy in this case can exceed 100. **Note** You cannot mix bandwidth types on a policy map. For example, you cannot configure bandwidth in a single policy map using both a bandwidth percent and in kilobits per second.

Step 5	**end**
	
	Example:
	
	`Controller(config-pmap-c)# end`
	`Controller#`
		Saves configuration changes.

Step 6	**show policy-map** **Example:** `Controller# show policy-map`	(Optional) Displays policy configuration information for all classes configured for all service policies.

What to Do Next

Configure any additional policy maps for QoS for your network. After creating the policy maps, attach the traffic policy or polices to an interface using the **service-policy** command.

Related Topics

Bandwidth, on page 234

Configuring Police

This procedure explains how to configure policing on your controller.

Before You Begin

You should have created a class map for policing before beginning this procedure.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map** *policy name*
3. **class** *class name*
4. **police** { *target_bit_rate* [*burst bytes* | **bc** | **conform-action** | **pir**] | **cir** { *target_bit_rate* | **percent** *percentage* } | **rate** { *target_bit_rate* | **percent** *percentage* } **conform-action transmit exceed-action** { **drop** [**violate action**] | **set-cos-transmit** | **set-dscp-transmit** | **set-prec-transmit** | **transmit** [**violate action**] } }
5. **end**
6. **show policy-map**

DETAILED STEPS

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

Step 2	**policy-map** *policy name*
	
	Example:
	
	`Controller(config)# policy-map policy_police01`
	`Controller(config-pmap)#`
		Enters policy map configuration mode.
		
		Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy.

Step 3	**class** *class name*
	
	Example:
	
	`Controller(config-pmap)# class class_police01`
	`Controller(config-pmap-c)#`
		Enters policy class map configuration mode. Specifies the name of the class whose policy you want to create or change. Command options for policy class map configuration mode include the following:
		
		• *word*—Class map name.
		
		• **class-default**—System default class matching any otherwise unclassified packets.

Step 4	**police** { *target_bit_rate* [*burst bytes*	**bc**	**conform-action**	**pir**]	**cir** { *target_bit_rate*	**percent** *percentage* }	**rate** { *target_bit_rate*	**percent** *percentage* } **conform-action transmit exceed-action** { **drop** [**violate action**]	**set-cos-transmit**	**set-dscp-transmit**	**set-prec-transmit**	**transmit** [**violate action**] } } **Example:** `Controller(config-pmap-c)# police 8000 conform-action transmit exceed-action`	The following **police** subcommand options are available: • *target_bit_rate*—Bits per second (from 8000 to 10000000000). ◦ *burst bytes*—Enter a value from 1000 to 512000000. ◦ **bc**—Conform burst. ◦ **conform-action**—Action taken when rate is less than conform burst. ◦ **pir**—Peak Information Rate. • **cir**—Committed Information Rate.

	Command or Action	Purpose
	drop Controller(config-pmap-c)#	◦ *target_bit_rate*—Target bit rate (8000 to10000000000). ◦ **percent**—Percentage of interface bandwidth for CIR. • **rate**—Specifies the police rate, PCR for hierarchical policies, or SCR for single-level ATM 4.0 policer policies. ◦ *target_bit_rate*—Target Bit Rate (8000-10000000000). ◦ **percent**—Percentage of interface bandwidth for rate. The following **police conform-action transmit exceed-action** subcommand options are available: • **drop**—Drops the packet. • **set-cos-transmit**—Sets the CoS value and sends it. • **set-dscp-transmit**—Sets the DSCP value and sends it. • **set-prec-transmit**—Rewrites the packet precedence and sends it. • **transmit**—Transmits the packet. **Note** Policer based markdown actions are only supported using table maps. Only one markdown table map is allowed for each marking field in the controller.
Step 5	**end** **Example:** Controller(config-pmap-c)# **end** Controller#	Saves configuration changes.
Step 6	**show policy-map** **Example:** Controller# **show policy-map**	(Optional) Displays policy configuration information for all classes configured for all service policies.

What to Do Next

Configure any additional policy maps for QoS for your network. After creating your policy maps, attach the traffic policy or polices to an interface using the **service-policy** command.

Related Topics

Single-Rate Two-Color Policing, on page 231

Examples: Single-Rate Two-Color Policing Configuration, on page 290

Dual-Rate Three-Color Policing, on page 231
Examples: Dual-Rate Three-Color Policing Configuration, on page 290
Policing, on page 227
Examples: Policing Action Configuration, on page 288
Token-Bucket Algorithm, on page 228
Examples: Policing Units, on page 289

Configuring Priority

This procedure explains how to configure priority on your controller.

The controller supports giving priority to specified queues. There are two priority levels available (1 and 2).

Note Queues supporting voice and video should be assigned a priority level of 1.

Before You Begin

You should have created a class map for priority before beginning this procedure.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map** *policy name*
3. **class** *class name*
4. **priority** [*Kb/s* [*burst_in_bytes*] | **level** *level_value* [*Kb/s* [*burst_in_bytes*] | **percent** *percentage* [*burst_in_bytes*]] | **percent** *percentage* [*burst_in_bytes*]]
5. **end**
6. **show policy-map**

DETAILED STEPS

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

Step 2	**policy-map** *policy name*
	
	Example:
	
	`Controller(config)# policy-map policy_priority01`
	`Controller(config-pmap)#`
		Enters policy map configuration mode.
		
		Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy.

	Command or Action	Purpose

Step 3	**class** *class name*
	
	Example:
	
	`Controller(config-pmap)# class class_priority01`
	`Controller(config-pmap-c)#`
		Enters policy class map configuration mode. Specifies the name of the class whose policy you want to create or change. Command options for policy class map configuration mode include the following:
		• *word*—Class map name.
		• **class-default**—System default class matching any otherwise unclassified packets.

Step 4	**priority** [*Kb/s* [*burst_in_bytes*] | **level** *level_value* [*Kb/s* [*burst_in_bytes*] | **percent** *percentage* [*burst_in_bytes*]] | **percent** *percentage* [*burst_in_bytes*]]
	
	Example:
	
	`Controller(config-pmap-c)# priority level 1`
	`Controller(config-pmap-c)#`
		The **priority** command assigns a strict scheduling priority for the class.
		The command options include:
		• *Kb/s*—Specifies the kilobits per second (from 1 to 2000000).
		◦ *burst_in_bytes*—Specifies the burst in bytes (from 32 to 2000000).
		• **level** *level_value*—Specifies the multilevel (1-2) priority queue.
		◦ *Kb/s*—Specifies the kilobits per second (from 1 to 2000000).
		◦ *burst_in_bytes*—Specifies the burst in bytes (from 32 to 2000000).
		◦ **percent**—Percentage of the total bandwidth.
		◦ *burst_in_bytes*—Specifies the burst in bytes (from 32 to 2000000).
		• **percent**—Percentage of the total bandwidth.
		◦ *burst_in_bytes*—Specifies the burst in bytes (32 to 2000000).
		
		Note Priority level 1 is more important than priority level 2. Priority level 1 reserves bandwidth that is processed first for QoS, so its latency is very low. Both priority level 1 and 2 reserve bandwidth.

Step 5	**end**
	
	Example:
	
	`Controller(config-pmap-c)# end`
	`Controller#`
		Saves configuration changes.

Step 6	**show policy-map** **Example:** `Controller# show policy-map`	(Optional) Displays policy configuration information for all classes configured for all service policies.

What to Do Next

Configure any additional policy maps for QoS for your network. After creating your policy maps, attach the traffic policy or polices to an interface using the **service-policy** command.

Related Topics

Priority Queues, on page 236

Configuring Queues and Shaping

Configuring Egress Queue Characteristics

Depending on the complexity of your network and your QoS solution, you may need to perform all of the procedures in this section. You need to make decisions about these characteristics:

- Which packets are mapped by DSCP, CoS, or QoS group value to each queue and threshold ID?
- What drop percentage thresholds apply to the queues, and how much reserved and maximum memory is needed for the traffic type?
- How much of the fixed buffer space is allocated to the queues?
- 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?

Note You can only configure the egress queues on the controller.

Configuring Queue Buffers

The controller allows you to allocate buffers to queues. If there is no allocation made to buffers, then they are divided equally for all queues. You can use the queue-buffer ratio to divide it in a particular ratio. Since by default DTS (Dynamic Threshold and Scaling) is active on all queues, these are soft buffers.

Note The queue-buffer ratio is supported on both wired and wireless ports, but the queue-buffer ratio cannot be configured with a queue-limit.

Before You Begin

The following are prerequisites for this procedure:

- You should have created a class map for the queue buffer before beginning this procedure.
- You must have configured either bandwidth, shape, or priority on the policy map prior to configuring the queue buffers.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map** *policy name*
3. **class** *class name*
4. **bandwidth** { *Kb/s* | **percent** *percentage* | **remaining** { **ratio** *ratio value* } }
5. **queue-buffers** {**ratio** *ratio value* }
6. **end**
7. **show policy-map**

DETAILED STEPS

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

Step 2	**policy-map** *policy name*
	
	Example:
	
	`Controller(config)# policy-map policy_queuebuffer01`
	`Controller(config-pmap)#`
		Enters policy map configuration mode.
		
		Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy.

Step 3	**class** *class name*
	
	Example:
	
	`Controller(config-pmap)# class class_queuebuffer01`
	`Controller(config-pmap-c)#`
		Enters policy class map configuration mode. Specifies the name of the class whose policy you want to create or change. Command options for policy class map configuration mode include the following:
		
		• *word*—Class map name.
		
		• **class-default**—System default class matching any otherwise unclassified packets.

Step 4	**bandwidth** { *Kb/s* | **percent** *percentage* | **remaining** { **ratio** *ratio value* } }
	
	Example:
	
	`Controller(config-pmap-c)# bandwidth percent 80`
	`Controller(config-pmap-c)#`
		Configures the bandwidth for the policy map. The command parameters include:
		
		• *Kb/s*—Use this command to configure a specific value. The range is 20000 to 10000000.
		
		• **percent**—Allocates a minimum bandwidth to a particular class using a percentage. The queue can oversubscribe bandwidth in case other queues do not utilize the entire port bandwidth. The total sum cannot exceed 100 percent, and in case it is less than 100 percent, the rest of the bandwidth is equally divided along all bandwidth queues.
		
		• **remaining**—Allocates a minimum bandwidth to a particular class. The queue can oversubscribe bandwidth in case other queues do not utilize

	Command or Action	Purpose
		entire port bandwidth. The total sum cannot exceed 100 percent. It is preferred to use this command when the **priority** command is used for certain queues in the policy. You can also assign ratios rather than a percentage to each queue; the queues will be assigned certain weights that are inline with these ratios. Ratios can range from 0 to 100. Total bandwidth ratio allocation for the policy in this case can exceed 100. **Note** You cannot mix bandwidth types on a policy map.

Step 5	**queue-buffers** {**ratio** *ratio value* }
	Example:
	`Controller(config-pmap-c)# queue-buffers ratio 10`
	`Controller(config-pmap-c)#`
		Configures the relative buffer size for the queue.
		Note The sum of all configured buffers in a policy must be less than or equal to 100 percent. Unallocated buffers are are evenly distributed to all the remaining queues.

Step 6	**end**
	Example:
	`Controller(config-pmap-c)# end`
	`Controller#`
		Saves configuration changes.

Step 7	**show policy-map** **Example:** `Controller# show policy-map`	(Optional) Displays policy configuration information for all classes configured for all service policies.

What to Do Next

Configure any additional policy maps for QoS for your network. After creating your policy maps, attach the traffic policy or polices to an interface using the **service-policy** command.

Related Topics

Queue Buffer Allocation, on page 237

Examples: Queue Buffers Configuration, on page 288

Configuring Queue Limits

You use queue limits to configure Weighted Tail Drop (WTD). WTD ensures the configuration of more than one threshold per queue. Each class of service is dropped at a different threshold value to provide for QoS differentiation. With the controller, each queue has 3 explicit programmable threshold classes—0, 1, 2. Therefore, the enqueue/drop decision of each packet per queue is determined by the packet's threshold class assignment, which is determined by the DSCP, CoS, or QoS group field of the frame header.

WTD also uses a soft limit, and therefore you are allowed to configure the queue limit to up to 400 percent (maximum four times the reserved buffer from common pool). This soft limit prevents overrunning the common pool without impacting other features.

Note You can only configure queue limits on the controller egress queues on wired ports.

Before You Begin

The following are prerequisites for this procedure:

- You should have created a class map for the queue limits before beginning this procedure.
- You must have configured either bandwidth, shape, or priority on the policy map prior to configuring the queue limits.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map** *policy name*
3. **class** *class name*
4. **bandwidth** { *Kb/s* | **percent** *percentage* | **remaining** { **ratio** *ratio value* } }
5. **queue-limit** { *packets* **packets** | **cos** {*cos value* { *maximum threshold value* | **percent** *percentage* } | **values** {*cos value* | **percent** *percentage* } } | **dscp** {*dscp value* { *maximum threshold value* | **percent** *percentage* } | *match packet* { *maximum threshold value* | **percent** *percentage* } | **default** { *maximum threshold value* | **percent** *percentage* } | **ef** { *maximum threshold value* | **percent** *percentage* } | **dscp values** *dscp value* } | **percent** *percentage* } }
6. **end**
7. **show policy-map**

DETAILED STEPS

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

Step 2	**policy-map** *policy name*
	
	Example:
	
	`Controller(config)# policy-map policy_queuelimit01`
	`Controller(config-pmap)#`
		Enters policy map configuration mode.
		
		Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy.

	Command or Action	Purpose

Step 3	**class** *class name*
	
	Example:
	
	`Controller(config-pmap)# class class_queuelimit01`
	`Controller(config-pmap-c)#`
		Enters policy class map configuration mode. Specifies the name of the class whose policy you want to create or change. Command options for policy class map configuration mode include the following:
		
		• *word*—Class map name.
		
		• **class-default**—System default class matching any otherwise unclassified packets.

Step 4	**bandwidth** { *Kb/s* | **percent** *percentage* | **remaining** { **ratio** *ratio value* } }
	
	Example:
	
	`Controller(config-pmap-c)# bandwidth 500000`
	`Controller(config-pmap-c)#`
		Configures the bandwidth for the policy map. The parameters include:
		
		• *Kb/s*—Use this command to configure a specific value. The range is 20000 to 10000000
		
		• **percent**—Allocates a minimum bandwidth to a particular class. The queue can oversubscribe bandwidth in case other queues do not utilize the entire port bandwidth. The total sum cannot exceed 100 percent, and in case it is less than 100 percent, the rest of the bandwidth is equally divided along all bandwidth queues.
		
		• **remaining**—Allocates a minimum bandwidth to a particular class. The queue can oversubscribe bandwidth in case other queues do not utilize entire port bandwidth. The total sum cannot exceed 100 percent. It is preferred to use this command when the **priority** command is used for certain queues in the policy. You can also assign ratios rather than a percentage to each queue; the queues will be assigned certain weights that are inline with these ratios. Ratios can range from 0 to 100. Total bandwidth ratio allocation for the policy in this case can exceed 100.
		
		Note You cannot mix bandwidth types on a policy map.

Step 5	**queue-limit** { *packets* **packets** | **cos** {*cos value* { *maximum threshold value* | **percent** *percentage* } | **values** {*cos value* | **percent** *percentage* } } | **dscp** {*dscp value* { *maximum threshold value* | **percent** *percentage* } | *match packet* { *maximum threshold value* | **percent** *percentage* } | **default** { *maximum threshold value* | **percent** *percentage* } | **ef** { *maximum threshold value* | **percent** *percentage* } | **dscp values** *dscp value* } | **percent** *percentage* } }
	
	Example:
	
	`Controller(config-pmap-c)# queue-limit dscp 3 percent 20`
	`Controller(config-pmap-c)# queue-limit dscp 4 percent 30`
	`Controller(config-pmap-c)# queue-limit`
		Sets the queue limit threshold percentage values.
		
		With every queue, there are three thresholds (0,1,2), and there are default values for each of these thresholds. Use this command to change the default or any other queue limit threshold setting. For example, if DSCP 3, 4, and 5 packets are being sent into a specific queue in a configuration, then you can use this command to set the threshold percentages for these three DSCP values. For additional information about queue limit threshold values, see Weighted Tail Drop, on page 235.
		
		Note The controller does not support absolute queue-limit percentages. The controller only supports dscp or cos queue-limit percentages.

	Command or Action	Purpose
	`dscp 5 percent 40`	

Step 6	**end**
	
	Example:
	
	`Controller(config-pmap-c)# end`
	`Controller#`
		Saves configuration changes.

Step 7	**show policy-map** **Example:** `Controller# show policy-map`	(Optional) Displays policy configuration information for all classes configured for all service policies.

What to Do Next

Proceed to configure any additional policy maps for QoS for your network. After creating your policy maps, proceed to attach the traffic policy or polices to an interface using the **service-policy** command.

Related Topics

Weighted Tail Drop, on page 235

Examples: Queue-limit Configuration, on page 287

Configuring Shaping

You use the **shape** command to configure shaping (maximum bandwidth) for a particular class. The queue's bandwidth is restricted to this value even though the port has additional bandwidth left. You can configure shaping as an average percent, as well as a shape average value in bits per second.

Before You Begin

You should have created a class map for shaping before beginning this procedure.

SUMMARY STEPS

1. **configure terminal**
2. **policy-map** *policy name*
3. **class** *class name*
4. **shape average** { *target bit rate* | **percent** *percentage* }
5. **end**
6. **show policy-map**

DETAILED STEPS

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

Step 2	**policy-map** *policy name*
	
	Example:
	
	`Controller(config)# policy-map policy_shaping01`
	`Controller(config-pmap)#`
		Enters policy map configuration mode.
		
		Creates or modifies a policy map that can be attached to one or more interfaces to specify a service policy.

Step 3	**class** *class name*
	
	Example:
	
	`Controller(config-pmap)# class class_shaping01`
	`Controller(config-pmap-c)#`
		Enters policy class map configuration mode. Specifies the name of the class whose policy you want to create or change. Command options for policy class map configuration mode include the following:
		
		• *word*—Class map name.
		
		• **class-default**—System default class matching any otherwise unclassified packets.

Step 4	**shape average** { *target bit rate* | **percent** *percentage* }
	
	Example:
	
	`Controller(config-pmap-c)# shape average percent 50`
	`Controller(config-pmap-c)#`
		Configures the average shape rate. You can configure the average shape rate by target bit rates (bits per second) or by percentage of interface bandwidth for the Committed Information Rate (CIR).

Step 5	**end**
	
	Example:
	
	`Controller(config-pmap-c)# end`
	`Controller#`
		Saves configuration changes.

Step 6	**show policy-map** **Example:** `Controller# show policy-map`	(Optional) Displays policy configuration information for all classes configured for all service policies.

What to Do Next

Configure any additional policy maps for QoS for your network. After creating your policy maps, attach the traffic policy or polices to an interface using the **service-policy** command.

Related Topics

Average Rate Shaping, on page 233

Examples: Average Rate Shaping Configuration, on page 286

Hierarchical Shaping, on page 233

Monitoring QoS

The following commands can be used to monitor QoS on the controller:

Table 39: Monitoring QoS

Command	Description
show class-map [*class_map_name*]	Displays a list of all class maps configured.
show policy-map [*policy_map_name*]	Displays a list of all policy maps configured. Command parameters include: • **policy map name** • **interface** • **session**

Command	Description
show policy-map interface { **Auto-template** | **Capwap** | **GigabitEthernet** | **GroupVI** | **InternalInterface** | **Loopback** | **Null** | **Port-channel** | **TenGigabitEthernet** | **Tunnel** | **Vlan** | **Brief** | **class** | **input** | **output** | **wireless** }	Shows the runtime representation and statistics of all the policies configured on the controller. Command parameters include: • **Auto-template—**Auto-Template interface • **Capwap—**CAPWAP tunnel interface • **GigabitEthernet—**Gigabit Ethernet IEEE.802.3z • **GroupVI—**Group virtual interface • **InternalInterface—**Internal interface • **Loopback—**Loopback interface • **Null—**Null interface • **Port-channel—**Ethernet channel of interfaces • **TenGigabitEthernet—**10-Gigabit Ethernet • **Tunnel—**Tunnel interface • **Vlan—**Catalyst VLANs • **Brief—**Brief description of policy maps • **Class—**Show statistics for individual class • **Input—**Input policy • **Output—**Output policy • **Wireless—**wireless
show policy-map interface wireless ap [*access point*]	Shows the runtime representation and statistics for all the wireless APs on the controller.
show policy-map interface wireless ssid [*ssid*]	Shows the runtime representation and statistics for all the SSID targets on the controller.

Command	Description
show policy-map interface wireless client [*client*]	Shows the runtime representation and statistics for all the client targets on the controller.
show policy-map session [**input** | **output** | **uid** *UUID*]	Shows the session QoS policy. Command parameters include: • **input**—Input policy • **output**—Output policy • **uid**—Policy based on SSS unique identification.
show table-map	Displays all the table maps and their configurations.

Configuration Examples for QoS

Examples: Classification by Access Control Lists

This example shows how to classify packets for QoS by using access control lists (ACLs):

```
Controller# configure terminal
Controller(config)# access-list 101 permit ip host 12.4.1.1 host 15.2.1.1
Controller(config)# class-map acl-101
Controller(config-cmap)# description match on access-list 101
Controller(config-cmap)# match access-group 101
Controller(config-cmap)#
```

After creating a class map by using an ACL, you then create a policy map for the class, and apply the policy map to an interface for QoS.

Related Topics

Creating a Traffic Class, on page 241

Class Maps, on page 225

Examples: Class of Service Layer 2 Classification

This example shows how to classify packets for QoS using a class of service Layer 2 classification:

```
Controller# configure terminal
Controller(config)# class-map cos
Controller(config-cmap)# match cos ?
  <0-7>  Enter up to 4 class-of-service values separated by white-spaces
Controller(config-cmap)# match cos 3 4 5
Controller(config-cmap)#
```

After creating a class map by using a CoS Layer 2 classification, you then create a policy map for the class, and apply the policy map to an interface for QoS.

Examples: Class of Service DSCP Classification

This example shows how to classify packets for QoS using a class of service DSCP classification:

```
Controller# configure terminal
Controller(config)# class-map dscp
Controller(config-cmap)# match dscp af21 af22 af23
Controller(config-cmap)#
```

After creating a class map by using a DSCP classification, you then create a policy map for the class, and apply the policy map to an interface for QoS.

Examples: VLAN ID Layer 2 Classification

This example shows how to classify for QoS using a VLAN ID Layer 2 classification:

```
Controller# configure terminal
Controller(config)# class-map vlan-120
Controller(config-cmap)# match vlan ?
  <1-4095>  VLAN id
Controller(config-cmap)# match vlan 120
Controller(config-cmap)#
```

After creating a class map by using a VLAN Layer 2 classification, you then create a policy map for the class, and apply the policy map to an interface for QoS.

Examples: Classification by DSCP or Precedence Values

This example shows how to classify packets by using DSCP or Precedence values:

```
Controller# configure terminal
Controller(config)# class-map prec2
Controller(config-cmap)# description matching precedence 2 packets
Controller(config-cmap)# match ip precedence 2
Controller(config-cmap)# exit
Controller(config)# class-map ef
Controller(config-cmap)# description EF traffic
Controller(config-cmap)# match ip dscp ef
Controller(config-cmap)#
```

After creating a class map by using a DSCP or Precedence values, you then create a policy map for the class, and apply the policy map to an interface for QoS.

Examples: Hierarchical Classification

The following is an example of a hierarchical classification, where a class named parent is created, which matches another class named child. The class named child matches based on the IP precedence being set to 2.

```
Controller# configure terminal
Controller(config)# class-map child
Controller(config-cmap)# match ip precedence 2
Controller(config-cmap)# exit
```

```
Controller(config)# class-map parent
Controller(config-cmap)# match class child
Controller(config-cmap)#
```

After creating the parent class map, you then create a policy map for the class, and apply the policy map to an interface for QoS.

Related Topics

Hierarchical QoS, on page 219

Examples: Hierarchical Policy Configuration

The following is an example of a configuration using hierarchical polices:

```
Controller# configure terminal
Controller(config)# class-map c1
Controller(config-cmap)# exit

Controller(config)# class-map c2
Controller(config-cmap)# exit

Controller(config)# class-map c3
Controller(config-cmap)# exit

Controller(config)# policy-map child
Controller(config-pmap)# class c1
Controller(config-pmap-c)# priority level 1
Controller(config-pmap-c)# police rate percent 20 conform-action transmit exceed action
drop
Controller(config-pmap-c-police)# exit
Controller(config-pmap-c)# exit
Controller(config-pmap)# class c2
Controller(config-pmap-c)# bandwidth 20000
Controller(config-pmap-c)# exit
Controller(config-pmap)# class class-default
Controller(config-pmap-c)# bandwidth 20000
Controller(config-pmap-c)# exit
Controller(config-pmap)# exit

Controller(config)# policy-map parent
Controller(config-pmap)# class class-default
Controller(config-pmap-c)# shape average 1000000
Controller(config-pmap-c)# service-policy child
Controller(config-pmap-c)# end
```

Examples: Classification for Voice and Video

This example describes how to classify packet streams for voice and video using controller specific information.

In this example, voice and video are coming in from end-point A into GigabitEthernet1/0/1 on the controller and have precedence values of 5 and 6, respectively. Additionally, voice and video are also coming from end-point B into GigabitEthernet1/0/2 on the controller with DSCP values of EF and AF11, respectively.

Assume that all the packets from the both the interfaces are sent on the uplink interface, and there is a requirement to police voice to 100 Mbps and video to 150 Mbps.

To classify per the above requirements, a class to match voice packets coming in on GigabitEthernet1/0/1 is created, named voice-interface-1, which matches precedence 5. Similarly another class for voice is created, named voice-interface-2, which will match voice packets in GigabitEthernet1/0/2. These classes are associated to two separate policies named input-interface-1, which is attached to GigabitEthernet1/0/1, and

input-interface-2, which is attached to GigabitEthernet1/0/2. The action for this class is to mark the qos-group to 10. To match packets with QoS-group 10 on the output interface, a class named voice is created which matches on QoS-group 10. This is then associated to another policy named output-interface, which is associated to the uplink interface. Video is handled in the same way, but matches on QoS-group 20.

The following example shows how classify using the above controller specific information:

```
Controller(config)#
Controller(config)# class-map voice-interface-1
Controller(config-cmap)# match ip precedence 5
Controller(config-cmap)# exit

Controller(config)# class-map video-interface-1
Controller(config-cmap)# match ip precedence 6
Controller(config-cmap)# exit

Controller(config)# class-map voice-interface-2
Controller(config-cmap)# match ip dscp ef
Controller(config-cmap)# exit

Controller(config)# class-map video-interface-2
Controller(config-cmap)# match ip dscp af11
Controller(config-cmap)# exit

Controller(config)# policy-map input-interface-1
Controller(config-pmap)# class voice-interface-1
Controller(config-pmap-c)# set qos-group 10
Controller(config-pmap-c)# exit

Controller(config-pmap)# class video-interface-1
Controller(config-pmap-c)# set qos-group 20

Controller(config-pmap-c)# policy-map input-interface-2
Controller(config-pmap)# class voice-interface-2
Controller(config-pmap-c)# set qos-group 10
Controller(config-pmap-c)# class video-interface-2
Controller(config-pmap-c)# set qos-group 20
Controller(config-pmap-c)# exit
Controller(config-pmap)# exit

Controller(config)# class-map voice
Controller(config-cmap)# match qos-group 10
Controller(config-cmap)# exit

Controller(config)# class-map video
Controller(config-cmap)# match qos-group 20

Controller(config)# policy-map output-interface
Controller(config-pmap)# class voice
Controller(config-pmap-c)# police 256000 conform-action transmit exceed-action drop
Controller(config-pmap-c-police)# exit
Controller(config-pmap-c)# exit

Controller(config-pmap)# class video
Controller(config-pmap-c)# police 1024000 conform-action transmit exceed-action drop
Controller(config-pmap-c-police)# exit
Controller(config-pmap-c)# exit
```

Examples: Average Rate Shaping Configuration

The following example shows how to configure average rate shaping:

```
Controller# configure terminal
Controller(config)# class-map prec1
Controller(config-cmap)# description matching precedence 1 packets
```

```
Controller(config-cmap)# match ip precedence 1
Controller(config-cmap)# end

Controller# configure terminal
Controller(config)# class-map prec2
Controller(config-cmap)# description matching precedence 2 packets
Controller(config-cmap)# match ip precedence 2
Controller(config-cmap)# exit

Controller(config)# policy-map shaper
Controller(config-pmap)# class prec1
Controller(config-pmap-c)# shape average 512000
Controller(config-pmap-c)# exit

Controller(config-pmap)# policy-map shaper
Controller(config-pmap)# class prec2
Controller(config-pmap-c)# shape average 512000
Controller(config-pmap-c)# exit

Controller(config-pmap)# class class-default
Controller(config-pmap-c)# shape average 1024000
```

After configuring the class maps, policy map, and shape averages for your configuration, proceed to then apply the policy map to the interface for QoS.

Related Topics

Configuring Shaping, on page 279

Average Rate Shaping, on page 233

Examples: Queue-limit Configuration

The following example shows how to configure a queue-limit policy based upon DSCP values and percentages:

```
Controller# configure terminal
Controller#(config)# policy-map port-queue
Controller#(config-pmap)# class dscp-1-2-3
Controller#(config-pmap-c)# bandwidth percent 20
Controller#(config-pmap-c)# queue-limit dscp 1 percent 80
Controller#(config-pmap-c)# queue-limit dscp 2 percent 90
Controller#(config-pmap-c)# queue-limit dscp 3 percent 100
Controller#(config-pmap-c)# exit

Controller#(config-pmap)# class dscp-4-5-6
Controller#(config-pmap-c)# bandwidth percent 20
Controller#(config-pmap-c)# queue-limit dscp 4 percent 20
Controller#(config-pmap-c)# queue-limit dscp 5 percent 30
Controller#(config-pmap-c)# queue-limit dscp 6 percent 20
Controller#(config-pmap-c)# exit

Controller#(config-pmap)# class dscp-7-8-9
Controller#(config-pmap-c)# bandwidth percent 20
Controller#(config-pmap-c)# queue-limit dscp 7 percent 20
Controller#(config-pmap-c)# queue-limit dscp 8 percent 30
Controller#(config-pmap-c)# queue-limit dscp 9 percent 20
Controller#(config-pmap-c)# exit

Controller#(config-pmap)# class dscp-10-11-12
Controller#(config-pmap-c)# bandwidth percent 20
Controller#(config-pmap-c)# queue-limit dscp 10 percent 20
Controller#(config-pmap-c)# queue-limit dscp 11 percent 30
Controller#(config-pmap-c)# queue-limit dscp 12 percent 20
Controller#(config-pmap-c)# exit

Controller#(config-pmap)# class dscp-13-14-15
Controller#(config-pmap-c)# bandwidth percent 10
```

```
Controller#(config-pmap-c)# queue-limit dscp 13 percent 20
Controller#(config-pmap-c)# queue-limit dscp 14 percent 30
Controller#(config-pmap-c)# queue-limit dscp 15 percent 20
Controller#(config-pmap-c)# end
Controller#
```

After finishing with the above policy map queue-limit configuration, you can then proceed to apply the policy map to an interface for QoS.

Related Topics

Configuring Queue Limits, on page 276

Weighted Tail Drop, on page 235

Examples: Queue Buffers Configuration

The following example shows how configure a queue buffer policy and then apply it to an interface for QoS:

```
Controller# configure terminal
Controller(config)# policy-map policy1001
Controller(config-pmap)# class class1001
Controller(config-pmap-c)# bandwidth remaining ratio 10
Controller(config-pmap-c)# queue-buffer ratio ?
  <0-100>  Queue-buffers ratio limit
Controller(config-pmap-c)# queue-buffer ratio 20
Controller(config-pmap-c)# end

Controller# configure terminal
Controller(config)# interface gigabitEthernet2/0/3
Controller(config-if)# service-policy output policy1001
Controller(config-if)# end
```

Related Topics

Configuring Queue Buffers, on page 274

Queue Buffer Allocation, on page 237

Examples: Policing Action Configuration

The following example displays the various policing actions that can be associated to the policer. These actions are accomplished using the conforming, exceeding, or violating packet configurations. You have the flexibility to drop, mark and transmit, or transmit packets that have exceeded or violated a traffic profile.

For example, a common deployment scenario is one where the enterprise customer polices traffic exiting the network towards the service provider and marks the conforming, exceeding and violating packets with different DSCP values. The service provider could then choose to drop the packets marked with the exceeded and violated DSCP values under cases of congestion, but may choose to transmit them when bandwidth is available.

Note The Layer 2 fields can be marked to include the CoS fields, and the Layer 3 fields can be marked to include the precedence and the DSCP fields.

One useful feature is the ability to associate multiple actions with an event. For example, you could set the precedence bit and the CoS for all conforming packets. A submode for an action configuration could then be provided by the policing feature.

This is an example of a policing action configuration:

```
Controller# configure terminal
Controller(config)# policy-map police
Controller(config-pmap)# class class-default
Controller(config-pmap-c)# police cir 1000000 pir 2000000
Controller(config-pmap-c-police)# conform-action transmit
Controller(config-pmap-c-police)# exceed-action set-dscp-transmit dscp table
exceed-markdown-table
Controller(config-pmap-c-police)# violate-action set-dscp-transmit dscp table
violate-markdown-table
Controller(config-pmap-c-police)# end
```

In this example, the exceed-markdown-table and violate-mark-down-table are table maps.

Note Policer based markdown actions are only supported using table maps. Only one markdown table map is allowed for each marking field in the controller.

Related Topics

Configuring Police, on page 269

Policing, on page 227

Examples: Policer VLAN Configuration

The following example displays a VLAN policer configuration. At the end of this configuration, the VLAN policy map is applied to an interface for QoS.

```
Controller# configure terminal
Controller(config)# class-map vlan100
Controller(config-cmap)# match vlan 100
Controller(config-cmap)# exit
Controller(config)# policy-map vlan100
Controller(config-pmap)# policy-map class vlan100
Controller(config-pmap-c)# police 100000 bc conform-action transmit exceed-action drop
Controller(config-pmap-c-police)# end
Controller# configure terminal
Controller(config)# interface gigabitEthernet1/0/5
Controller(config-if)#  service-policy input vlan100
```

Related Topics

Classifying, Policing, and Marking Traffic on SVIs by Using Policy Maps, on page 258

Policy Map on VLANs, on page 227

Examples: Policing Units

The following examples display the various units of policing that are supported for QoS. The policing unit is the basis on which the token bucket works .

The following units of policing are supported:

- CIR and PIR are specified in bits per second. The burst parameters are specified in bytes. This is the default mode; it is the unit that is assumed when no units are specified. The CIR and PIR can also be configured in percent, in which case the burst parameters have to be configured in milliseconds.
- CIR and PIR are specified in packets per second. In this case, the burst parameters are configured in packets as well.

The following is an example of a policer configuration in bits per second:

```
Controller(config)# policy-map bps-policer
Controller(config-pmap)# class class-default
Controller(config-pmap-c) # police rate 256000 bps burst 1000 bytes
conform-action transmit exceed-action drop
```

The following is an example of a policer configuration in packets per second. In this configuration, a dual-rate three-color policer is configured where the units of measurement is packet. The burst and peak burst are all specified in packets.

```
Controller(config)# policy-map pps-policer
Controller(config-pmap)# class class-default
Controller(config-pmap-c)# police rate 5000 pps burst 100 packets
peak-rate 10000 pps peak-burst 200 packets conform-action transmit
exceed-action drop violate-action drop
```

Related Topics

Configuring Police, on page 269

Token-Bucket Algorithm, on page 228

Examples: Single-Rate Two-Color Policing Configuration

The following example shows how to configure a single-rate two-color policer:

```
Controller(config)# class-map match-any prec1
Controller(config-cmap)# match ip precedence 1
Controller(config-cmap)# exit
Controller(config)# policy-map policer
Controller(config-pmap)# class prec1
Controller(config-pmap-c)# police cir 256000 conform-action transmit exceed-action drop
Controller(config-pmap-c-police)# exit
Controller(config-pmap-c)#
```

Related Topics

Configuring Police, on page 269

Single-Rate Two-Color Policing, on page 231

Examples: Dual-Rate Three-Color Policing Configuration

The following example shows how to configure a dual-rate three-color policer:

```
Controller# configure terminal
Controller(config)# policy-Map dual-rate-3color-policer
Controller(config-pmap)# class class-default
Controller(config-pmap-c)# police cir 64000 bc 2000 pir 128000 be 2000
Controller(config-pmap-c-police)# conform-action transmit
```

```
Controller(config-pmap-c-police)# exceed-action set-dscp-transmit dscp table
exceed-markdown-table
Controller(config-pmap-c-police)# violate-action set-dscp-transmit dscp table
violate-markdown-table
Controller(config-pmap-c-police)# exit
Controller(config-pmap-c)#
```

In this example, the exceed-markdown-table and violate-mark-down-table are table maps.

Note Policer based markdown actions are only supported using table maps. Only one markdown table map is allowed for each marking field in the controller.

Related Topics

Configuring Police, on page 269

Dual-Rate Three-Color Policing, on page 231

Examples: Table Map Marking Configuration

The following steps and examples show how to use table map marking for your QoS configuration:

1 Define the table map.

Define the table-map using the **table-map** command and indicate the mapping of the values. This table does not know of the policies or classes within which it will be used. The default command in the table map indicates the value to be copied into the 'to' field when there is no matching 'from' field. In the example, a table map named table-map1 is created. The mapping defined is to convert the value from 0 to 1 and from 2 to 3, while setting the default value to 4.

```
Controller(config)# table-map table-map1
Controller(config-tablemap)# map from 0 to 1
Controller(config-tablemap)# map from 2 to 3
Controller(config-tablemap)# default 4
Controller(config-tablemap)# exit
```

2 Define the policy map where the table map will be used.

In the example, the incoming CoS is mapped to the DSCP based on the mapping specified in the table table-map1. For this example, if the incoming packet has a DSCP of 0, the CoS in the packet is set 1. If no table map name is specified the command assumes a default behavior where the value is copied as is from the 'from' field (DSCP in this case) to the 'to' field (CoS in this case). Note however, that while the CoS is a 3-bit field, the DSCP is a 6-bit field, which implies that the CoS is copied to the first three bits in the DSCP.

```
Controller(config)# policy map policy1
Controller(config-pmap)# class class-default
Controller(config-pmap-c)# set cos dscp table table-map1
Controller(config-pmap-c)# exit
```

3 Associate the policy to an interface.

```
Controller(config)# interface GigabitEthernet1/0/1
Controller(config-if)# service-policy output policy1
Controller(config-if)# exit
```

Related Topics

Configuring Table Maps, on page 261

Table Map Marking, on page 229

Example: Table Map Configuration to Retain CoS Markings

The following example shows how to use table maps to retain CoS markings on an interface for your QoS configuration.

The cos-trust-policy policy (configured in the example) is enabled in the ingress direction to retain the CoS marking coming into the interface. If the policy is not enabled, only the DSCP is trusted by default. If a pure Layer 2 packet arrives at the interface, then the CoS value will be rewritten to 0 when there is no such policy in the ingress port for CoS.

```
Controller# configure terminal
Controller(config)# table-map cos2cos
Controller(config-tablemap)# default copy
Controller(config-tablemap)# exit


Controller(config)# policy map cos-trust-policy
Controller(config-pmap)# class class-default
Controller(config-pmap-c)# set cos cos table cos2cos
Controller(config-pmap-c)# exit

Controller(config)# interface GigabitEthernet1/0/2
Controller(config-if)# service-policy input cos-trust-policy
Controller(config-if)# exit
```

Additional References for QoS

Related Documents

Related Topic	Document Title
QoS CLI commands	
Cisco Flexible NetFlow	*Flexible NetFlow Configuration Guide, Cisco IOS XE Release 3.2SE (Cisco 5700 Series Wireless Controller)*
Call Admission Control (CAC)	*System Management Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 5700 Series Wireless Controller)*

Standards and RFCs

Standard/RFC	Title
Not applicable	

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

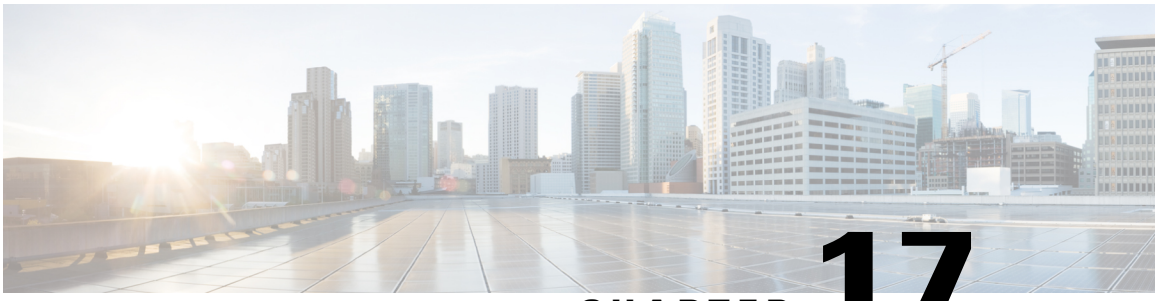
Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, 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 QoS

This table lists the features in this module and provides links to specific configuration information.

Table 40: Feature Information for QoS

Feature Name	Release	Feature Information
QoS functionality		The following functionality is supported: • QoS for wired targets • QoS for wireless targets



CHAPTER 17

Configuring Wireless QoS

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Wireless QoS

Before configuring wireless QoS, you must have a thorough understanding of these items:

- QoS concepts.
- Wireless concepts and network topologies.
- Classic Cisco IOS QoS.
- Modular QoS CLI (MQC).
- Understanding of QoS implementation.
- 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.

Restrictions for Wireless QoS

General Restrictions

- A target is an entity where a policy is applied. You can apply a policy to either a wired or wireless target. A wired target can be either a port, client, or VLAN. A wireless target can be either a port, SSID, client, or radio. Wireless QoS policies for port, SSID, client, and radio are applied in the downstream direction. That is, when traffic is flowing from the controller to wireless client.

  Only port, SSID, and client (using AAA and Cisco IOS command-line interface) policies are user-configurable. Radio policies are set by the wireless control module and are not user-configurable.
- Port and radio policies are applicable only in the downstream direction (traffic flowing from a wired source to a wireless target).
- SSID and client support non-queuing policies in the upstream direction. SSID and client targets can be configured with marking and policing policies.
- One policy per target per direction is supported.
- For marking rules for access points associated with the controller, the following rules apply:
  - Policing at the access point is not supported.
  - Client policies that are passed to the access points in the upstream direction are not supported.
  - The following rules apply for QoS at the SSID:
    - One table map is supported at the ingress policy.
    - Up to three table maps can be configured in the egress direction for SSID when a QoS-group is involved.

    Note Table maps are not supported at the client targets.

  - IPv6 QoS for wireless clients is not supported.
- Port policies are not supported. Radio, SSID, and client policies are supported.
- QoS policies are not supported on wired clients. To configure QoS on wired clients, configure the policy on the port. The UP values for wireless packets can be added when the ports are configured with the policies.
- By default, all wireless traffic is treated as untrusted. That is, in the absence of a table map on the SSID, the packets coming in and going out of the wireless targets are re-marked down to 0. To change the

wireless traffic from being untrusted by default to being trusted, you must configure the table map to ensure appropriate packet markings are defined.

Wireless QoS Restrictions on SSID

The following are restrictions for applying QoS features on SSID:

- Set and priority cannot coexist under the same class.
- Set (non-table map) policies at the SSID level will be restricted.
- Policing (without priority) is not supported in both the ingress and egress direction.
- Priority configuration at the SSID level is used only to configure the RT1 and RT2 policers (AFD for policer). Priority configuration does not include the shape rate. Therefore, priority is restricted for SSID policies without police.
- If **set** is not enabled in class-default, the classification at the SSID for voice or video must be a subset of the classification for the voice or video class at the port level.
- The mapping in the DSCP2DSCP and COS2COS table should be based on the classification mechanism for the voice and video classes in the port level policy.
- Two or three **set** commands with table-map related functionality is supported in the parent class-default policy.
- If one SSID policy has only voice class in a QoS configuration, for example:

```
SSID-1
  Class voice
  Priority level 1
  Class video (af11)
  Priority level 2
```

And another SSID policy has both voice and video class, for example:

```
SSID-2
  Class voice
  Priority level 1
  Class video (af11)
  Priority level 2
```

The table map for the SSID 1 must be configured so that the video DSCP is best effort marking. This action ensures that video packets for SSID1 are mapped to the NRT queue at the port level.

- The **set table-maps** configuration can only be applied in the parent class-default at the SSID level.
- Policing not supported in the upstream direction.
- Table map action is supported only in this class-default class.
- No action is allowed under the class-default of a child policy.
- For a flat policy (non-hierarchical):
  - In the ingress direction, the policy configuration must be a set or policing policy.

  For a hierarchical policy:
  - In the egress direction, the following is supported:
    - Table map in parent with only police allowed in child.

- Table map and queuing configured in the parent policy—The parent policy will have a Bandwidth Remaining Ratio (BRR) or shape (either one can be configured), and the **set (table-map)** configuration command. The child policy should have a priority and police configuration, in which the priority matches the priority level configured at the AP port.

Wireless QoS Restrictions on Radio

The following are restrictions for applying QoS policies on radio targets:

- Ingress policies are not supported.
- For egress policies, only the shape and BRR queuing are supported on class-default.

Wireless QoS Restrictions on Clients

The following are restrictions for applying QoS policies on client targets:

- Queuing is not supported.
- Attaching or removing client policies on a WLAN in the up state is not supported. You must shut down the WLAN to apply or remove a policy.
- Table map based set is blocked at the client.
- Police in child policy class-default is not supported in the egress and ingress directions. For example, the following policy is not supported:

```
Policy-map parent-client
class class-default
police X
service-policy child-client

Policy-map child-client
class class-default
police Y
```

- Policing and set in class-default is blocked in both the upstream and downstream direction:

```
policy-map foo
class class-default
police X
set dscp Y
```

- The following policy configuration is not supported:

```
policy map foo
  class acl-101 (match on 3 tuple)
  Police X
  class acl-102
  (match on 5 tuple)
Police Y
```

- In a flat (nonhierarchical) policy, only police is supported. For user-defined classes, only the following filters are supported:
  - ACL
  - DSCP
  - COS

- WLAN UP

- No child-policy support under class-default, if the parent policy contains other user-defined class maps in it.

Related Topics

Queuing in Wireless, on page 306

Port Policy Format, on page 304

Port, on page 301

Radio, on page 302

Restrictions for QoS on Wired Targets, on page 216

Information about Wireless QoS

Wireless QoS Overview

Wireless QoS can be configured on the following wireless targets:

- Wireless ports
- Radio
- SSID (applicable on a per-radio, per-AP, and per-SSID)
- Client

QoS policies are configured by using Modular QoS CLI (MQC). Port, SSID, and client policies are user configurable. Radio policies are controlled by the wireless control module.

A target is the entity where the policy is applied. Wireless QoS policies for port, SSID, client, and radio are applied in the downstream direction. That is, when traffic is flowing from the controller to wireless client.

Note SSID and client policies are supported only in the upstream direction.

The following are some of the specific features provided by wireless QoS:

- Policies on wireless QoS targets:
  - port
  - radio
  - SSID
  - client
- Queuing support
- Policing of wireless traffic
- Shaping of wireless traffic

- Rate limiting in both downstream and upstream direction
- Approximate Fair Drop (AFD)
- Mobility support for QoS
- Compatibility with precious metal QoS policies available on Cisco Unified Wireless Controllers.

User-Defined Policies

You can configure the following kinds of QoS policies:

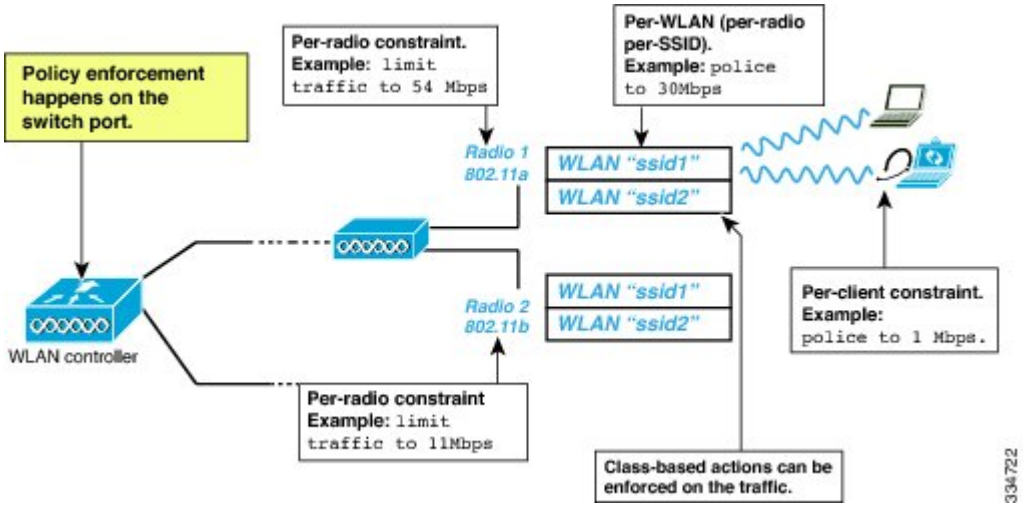
- Port policies
- SSID policies
- Client policies
- Multidestination policers
- VLAN policies

Hierarchical Wireless QoS

The controller supports hierarchical QoS for wireless targets. Hierarchical QoS policies are applicable on port, radio, SSID, and client. QoS policies configured on the device (including marking, shaping, policing) can be applied across the targets. If the network contains non-realtime traffic, the non-realtime traffic is subject to approximate fair drop. Hierarchy refers to the process of application of the various QoS policies on the packets arriving to the device.

This figure shows the various targets available on a wireless network.

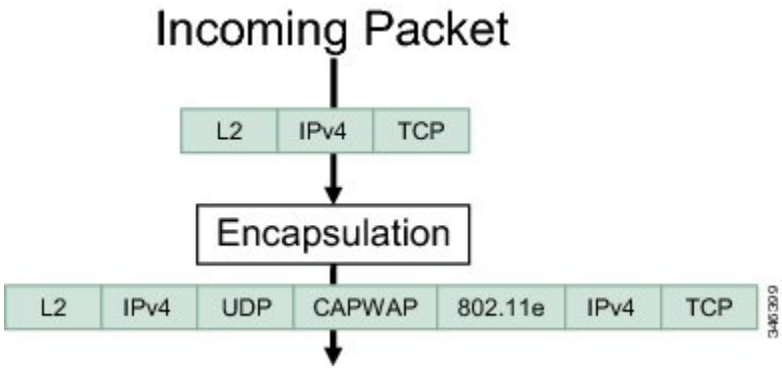
Figure 5: Hierarchical QoS



Wireless Packet Format

This figure describes the wireless packet flow. The incoming packet enters the controller. The controller encapsulates this incoming packet and adds the 802.11e and CAPWAP headers.

Figure 6: Wireless Packet Path in the Egress direction during first pass



Hierarchical AFD

Approximate Fair Dropping (AFD) is a feature provided by the QoS infrastructure in Cisco IOS. For wireless targets, AFD can be configured on SSID (via shaping) and clients (via policing). AFD shaping rate is only applicable for downstream direction. Unicast real-time traffic is not subjected to AFD drops.

Wireless QoS Targets

This section describes the various wireless QoS targets available on a controller.

Port

The controller supports port-based policies. The port policies includes port shaper and a child policy (port_child_policy).

Port shaper specifies the traffic policy between the device to the AP. This is the sum of the radio rates supported on the access point.

The child policy determines the mapping between packets and queues defined by the port-child policy. The child policy can be configured to include voice, video, and class-default classes where voice and video are based on DSCP value (which is the outer CAPWAP header DSCP value). The definition of class-default is known to the system as any value other than voice and video DSCP.

The DSCP value is assigned when the packet reaches the port. Before the packet arrives at the port, the SSID policies are applied on the packet. Port child policy also includes multicast percentage for a given port traffic. By default, the port child policy allocates up to 10 percent of the available rate.

Related Topics

Queuing in Wireless, on page 306

Restrictions for Wireless QoS, on page 296

Supported QoS Features on Wireless Targets, on page 303

Examples: Wireless QoS Policy Classified by Voice, Video, and Multicast Traffic, on page 315

Radio

The controller enables you to create policies. The radio policies are system defined and are not user configurable. Radio wireless targets are only applicable in the downstream direction.

Radio policies are applicable on a per-radio, per-access point basis. The rate limit on the radios is the practical limit of the AP radio rate.

Related Topics

Restrictions for Wireless QoS, on page 296

Supported QoS Features on Wireless Targets, on page 303

SSID

You can create QoS policies on SSID (BSSID) in both the upstream and downstream directions. By default, there is no SSID policy. All traffic is transmitted as best effort because the wireless traffic in untrusted. You can configure an SSID policy based on the SSID name. The policy is applicable on a per BSSID.

The types of policies you can create on SSID include marking by using table maps (table-maps), shape rate, and RT1 and RT2 policiers. If traffic is upstream, you usually configure a marking policy on the SSID. If traffic is downstream, you can configure marking and queuing.

There should be a one-to-one mapping between the policies configured on port and SSID. For example, if you configure class voice and class video on the port, you can have a similar policy on the SSID.

The policy on the port is mandatory if you want to preserve the voice and video behavior priority at the port level. Queuing policy is applicable in a downstream direction. You can configure one queue in the upstream direction. When packets arrive from the AP, you can only configure policing and rate limiting.

SSID priorities can be specified by configuring bandwidth remaining ratio. Queuing SSID policies are applied in the downstream direction.

Related Topics

Supported QoS Features on Wireless Targets, on page 303

Examples: SSID Policy, on page 316

Examples: Configuring Downstream BSSID Policy, on page 316

Client

Client policies are applicable in the upstream and downstream direction. The wireless control module of the controller applies the client policies when admission control is enabled for WMM clients. When admission control is disabled, there is no default client policy. You can configure policing and marking policies on clients.

You can configure client policies in the following ways:

- Using AAA—You can use a combination of AAA and TCLAS, AAA and SIP snooping when configuring via AAA.
- Using the IOS MQC CLI—You can use a combination of CLI and TCLAS and CLI and SIP snooping.

- Using the default configuration

Note When an installed policy gets modified on a WLAN, the WLAN must be restarted for the changes to take effect. For SSID policies, a restart is not required.

Note If you configured AAA by configuring the classic unified wireless controller procedure, and using the MQC QoS commands, the policy configuration performed through the MQC QoS commands takes precedence.

Related Topics

Supported QoS Features on Wireless Targets, on page 303

Examples: Client Policies, on page 317

Supported QoS Features on Wireless Targets

This table describes the various features available on wireless targets.

Table 41: QoS Features Available on Wireless Targets

Target	Features	Traffic	Direction Where Policies Are Applicable	Comments
Port	• Port Shaper • Priority Queuing • Shaping • Multicast Policing • HQF (Hierarchical QoS Framework) • BRR (Bandwidth Remaining Ratio)	Non-Real Time (NRT), Real Time (RT)	Downstream	
Radio	• Shaping	Non-Real Time	Downstream	Radio policies are not user configurable.

Target	Features	Traffic	Direction Where Policies Are Applicable	Comments
SSID	• Shaping • Police • Set • Table map • BRR **Note** Set without table map is not supported on SSID.	Non-Real Time, Real Time	Upstream and downstream	Queuing actions such as shaping and BRR are allowed only in the downstream direction.
Client	• Set • Police	Non-Real Time, Real time	Upstream and downstream	

Downstream Traffic

Traffic flows from a wired source to a wireless target.

Upstream Traffic

Traffic flows from a wireless source to a wired target.

Related Topics

Queuing in Wireless, on page 306
Port Policy Format, on page 304
Port, on page 301
Radio, on page 302
SSID, on page 302
Client, on page 302

Port Policy Format

To configure quality of service (QoS) on a controller, you must configure a port map policy. Because a Cisco 5700 Series Wireless Controller contains 6 10-Gigabit ports, the policy map must be configured on all ports.

Note The policy must be configured on all of the six physical ports on the controller even if LAG is configured.

The following basic port policy must be configured on the physical ports. You can add further classification if required:

```
Policy-map <port-policy-name>
  Class voice
      Priority level 1
  class video
     Priority level 2
```

Related Topics

Queuing in Wireless, on page 306

Restrictions for Wireless QoS, on page 296

Supported QoS Features on Wireless Targets, on page 303

Examples: Wireless QoS Policy Classified by Voice, Video, and Multicast Traffic, on page 315

Wireless QoS Rate Limiting

QoS per Client Rate Limit—Wireless

You can configure client rate limiting by the following means:

- AFD
- NetFlow policing

Note For client policy, the voice and video rate limits are applied at the same time.

QoS Downstream Rate Limit—Wireless

Downstream rate limiting is done using policing at the SSID level. AFD cannot drop real-time traffic, it can only be policed in the traffic queues. Real-time policing and AFD shaping is performed at the SSID level.

The radio has a default shaping policy. This shaping limit is the physical limit of the radio itself. For example, the 802.11b/g/n radio can shape up to 100 Mbps on a 2.4 GHz frequency. You can check the policy maps on the radio by using the **show policy-map interface wireless radio** command.

Note The controller does not support port shaping rate.

QoS Upstream Rate Limit—Wireless

Note The controller does not support upstream rate-limiting policy

Wireless QoS Multicast

There are two modes of Multicast configuration in Cisco 5700 Series Wireless Controller:

- multicast-unicast mode: Multicast traffic is copied as unicast traffic to the APs. QoS on multicast traffic when multicast-unicast mode is not supported on 5700.
- multicast-multicast mode: In this scenario, the controller sends the traffic to the multicast group. The APs in the multicast group are then served the multicast traffic.

Related Topics

Configuring QoS Policy for Multicast Traffic, on page 314

Queuing in Wireless

Queuing in the wireless component is performed based on the port policy and is applicable only in the downstream direction. The wireless module supports the following four queues:

- Voice—This is a strict priority queue. Represented by Q0, this queue processes control traffic and multicast or unicast voice traffic. All control traffic (such as CAPWAP packets) is processed through the voice queue. The QoS module uses a different threshold within the voice queue to process control and voice packets to ensure that control packets get higher priority over other non-control packets.
- Video—This is a strict priority queue. Represented by Q1, this queue processes multicast or unicast video traffic.
- Data NRT—Represented by Q2, this queue processes all non-real-time unicast traffic.
- Multicast NRT—Represented by Q3, this queue processes Multicast NRT traffic. Any traffic that does not match the traffic in Q0, Q1, or Q2 is processed through Q3.

Note By default, the queues Q0 and Q1 are not enabled.

Note A weighted round-robin policy is applied for traffic in the queues Q2 and Q3.

For upstream direction only one queue is available. Port and radio policies are applicable only in the downstream direction.

Note The queues on the controller is different from the queues on the access points.

Related Topics

Port Policy Format, on page 304

Port, on page 301

Restrictions for Wireless QoS, on page 296

Supported QoS Features on Wireless Targets, on page 303
Examples: Wireless QoS Policy Classified by Voice, Video, and Multicast Traffic, on page 315

Wireless QoS Mobility

Wireless QoS Mobility enables you to configure QoS policies so that the network provides the same service anywhere in the network. A wireless client can roam from one location to another and as a result the client can get associated to different access points associated with a different controller. Wireless client roaming can be classified into two types:

- Intra-controller roaming
- Inter-controller roaming

Note The client policies must be available on all of the controllers in the mobility group. The same SSID and port policy must be applied to all controllers in the mobility group so that the clients get consistent treatment.

Inter-Controller Roaming

When a client roams from one location to another, the client can get associated to access points either associated to the same controller (anchor controller) or a different controller (foreign controller). Inter-controller roaming refers to the scenario where the client gets associated to an access point that is not associated to the same device before the client roamed. The host device is now foreign to the device to which the client was initially anchored.

In the case of inter-controller roaming, the client QoS policy is always executed on the foreign controller. When a client roams from anchor controller to foreign controller, the QoS policy is uninstalled on the anchor controller and installed on the foreign controller. In the mobility handoff message, the anchor device passes the name of the policy to the foreign controller. The foreign controller should have a policy with the same name configured for the QoS policy to be applied correctly.

In the case of inter-controller roaming, all of the QoS policies are moved from the anchor device to the foreign device. While the QoS policies are in transition from the anchor device to the foreign device, the traffic on the foreign device is provided the default treatment. This is comparable to a new policy installation on the client target.

Note If the foreign device is not configured with the user-defined physical port policy, the default port policy is applicable to all traffic is routed through the NRT queue, except the control traffic which goes through RT1 queue. The network administrator must configure the same physical port policy on both the Anchor and Foreign devices symmetrically.

Intra-Controller Roaming

With intra-controller roaming, the client gets associated to an access point that is associated to the same controller before the client roamed, but this association to the device occurs through a different access point.

Note QoS policies remain intact in the case of intra-controller roaming.

Precious Metal Policies for Wireless QoS

Wireless QoS is backward compatible with the precious metal policies offered by the unified wireless controller platforms. The precious metal policies are system-defined policies that are available on the controller.

The following policies are available:

- Platinum—Used for VoIP clients.
- Gold—Used for video clients.
- Silver— Used for traffic that can be considered best-effort.
- Bronze—Used for NRT traffic.

These policies (also known as profiles) can be applied to a WLAN based on the traffic. We recommend the configuration via the Cisco IOS MQC configuration. The policies are available in the system based on the precious metal policy required.

Based on the policies applied, the 802.11p, 802.11e (WMM) and DSCP fields in the packets are affected. These values are pre configured and installed when the controller is booted.

Note Unlike the precious metal policies that were applicable in the Cisco Unified Wireless controllers, the attributes `rt-average-rate`, `nrt-average-rate`, and peak rates are not applicable for the precious metal policies configured on this controller platform.

Related Topics

Configuring Precious Metal Policies, on page 308

How to Configure Wireless QoS

Configuring Precious Metal Policies

You can configure precious metal QoS policies on a per-WLAN basis.

Note Upstream policies differ from downstream policies. The upstream policies have a suffix of -up.

Before You Begin

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan-name*
3. **service-policy** [**client**] **output** *policy-name*
4. **end**
5. **show wlan** {*wlan-id* | *wlan-name*}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global command mode.
Step 2	**wlan** *wlan-name* **Example:** `Controller(config)# wlan test4`	Enters the WLAN configuration sub-mode.
Step 3	**service-policy** [**client**] **output** *policy-name* **Example:** `Controller(config-wlan)# service-policy output platinum`	Configures the WLAN with the QoS policy. To configure the WLAN with precious metal policies, you must enter one of the following keywords: **platinum**, **gold**, **silver**, or **bronze**.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit the global configuration mode.

Step 5	**show wlan** {*wlan-id* | *wlan-name*}
	
	Example:
	`Controller# show wlan qos-wlan`
		Verifies the configured QoS policy on the WLAN.
		
		`Controller# show wlan qos-wlan`
		`. . .`
		`. . .`
		`. . .`
		`QoS Service Policy - Output`
		`  Policy Name : platinum`
		`  Policy State : Validation Pending`
		`QoS Client Service Policy`
		`  Input  Policy Name : gold`
		`  Output Policy Name : qos-wlan-client-service-policy`
		`. . .`
		`. . .`

Related Topics

Precious Metal Policies for Wireless QoS, on page 308

Configuring Class Maps for Voice and Video

To configure class maps for voice and video traffic, follow these steps:

SUMMARY STEPS

1. **configure terminal**
2. **class-map** *class-map-name*
3. **match dscp** *dscp-value-for-voice*
4. **end**
5. **configure terminal**
6. **class-map** *class-map-name*
7. **match dscp** *dscp-value-for-video*
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**class-map** *class-map-name* **Example:** `Controller(config)# class-map voice`	Creates a class map.
Step 3	**match dscp** *dscp-value-for-voice* **Example:** `Controller(config-cmap)# match dscp 46`	Matches the DSCP value in the IPv4 and IPv6 packets. Set this value to 46.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.
Step 5	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 6	**class-map** *class-map-name* **Example:** `Controller(config)# class-map video`	Configures a class map.
Step 7	**match dscp** *dscp-value-for-video* **Example:** `Controller(config-cmap)# match dscp 34`	Matches the DSCP value in the IPv4 and IPv6 packets. Set this value to 34.
Step 8	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring Client Policies

Before You Begin

You must have the following features configured before configuring client policies:

- Access lists
- Access group name

SUMMARY STEPS

1. **configure terminal**
2. **ip access-list extended** *ext-name*
3. **permit ip host** *host-ip-address*
4. **end**
5. **configure terminal**
6. **class map** *acl-name*
7. **match access-group name** *access-list-name*
8. **end**

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	**ip access-list extended** *ext-name* **Example:** `Controller(config)# ip access-list extended`	Configures a named access list.
Step 3	**permit ip host** *host-ip-address* **Example:** `Controller(config-ext-nacl)# permit ip host 203.0.113.3 host 203.0.113.5`	Configures IP protocol traffic from a source address to a destination address.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.
Step 5	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 6	**class map** *acl-name* **Example:** `Controller(config)# class-map acl-a1`	Configures the class map name.
Step 7	**match access-group name** *access-list-name* **Example:** `Controller(config-cmap)# match access-group name a1`	Assigns the class map to an access group name.
Step 8	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring Table Maps

SUMMARY STEPS

1. **configure terminal**
2. table-map *table-map-name*
3. **map from** *from-value* **to** *to-value*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	table-map *table-map-name* **Example:** `Controller(config)# table-map mutate-dscp`	Create the table map.

Step 3	**map from** *from-value* **to** *to-value*
	
	Example:
	
	`Controller(config-tablemap)# map from 10 to 34`
	`Controller(config-tablemap)# map from 34 to 40`
	`Controller(config-tablemap)# map from 46 to 48`
		Map a to value to a from value.

Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Applying an SSID Policy

Before You Begin

You must have a service-policy map configured before applying it on an SSID.

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan-name*
3. **service-policy** [**client**] **qos** [**input** | **output**] *ssid-policy-name*
4. **end**

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	**wlan** *wlan-name* **Example:** `Controller# wlan test4`	Enters the WLAN configuration submode. The *wlan-name* is the profile name of the configured WLAN.
Step 3	**service-policy** [**client**] **qos** [**input** | **output**] *ssid-policy-name* **Example:** `Controller(config-wlan)# service-policy input policy-map-ssid`	Applies the policy. The following options are available: • **input**: Assigns the policy map to WLAN input traffic. • **output**: Assigns the policy map to WLAN output traffic. • **client**: Assigns the policy map to all clients on the WLAN.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

What to Do Next

Proceed to configure client policies.

Configuring QoS Policy for Multicast Traffic

Before You Begin

The following are the prerequisites for configuring a QoS policy for multicast traffic:

- You must have a multicast service policy configured.
- You must enable multicast-multicast mode before applying the policy.

SUMMARY STEPS

1. **configure terminal**
2. **ap capwap multicast service-policy output** *service -policy-name*
3. **end**

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	**ap capwap multicast service-policy output** *service -policy-name* **Example:** `Controller(config)#ap capwap multicast service-policy output service-policy-mcast`	Applies the configured multicast policy.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Related Topics

Wireless QoS Multicast, on page 306

Configuration Examples

Examples: Wireless QoS Policy Classified by Voice, Video, and Multicast Traffic

The following example provides a template for creating a port child policy for managing quality of service for voice and video traffic.

```
Policy-map port_child_policy
    Class voice (match dscp ef)
            Priority level 1
            Police  Multicast Policer
    Class video (match dscp af11)
            Priority level 2
            Police  Multicast Policer
    Class mcast-data  (match non-client-nrt)
            Bandwidth remaining ratio <>
    Class class-default  (NRT Data)
            Bandwidth remaining ratio <>
Policy-map parent_port
    Class class-default
         Shape average    Port Shaper
         Service-policy port_child_policy
```

Note Multicast Policer in the example above is not a keyword. It refers to the policing policy configured.

Two class maps with name voice and video are configured with DSCP assignments of 46 and 34. The voice traffic is assigned the priority of 1 and the video traffic is assigned the priority level 2 and is processed using Q0 and Q1. If your network receives multicast voice and video traffic, you can configure multicast policers. The non-client NRT data and NRT data are processed using the Q2 and Q3 queues.

Related Topics

Queuing in Wireless, on page 306

Port Policy Format, on page 304

Port, on page 301

Examples: SSID Policy

SSID Policy 1

The following is an example of an SSID policy for voice and video:

```
Policy-map enterprise-ssid-1
     Class  voice (match dscp ef)
          Priority level 1
         Police Unicast Policer
Class video (match dscp af11)
        Priority level 2
        Police Unicast Policer
Policy-map ssid-shaper
Class class-default  (NRT Data)
        queue-buffer 0
        shape average 100000000
        set wlan-user-priority dscp table dscp2up
        set dscp dscp table dscp2dscp
        service-policy  enterprise-ssid-1
```

SSID Policy 2

The following is an example of SSID policy configured with an average SSID shaping rate:

```
Policy-map enterprise-ssid-2
     Class  voice (match dscp af11)
          Priority level 1
         Police Unicast Policer
Class video (match dscp  ef)
        Priority level 2
        Police Unicast Policer
Policy-map ssid-shaper
Class class-default  (NRT Data)
       shape average 1000000000
       service-policy enterprise-ssid-2
       set wlan-user-priority dscp table dscp2up
       set dscp dscp table dscp2dscp
```

Related Topics

SSID, on page 302

Examples: Configuring Downstream BSSID Policy

To configure a downstream BSSID policy, you must first configure a port child policy with priority level queuing.

Configuring a User-Defined Port Child Policy

The following is an example of configuring a user-defined port child policy:

```
policy-map port_child_policy
   class voice
     priority level 1 20000

   class video
     priority level 2 10000
```

```
   class non-client-nrt-class
     bandwidth remaining ratio 10

   class class-default
     bandwidth remaining ratio 15
```

Configuring Downstream BSSID Policy

The following configuration example displays how to configure a downstream BSSID policy:

```
policy-map bssid-policer
   class class-default
   shape average 30000000
set dscp dscp table dscp2dscp
set wlan user-priority dscp table dscp2up
service-policy ssid_child_qos
```

The SSID child QoS policy may be defined as below:

```
Policy Map ssid-child_qos
    Class voice
      priority level 1
      police cir 5m
      admit cac wmm-tspec
           UP 6,7  / tells WCM allow 'voice' TSPEC\SIP snoop for this ssid
            rate 4000 / must be police rate value is in kbps)
   Class video
      priority level 2
      police cir 60000
```

Related Topics

SSID, on page 302

Examples: Client Policies

The following example shows a default client policy in the downstream direction. Any incoming traffic contains the user-priority as 0:

```
Policy-map client-def-down
  class class-default
      set wlan user-priority 0
```

The following example shows the default client policy in the upstream direction. Any traffic that is sent to the wired network from wireless network will result in the DSCP value being set to 0.

```
Policy-map client-def-up
   class class-default
    set dscp 0
```

The following examples shows client policies that are generated automatically and applied to the client when the client authenticates to a profile in AAA with a QoS-level attribute configured.

```
Policy Map platinum-WMM
Class voice-plat
  set wlan user-priority 6
 Class video-plat
 set wlan user-priority 4
Class class-default
  set wlan user-priority 0

Policy Map gold-WMM
Class voice-gold
  set wlan user-priority 4
```

```
 Class video-gold
 set wlan user-priority 4
Class class-default
  set wlan user-priority 0
```

The following is an example of client precious metal policies:

```
Policy Map platinum
  set wlan user-priority 6
```

Any traffic matching class voice1 the user priority is set to a pre-defined value. The class can be set to assign a DSCP or ACL.

```
Policy Map client1-down
Class voice1     //match dscp, cos
  set wlan user-priority <>
Class voice2    //match acl
   set wlan user-priority <>
Class voice3
   set wlan user-priority <>
Class class-default
  set wlan user-priority 0
```

The following is an example of a client policy based on AAA and TCLAS:

```
Policy Map client2-down[ AAA+ TCLAS pol example]
Class      voice\\match dscp
      police <>
      set <>
Class class-default
    set <>
Class voice1|| voice2 [match acls]
      police <>
      class voice1
        set <>
      class voice2
        set <>
```

The following is an example of a client policy for voice and video for traffic in the downstream direction:

```
Policy Map client3-down
    class voice \\match dscp, cos
          police X
     class video
         police Y
    class default
        police Z
```

The following is an example of a client policy for voice and video for traffic in the upstream direction using policing:

```
Policy Map client1-up
    class voice     \\match dscp, up, cos
      police X
    class video
     police Y
  class class-default
    police Z
```

The following is an example of a client policy for voice and video based on DSCP:

```
Policy Map client2-up
    class voice     \\match dscp, up, cos
set dscp <>
    class video
     set dscp <>
  class class-default
    set dscp <>
```

Related Topics

Client, on page 302

Additional References

Related Documents

Related Topic	Document Title
QoS Command Reference	*QoS Command Reference (Cisco WLC 5700 Series)*
Mobility Configuration Guide	*Mobility Configuration Guide, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*
Quality of Service Solutions Configuration Guide (Cisco IOS Software)	*Quality of Service Solutions Configuration Guide Library, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*

MIBs

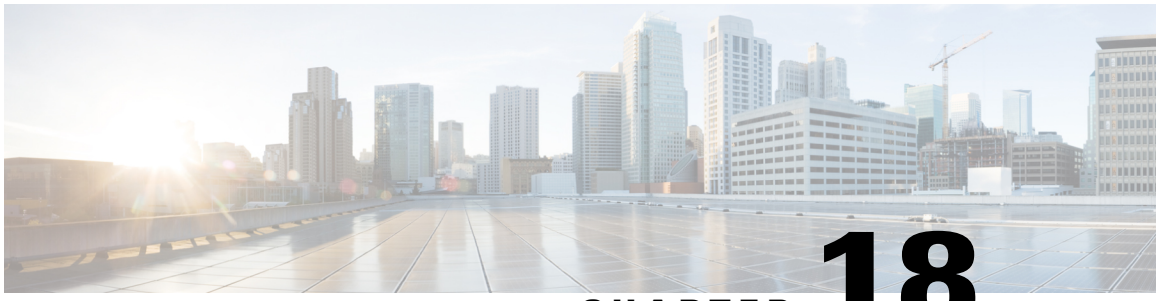
MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

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

PART III

Interface



CHAPTER 18

Configuring Interfaces

This chapter contains the following topics:

Configuring Interfaces

This module contains the following topics:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. 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.

Pre-requisites for Configuring Interfaces

You can define the wireless management, AP-manager, virtual, and management interface parameters using the Startup Wizard. However, you can display and configure interface parameters through either the GUI or CLI after the controller is running.

For Cisco 5700 Series Controllers in a non-link-aggregation (non-LAG) configuration, the management interface must be on a different VLAN than any dynamic AP-manager interface. Otherwise, the management interface cannot fail over to the port that the AP-manager is on.

To configure interfaces, you must configure the default gateway, router, and the IP route using the following commands:

- **ip default-gateway** 154.4.0.1
- **default-router** 154.51.0.1
- **ip route** 0.0.0.0 0.0.0.0 154.4.0.1

Restrictions for Configuring Interfaces

Information About Interfaces

An interface is a logical entity on the controller. An interface has multiple parameters associated with it, including an IP address, default gateway, VLAN identifier, and DHCP server. The following interfaces available on the controller:

- Wireless Management Interface
- AP Manager Interface
- Dynamic Interface

The wireless management interface is used for access point join functions, mobility, RRM , and also used for peer connections (MC - MC connections) and MC to MA connections.

Typically, you define the management, AP-manager, virtual, and service-port interface parameters using the Startup Wizard. However, you can display and configure interface parameters through either the GUI or CLI after the controller is running.

Interface Types

This section describes the different types of interfaces supported by the controller. 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 controller 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.

Switch Ports

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

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

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

Access Ports

An access port belongs to and carries the traffic of only one VLAN (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.

Two types of access ports are supported:

- 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 controller are assigned to a VLAN by a VLAN Membership Policy Server (VMPS). The VMPS can be a Catalyst 6500 series switch; the controller 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.

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 feature set supports static routing and the Routing Information Protocol (RIP). For full Layer 3 routing or for fallback bridging, you must enable the IP services feature set 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. Only one SVI can be associated with a VLAN, but you need to configure an SVI for a VLAN only when you wish to route between VLANs, to fallback-bridge nonroutable protocols between VLANs, or to provide IP host connectivity to the controller. By default, an SVI is created for the default VLAN (VLAN 1) to permit remote controller administration. Additional SVIs must be explicitly configured.

Note You cannot delete interface VLAN 1.

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

Although the switch stack or controller 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.

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.

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

SVIs support routing protocols and bridging configurations.

Note The IP base feature set supports static routing and RIP. For more advanced routing or for fallback bridging, enable the IP services feature set on the standalone switch or the active switch. For information about using the software activation feature to install a software license for a specific feature set, see the *Cisco IOS Software Activation* document.

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 controller

- 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, such as routing black holes.

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 controllers or between controllers 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.

10-Gigabit Ethernet Interfaces

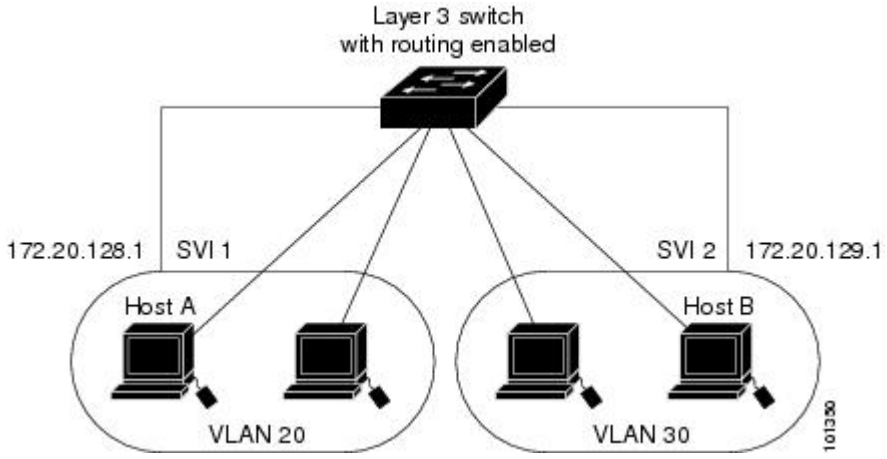
A 10-Gigabit Ethernet interface operates only in full-duplex mode. The interface can be configured as a switched or routed port.

For more information about the Cisco TwinGig Converter Module, see the controller hardware installation guide and your transceiver module documentation.

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 7: Connecting VLANs with the Switch



When the IP services feature set is running on the switch or the active switch, the switch uses two methods to forward traffic between interfaces: routing and fallback bridging. If the IP base feature set is on the switch or the active switch, only basic routing (static routing and RIP) is supported. Whenever possible, to maintain high performance, forwarding is done by the switch hardware. However, only IPv4 packets with Ethernet II encapsulation are routed in hardware. Non-IP traffic and traffic with other encapsulation methods are fallback-bridged by hardware.

- The routing function can be enabled on all SVIs and routed ports. The switch routes only IP traffic. When IP routing protocol parameters and address configuration are added to an SVI or routed port, any IP traffic received from these ports is routed.
- Fallback bridging forwards traffic that the switch does not route or traffic belonging to a nonroutable protocol, such as DECnet. Fallback bridging connects multiple VLANs into one bridge domain by bridging between two or more SVIs or routed ports. When configuring fallback bridging, you assign SVIs or routed ports to bridge groups with each SVI or routed port assigned to only one bridge group. All interfaces in the same group belong to the same bridge domain.

Interface Configuration Mode

The controller supports these interface types:

- Physical ports—controller 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, stack member number (only stacking-capable switches), module number, and controller port number, and enter interface configuration mode.

- Type—Gigabit Ethernet (gigabitethernet or gi) for 10/100/1000 Mb/s Ethernet ports, 10-Gigabit Ethernet (tengigabitethernet or te) for 10,000 Mb/s, 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 switch number range is 1 to 9 and 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: switch (downlink) ports are 0, and uplink ports are 1.
- 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.

  On a controller with SFP uplink ports, the module number is 1 and the port numbers restart. For example, if the switch has 24 10/100/1000 ports, the SFP module ports are gigabitethernet1/1/1 through gigabitethernet1/1/4 or tengigabitethernet1/1/1 through tengigabitethernet1/1/4.

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 controller, enter this command:

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

- To configure 10-Gigabit Ethernet port 1 on a standalone controller, enter this command:

```
Controller(config)# interface tengigabitethernet1/0/1
```

- To configure 10-Gigabit Ethernet port on stack member 3, enter this command:

```
Controller(config)# interface tengigabitethernet3/0/1
```

- To configure the first SFP module (uplink) port on a standalone controller, enter this command:

```
Controller(config)# interface gigabitethernet1/1/1
```

Default Ethernet Interface Configuration

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.

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

Table 42: 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 (Layer 2 interfaces only).
Native VLAN (for IEEE 802.1Q trunks)	VLAN 1 (Layer 2 interfaces only).
VLAN trunking	Switchport mode dynamic auto (supports DTP) (Layer 2 interfaces only).
Port enable state	All ports are enabled.
Port description	None defined.
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) (Layer 2 interfaces only).
Broadcast, multicast, and unicast storm control	Disabled.
Protected port	Disabled (Layer 2 interfaces only).
Port security	Disabled (Layer 2 interfaces only).

Feature	Default Setting
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).

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

Configuring Interfaces

This module lists the generic steps used to configure any interface on the controller. You must use the following steps to configure interfaces on the controller:

Before You Begin

-

SUMMARY STEPS

1. **configure terminal**
2. **global configuration**
3. **interface**
4. **show interface summary**
5. **show interface detail management**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:**	Enables you to enter configure terminal configured mode at the privileged prompt.
Step 2	**global configuration** **Example:** global configuration	Identify interface details, for example the interface type, connector, and so on and enter global configuration mode. Enables you to identify the interface and enter global configuration mode.
Step 3	**interface** **Example:**	Follow each **interface** command with the interface configuration commands that the interface requires. The commands that you enter define the protocols and applications that will run on the configuration commands. Interfaces configured in

	Command or Action	Purpose
		a range must be the same type and must be configured with the same feature options. The commands are collected and applied to the interface when you enter another **interface** command or enter end to return to privileged EXEC mode. Enables you to configure the supported interfaces on the controller.
Step 4	**show interface summary** **Example:**	Verify the status of the configured interface using the **show interface summary**. Enables you to view the status of the configured interface.
Step 5	**show interface detail management** **Example:**	Verify the status of the configured interface using the **show interface detail management**. Enables you to view the status of the configured interface.

Adding a Description for an Interface

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **description** *string*
4. **end**
5. **show interfaces** *interface-id* **description**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet1/0/2`	Specifies the interface for which you are adding a description, and enter interface configuration mode.

	Command or Action	Purpose
Step 3	**description** *string* **Example:** `Controller(config-if)# description Connects to Marketing`	Adds a description (up to 240 characters) for an interface.
Step 4	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show interfaces** *interface-id* **description**	Verifies your entry.

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:

```
Controller# configure terminal
Controller(config)# interface range gigabitethernet1/0/1 - 4
Controller(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:

```
Controller# configure terminal
Controller(config)# interface range gigabitethernet1/0/1 - 3 , tengigabitethernet1/0/1 - 2
Controller(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:

```
Controller# configure terminal
Controller(config)# define interface-range enet_list gigabitethernet1/0/1 - 2
Controller(config)# end
Controller# 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*:

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

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

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

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

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

Configuring Interfaces Procedure

These general instructions apply to all interface configuration processes.

DETAILED STEPS

	Command or Action	Purpose

Step 1	Enter the **configure terminal** command at the privileged EXEC prompt:
	
	Example:
	
	`Controller# configure terminal`
	`Enter configuration commands, one per line. End with CNTL/Z.`
	`Controller(config)#`	

Step 2	Enter the **interface** global configuration command. Identify the interface type, the switch number (only on stacking-capable switches), and the number of the connector. In this example, Gigabit Ethernet port 1 on switch 1 is selected:
	
	Example:
	
	`Controller(config)# interface gigabitethernet1/0/1`
	`Controller(config-if)#`
		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 3	Follow each **interface** command with the interface configuration commands that the interface requires. The commands that you enter define the protocols and applications that will run on the	You can also configure a range of interfaces by using the **interface range** or **interface range macro** global configuration commands. Interfaces configured in a range must be the same type and must be configured with the same feature options.

	Command or Action	Purpose
	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.	
Step 4	After you configure an interface, verify its status by using the **show** privileged EXEC commands. **Example:**	Enter the **show interfaces** privileged EXEC command to see a list of all interfaces on or configured for the switch. A report is provided for each interface that the device supports or for the specified interface.

Configuring Layer 3 Interfaces

SUMMARY STEPS

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

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**interface** {**gigabitethernet** *interface-id*} | {**vlan** *vlan-id*} | {**port-channel** *port-channel-number*} **Example:** `Controller(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 3	**no switchport** **Example:** `Controller(config-if)# no switchport`	For physical ports only, enters Layer 3 mode.
Step 4	**ip address** *ip_address subnet_mask* **Example:** `Controller(config-if)# ip address 192.20.135.21 255.255.255.0`	Configures the IP address and IP subnet.
Step 5	**no shutdown** **Example:** `Controller(config-if)# no shutdown`	Enables the interface.
Step 6	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.
Step 7	**show interfaces** [*interface-id*]	Verifies the configuration.

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. **configure terminal**
2. **interface** {**vlan** *vlan-id*} | {**gigabitethernet** *interface-id*} | {**port-channel** *port-channel-number*}
3. **shutdown**
4. **no shutdown**
5. **end**

DETAILED STEPS

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

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.

This table lists some of the available interface monitoring commands.

Table 43: Show Commands for Interfaces

Command	Purpose
show interfaces [*interface-id*]	Displays the status and configuration of all interfaces or a specific interface.
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.
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 44: 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.

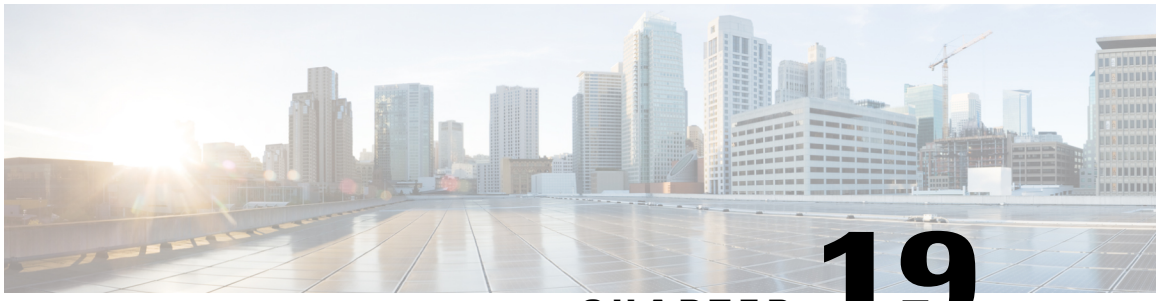
Note The **clear counters** privileged EXEC command does not clear counters retrieved by using Simple Network Management Protocol (SNMP), but only those seen with the **show interface** privileged EXEC command.

Viewing Wireless Interfaces in the Controller GUI

You can view the wireless interfaces available in the controller by choosing **Monitor > Controller > System > Wireless Interface.** in the controller GUI. The following details of the wireless interface page are displayed.

Parameter	Description
Interface Type	Display Only—Dynamic—Operator-defined interfaces. Values are as follows: • Static—Wireless Management. • AP-Manager. • Service-Port— The Ten Gigabit Ethernet port located in the controller's port at the back. • Virtual interfaces.

Parameter	Description
Interface Name	Name of the interface. Values are as follows: • Management—802.11 Distribution System wired network. • Service-port—System Service interface. • Virtual—Loopback interface for the GUI to work. This is available in the controller by default. You need not explicitly configure this interface. • AP-manager—Can be on the same subnet as the management IP address, but must have a different IP address than the management interface. • *name*—Operator-Defined Interface assignment, without any spaces.
IP Address	IP address of the Controller and its distribution port.
IP Netmask	Destination subnet mask.
MAC Address	MAC address of the interface.
VLAN ID	Virtual LAN assignment of the interface.



CHAPTER 19

Configuring Management Interfaces

This module lists the following topics:

Configuring the Management Interface

This module contains the following topics:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. 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 the Management Interface

The management interface is the default interface for in-band management of the controller and connectivity to enterprise services such as AAA servers. It is also used for communications between the controller and access points. The management interface has the only consistently “pingable” in-band interface IP address on the controller. You can access the GUI of the controller by entering the management interface IP address of the controller in the address field of either Internet Explorer or Mozilla Firefox browser.

For CAPWAP, the controller requires one management interface to control all inter-controller communications and one AP-manager interface to control all controller-to-access point communications, regardless of the number of ports.

Note To prevent or block a wired or wireless client from accessing the management network on a controller (from the wireless client dynamic interface or VLAN), the network administrator must ensure that only authorized clients gain access to the management network through proper CPU ACLs, or use a firewall between the client dynamic interface and the management network.

Pre-requisites for Configuring Management Interfaces

The pre-requisites for configuring the controller's management interfaces follow:

- For Cisco 5700 Series Controllers in a non-link-aggregation (non-LAG) configuration, the management interface must be on a different VLAN than any dynamic AP-manager interface. Otherwise, the management interface cannot fail over to the port that the AP-manager is on.
- If the service port is in use, the management interface must be on a different supernet from the service-port interface.
- To prevent or block a wired or wireless client from accessing the management network on a controller (from the wireless client dynamic interface or VLAN), the network administrator must ensure that only authorized clients gain access to the management network through proper CPU ACLs, or use a firewall between the client dynamic interface and the management network.

Restrictions for Configuring Management Interfaces

The following are the restrictions for configuring the controller's management interface:

- Do not map a guest WLAN to the management interface. If the EoIP tunnel breaks, the client could obtain an IP and be placed on the management subnet.
- Do not configure wired clients in the same VLAN or subnet of the service port of the controller on the network. If you configure wired clients on the same subnet or VLAN as the service port, it is not possible to access the management interface of the controller.

Configuring the Management Interface using the CLI

Before You Begin

You must use the following steps to configure management interfaces on the controller. You can also use these steps to configure the AP manager interfaces on the controller. These general instructions apply to all management interfaces.

SUMMARY STEPS

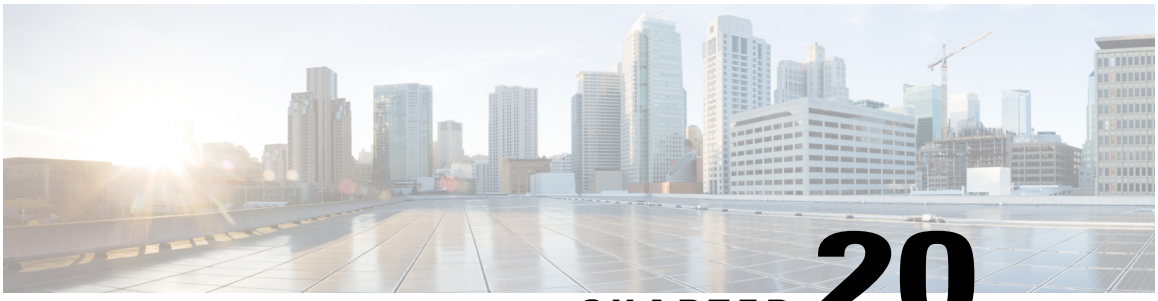
1. **config terminal**
2. **show ip interface brief**
3. **wireless management interface vlan vlanID**
4. **show wireless interface summary**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**config terminal**	Enters global configuration mode.
Step 2	**show ip interface brief** **Example:**	Displays all the interfaces in the controller.
Step 3	**wireless management interface vlan vlanID**	Creates a management interface by providing the values for the VLAN (VLAN identifier).
Step 4	**show wireless interface summary**	Displays all the wireless interfaces in the controller.
Step 5	**end** **Example:**	Returns to EXEC mode.

Configuring the Management Interface

This module contains the following topics:



CHAPTER 20

Configuring AP Manager Interfaces

This module lists the following sections:

Configuring AP Manager Interfaces

Configuring AP Manager Interface

This module contains the following topics:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. 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.

Pre-requisites for Configuring Access Point Management Interface

Restrictions for Configuring AP Manager Interfaces

- The MAC address of the management interface and the AP-manager interface is the same as the base LAG MAC address.
- If only one distribution system port can be used, you should use distribution system port 1.
-
-
-

•

Information the About AP-Manager Interface

A controller has one or more AP-manager interfaces, which are used for all Layer 3 communications between the controller and lightweight access points after the access points have joined the controller. The AP-manager IP address is used as the tunnel source for CAPWAP packets from the controller to the access point and as the destination for CAPWAP packets from the access point to the controller.

Note The Controller does not support transmitting the jumbo frames. To avoid having the controller transmit CAPWAP packets to the AP that will necessitate fragmentation and reassembly, reduce MTU/MSS on the client side.

The AP-manager interface communicates through any distribution system port by listening across the Layer 3 network for access point CAPWAP or LWAPP join messages to associate and communicate with as many lightweight access points as possible.

Configuring AP Join in an AP Manager Interface

Before You Begin

The controller sends the access point a CAPWAP join response allowing the access point to join the controller. When the access point joins the controller, the controller manages its configuration, firmware, control and data transactions.

When an access point performs a reboot or is disconnected from the controller, the join statistics for an access point is maintained from the controller. But this statistics are lost when the controller performs a reboot or disconnects.

SUMMARY STEPS

1. **conf t**
2. **int portnumber**
3. **switchport mode access**
4. **switchport access vlan name**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**conf t**	Enters global configuration mode.
Step 2	**int portnumber** **Example:**	The interface name. Example of an interface name is tengig1/0/1.

	Command or Action	Purpose
Step 3	**switchport mode access** **Example:**	Specifies that the access interface is a management interface.
Step 4	**switchport access vlan name** **Example:**	Enables the access point to receive the IP address and join the specified VLAN.
Step 5	**end** **Example:**	Returns to EXEC mode.

Viewing Configured Access Point Join Management Interfaces

Before You Begin

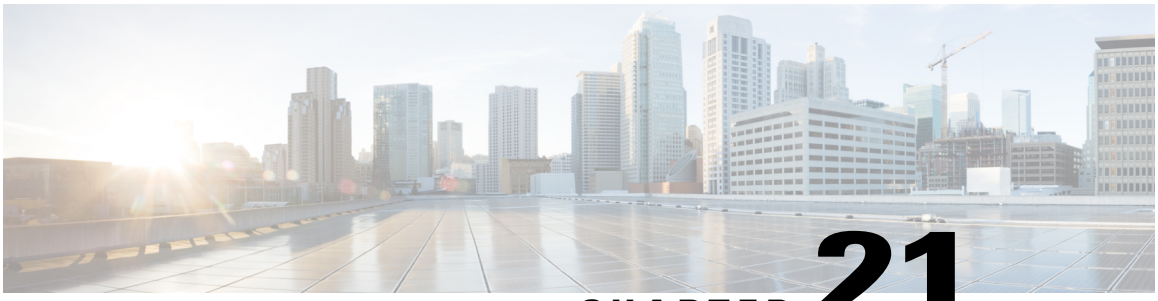
You can view the access point join interfaces configured in the controller using the following steps:

SUMMARY STEPS

1. show ap summary
2. show ap name
3. show ap name apname config general
4. show wireless interface summary

DETAILED STEPS

	Command or Action	Purpose
Step 1	show ap summary	Displays the summary of all the access points configured in the interface.
Step 2	show ap name	Displays the summary of all the access points configured in the interface.
Step 3	show ap name apname config general	Displays all the general parameter configuration for the access point.
Step 4	show wireless interface summary	Displays all the wireless- management and AP manager interfaces in the controller.



CHAPTER 21

Configuring Dynamic Interfaces

This module lists the following sections:

Configuring Dynamic Interfaces

This module contains the following topics:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. 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.

Pre - requisites for Configuring Dynamic Interfaces

While configuring on the dynamic interface of the controller, you must ensure the following:

-
- You must use tagged VLANs for dynamic interfaces.

Restrictions for Configuring Dynamic Interfaces

The following restrictions apply for configuring the dynamic interfaces on the controller:

- You must not configure a dynamic interface in the same subnetwork as a server that is reachable by the controller CPU, such as a RADIUS server, as it might cause asymmetric routing issues.
-
-
-

Information About Dynamic AP Management

A dynamic interface is created as a WLAN interface by default. However, any dynamic interface can be configured as an AP-manager interface, with one AP-manager interface allowed per physical port. A dynamic interface with the Dynamic AP Management option enabled is used as the tunnel source for packets from the controller to the access point and as the destination for CAPWAP packets from the access point to the controller. The dynamic interfaces for AP management must have a unique IP address and are usually configured on the same subnet as the management interface.

Note If link aggregation (LAG) is enabled, there can be only one AP-manager interface.

We recommend having a separate dynamic AP-manager interface per controller port.

Configuring Dynamic Interfaces

Before You Begin

You must create the Layer 2 interface that you plan to use in the WLAN.

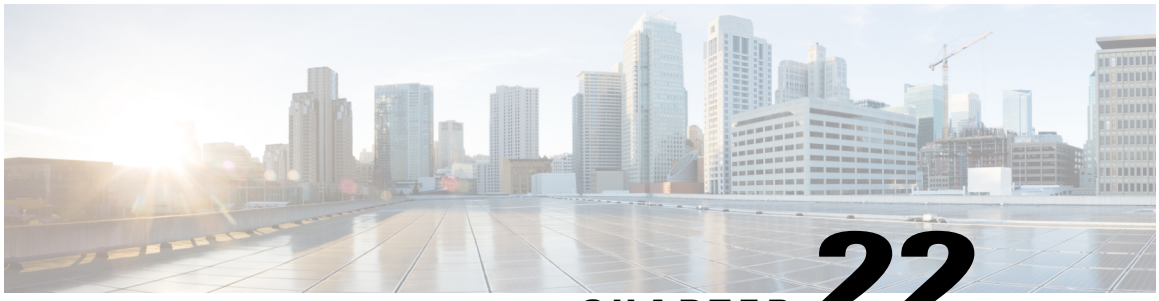
You can configure the dynamic interface using the following steps:

SUMMARY STEPS

1. **show VLAN**
2. **show int VLAN**
3. **configure terminal**
4. **configure WLAN wlan number wlan name client vlan vlan name**
5. **show vlan**
6. **end**
7. **Show WLAN summary**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show VLAN**	Displays all the VLANs.
Step 2	**show int VLAN**	Displays all the VLAN interfaces.
Step 3	**configure terminal**	Enters global configuration mode.
Step 4	**configure WLAN wlan number wlan name client vlan vlan name**	Configures the WLAN.
Step 5	**show vlan**	Displays all the VLANs in the WLAN.
Step 6	**end**	Exit configuration mode.
Step 7	**Show WLAN summary**	Displays a summary of all the configured VLANs.



CHAPTER 22

Configuring Multiple AP Manager Interfaces

This module lists the following sections:

Configuring Multiple AP Manager Interfaces

This section has the following topics:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. 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.

Pre-requisites For Configuring AP Manager Interfaces

Restrictions for Configuring Multiple AP Manager Interfaces

The following restrictions apply while configuring the multiple AP manager interfaces in the controller:

- You must assign an AP-manager interface to each port on the controller.

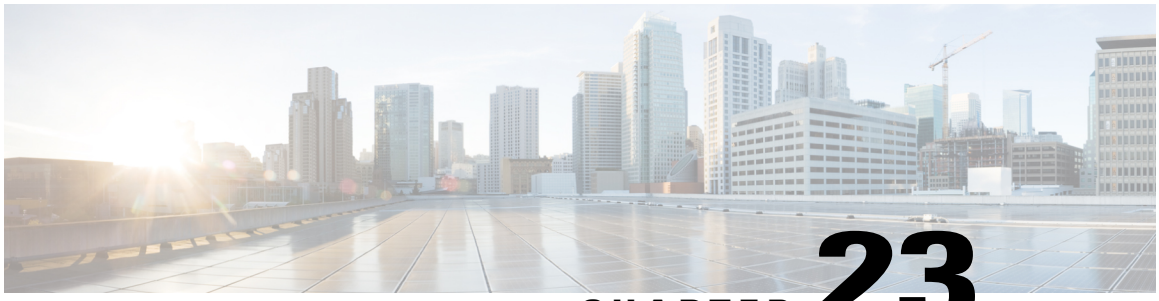
- Before implementing multiple AP-manager interfaces, you should consider how they would impact your controller's port redundancy.
-
-
-

Information About Multiple AP-Manager Interfaces

When you create two or more AP-manager interfaces, each one is mapped to a different port. The ports should be configured in sequential order so that AP-manager interface 2 is on port 2, AP-manager interface 3 is on port 3, and AP-manager interface 4 is on port 4.

Before an access point joins a controller, it sends out a discovery request. From the discovery response that it receives, the access point can tell the number of AP-manager interfaces on the controller and the number of access points on each AP-manager interface. The access point generally joins the AP-manager with the least number of access points. In this way, the access point load is dynamically distributed across the multiple AP-manager interfaces.

Note Access points may not be distributed completely evenly across all of the AP-manager interfaces, but a certain level of load balancing occurs.



CHAPTER 23

Configuring Interface Groups

This module lists the following sections:

Configuring Interface Groups

This module contains following topics:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. 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 Interface Groups

Interface groups are logical groups of interfaces. Interface groups facilitate user configuration where the same interface group can be configured on multiple WLANs or while overriding a WLAN interface per AP group. An interface group can exclusively contain either quarantine or nonquarantine interfaces. An interface can be part of multiple interface groups.

A WLAN can be associated with an interface or interface group. The interface group name and the interface name cannot be the same.

This feature also enables you to associate a client to specific subnets based on the foreign controller that they are connected to. The anchor controller WLAN can be configured to maintain a mapping between foreign controller MAC and a specific interface or interface group (Foreign maps) as needed. If this mapping is not configured, clients on that foreign controller gets VLANs associated in a round robin fashion from interface group configured on WLAN.

Creating Interface Groups

Before You Begin

You must create the interface groups using the following commands after you configure the terminal.

SUMMARY STEPS

1. **vlan group groupname vlan-list 1-256**
2. **wlan wlanname 1 wlanname**
3. **client vlan vlangrp1**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**vlan group groupname vlan-list 1-256**	Creates a VLAN group with the given group name and adds all the VLANs listed in the command. The recommended number of VLANs in a group is 128.
Step 2	**wlan wlanname 1 wlanname**	Enables the WLAN to map a VLAN group.
Step 3	**client vlan vlangrp1**	Maps the VLAN group to the WLAN.

Adding a VLAN Group to a WLAN

SUMMARY STEPS

1. **conf t**
2. **wlan wlanname 1 wlanname**
3. **client vlan vlangrp1**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**conf t** **Example:**	Enters global configuration mode.
Step 2	**wlan wlanname 1 wlanname** **Example:**	Enables the WLAN to map a VLAN group.
Step 3	**client vlan vlangrp1**	Maps the VLAN group to the WLAN.
Step 4	**end** **Example:**	Returns back to exec mode.

Configuring the Trunk Port

Before You Begin

You must configure the VLAN after configuring the controller port as a trunk port. We recommend that you configure the trunk port first and then associate the VLANs to the trunk port.

SUMMARY STEPS

1. **show wireless interface summary**
2. **show run int te1/0/1**
3. **interface TenGigabitEthernet1/0/1**
4. **switchport trunk allowed vlan 1-10**
5. **switchport mode trunk**
6. **nmsp attachment suppress**
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show wireless interface summary** **Example:**	Displays all the wireless interfaces in the controller.
Step 2	**show run int te1/0/1** **Example:**	Displays the running configuration available in the controller.

	Command or Action	Purpose
Step 3	**interface TenGigabitEthernet1/0/1** **Example:** interface TenGigabitEthernet1/0/1	Configures the 10-Gigabit Ethernet.
Step 4	**switchport trunk allowed vlan 1-10**	Configures the trunk port.
Step 5	**switchport mode trunk**	Configures the switch port mode as a trunk.
Step 6	**nmsp attachment suppress**	
Step 7	**end**	Returns to EXEC mode.

Configuring VLAN Interfaces using the GUI

Before You Begin

You can configure VLANs in the Cisco 5700 Wireless LAN Controller using the graphical user interface (GUI). To do this, you must follow the steps defined in this module in the GUI.

SUMMARY STEPS

1. Choose **Configuration > Controller > System > Interfaces** > **VLAN Summary.**
2. Click the VLAN ID field in the table to view the details of the selected VLAN.
3. Enter the values.
4. Click **Apply**.
5. To create a new VLAN, click **New**.
6. To delete a VLAN, check the check box in the VLAN summary page, and click **Remove**.

DETAILED STEPS

Step 1 Choose **Configuration > Controller > System > Interfaces** > **VLAN Summary.**
This page displays all the VLANs and details of the VLANs in the controller.

Step 2 Click the VLAN ID field in the table to view the details of the selected VLAN.
The Edit VLAN details page appears.

Step 3 Enter the values.

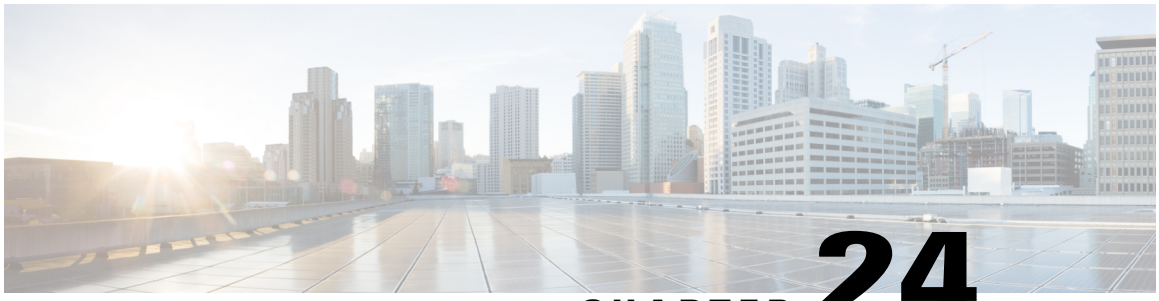
Step 4 Click **Apply**.

Step 5 To create a new VLAN, click **New**.

Step 6 To delete a VLAN, check the check box in the VLAN summary page, and click **Remove**.

PART IV

VLAN



CHAPTER 24

Configuring VTP

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 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 controllers and have those changes automatically communicated to all the other controllers in the network. Without VTP, you cannot send information about VLANs to other controllers.

VTP is designed to work in an environment where updates are made on a single controller and are sent through VTP to other controllers in the domain. It does not work well in a situation where multiple updates to the

VLAN database occur simultaneously on controllers in the same domain, which would result in an inconsistency in the VLAN database.

The controller supports a total of 4094 VLANs. However, the number of routed ports, SVIs, and other configured features affects the usage of the controller hardware. If the controller is notified by VTP of a new VLAN and the controller 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.

Because trunk ports send and receive VTP advertisements, you must ensure that at least one trunk port is configured on the controller and that this trunk port is connected to the trunk port of another controller. Otherwise, the controller cannot receive any VTP advertisements.

Related Topics

VTP Advertisements, on page 368

Adding a VTP Client Switch to a VTP Domain, on page 383

VTP Domain, on page 366

VTP Modes, on page 367

Restrictions for VTP

Caution Before adding a VTP client controller to a VTP domain, always verify that its VTP configuration revision number is lower than the configuration revision number of the other controllers in the VTP domain. Controllers in a VTP domain always use the VLAN configuration of the controller with the highest VTP configuration revision number. If you add a controller that has a revision number higher than the revision number in the VTP domain, it can erase all VLAN information from the VTP server and VTP domain.

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 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.

VTP Domain

A VTP domain (also called a VLAN management domain) consists of one controller or several interconnected controllers or controller stacks under the same administrative responsibility sharing the same VTP domain

name. A controller can be in only one VTP domain. You make global VLAN configuration changes for the domain.

By default, the controller 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 controller receives a VTP advertisement over a trunk link, it inherits the management domain name and the VTP configuration revision number. The controller 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 controllers 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 controller for VTP transparent mode, you can create and modify VLANs, but the changes are not sent to other controllers in the domain, and they affect only the individual controller. However, configuration changes made when the controller is in this mode are saved in the controller running configuration and can be saved to the controller startup configuration file.

Related Topics

Adding a VTP Client Switch to a VTP Domain, on page 383

Prerequisites for VTP, on page 365

VTP Modes

Table 45: 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 controllers in the same VTP domain and synchronize their VLAN configurations with other controllers 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 controller detects a failure while writing a configuration to NVRAM, VTP mode automatically changes from server mode to client mode. If this happens, the controller 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 controller 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 controllers do not participate in VTP. A VTP transparent controller 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 controllers do forward VTP advertisements that they receive from other controllers through their trunk interfaces. You can create, modify, and delete VLANs on a controller in VTP transparent mode. When the controller is in VTP transparent mode, the VTP and VLAN configurations are saved in NVRAM, but they are not advertised to other controllers . In this mode, VTP mode and domain name are saved in the controller running configuration, and you can save this information in the controller startup configuration file by using the **copy running-config startup-config** privileged EXEC command.
VTP off	A controller in VTP off mode functions in the same manner as a VTP transparent controller, except that it does not forward VTP advertisements on trunks.

Related Topics

Prerequisites for VTP, on page 365

Configuring VTP Mode, on page 375

VTP Advertisements

Each controller in the VTP domain sends periodic global configuration advertisements from each trunk port to a reserved multicast address. Neighboring controllers receive these advertisements and update their VTP and VLAN configurations as necessary.

VTP advertisements distribute this global domain information:

- VTP domain name
- VTP configuration revision number
- Update identity and update timestamp
- MD5 digest VLAN configuration, including maximum transmission unit (MTU) size for each VLAN
- Frame format

VTP advertisements distribute this VLAN information for each configured VLAN:

- VLAN IDs (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.

Related Topics

Prerequisites for VTP, on page 365

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 controller is operating in VTP server mode.
- Version-Dependent Transparent Mode—In VTP version 1, a VTP transparent controller 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 controller 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.

Related Topics

Enabling the VTP Version, on page 378

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.

- 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 controller

- The option to turn VTP on or off on a per-trunk (per-port) basis—You can enable or disable VTP per port by entering the [**no**] **vtp** interface configuration command. When you disable VTP on trunking ports, all VTP instances for that port are disabled. You cannot set VTP to *off* for the MST database and *on* for the VLAN database on the same port.

  When you globally set VTP mode to off, it applies to all the trunking ports in the system. However, you can specify on or off on a per-VTP instance basis. For example, you can configure the controller as a VTP server for the VLAN database but with VTP *off* for the MST database.

Related Topics

Enabling the VTP Version, on page 378

VTP Pruning

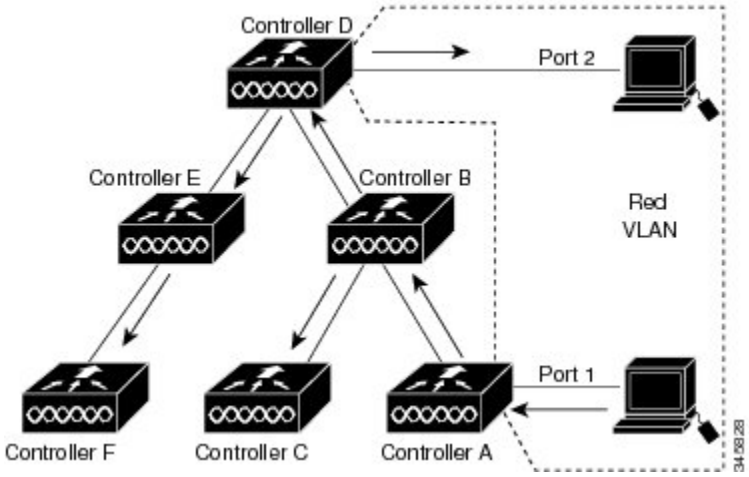
VTP pruning increases network available bandwidth by restricting flooded traffic to those trunk links that the traffic must use to reach the destination devices. Without VTP pruning, a controller floods broadcast, multicast, and unknown unicast traffic across all trunk links within a VTP domain even though receiving controllers 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 controller trunk ports. If the VLANs are configured as pruning-ineligible, the flooding continues. VTP pruning is supported in all VTP versions.

VTP pruning is disabled in the switched network. Port 1 on Controller A and Port 2 on Controller D are assigned to the Red VLAN. If a broadcast is sent from the host connected to Controller A, Controller A floods

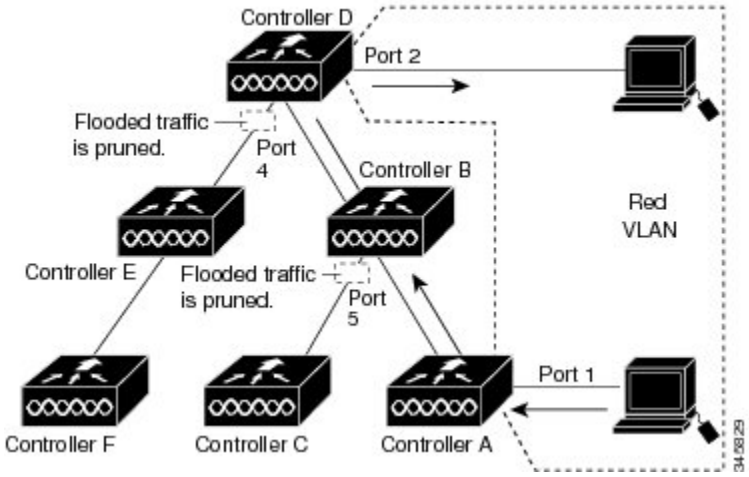
the broadcast and every controller in the network receives it, even though Controllers C, E, and F have no ports in the Red VLAN.

Figure 8: Flooding Traffic Without VTP Pruning



VTP pruning is enabled in the switched network. The broadcast traffic from Controller A is not forwarded to Controllers C, E, and F because traffic for the Red VLAN has been pruned on the links shown (Port 5 on Controller B and Port 4 on Controller D).

Figure 9: Optimized Flooded Traffic VTP Pruning



Enabling VTP pruning on a VTP server enables pruning for the entire management domain. Making VLANs pruning-eligible or pruning-ineligible affects pruning eligibility for those VLANs on that trunk only (not on all controllers 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.

Related Topics

Enabling VTP Pruning, on page 380

VTP Configuration Guidelines

VTP Configuration Requirements

When you configure VTP, you must configure a trunk port so that the controller can send and receive VTP advertisements to and from other controllers in the domain.

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 controller running configuration file, and you can save it in the controller 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 controller resets.

When you save VTP information in the controller startup configuration file and reboot the controller, the controller 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.

Related Topics

Configuring VTP on a Per-Port Basis, on page 381

Configuring a VTP Version 3 Primary Server, on page 378

Domain Names for Configuring VTP

When configuring VTP for the first time, you must always assign a domain name. You must configure all controllers in the VTP domain with the same domain name. Controllers in VTP transparent mode do not exchange VTP messages with other controllers, and you do not need to configure a VTP domain name for them.

Note If the NVRAM and DRAM storage is sufficient, all controllers in a VTP domain should be in VTP server mode.

Caution Do not configure a VTP domain if all controllers 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 controller in the VTP domain for VTP server mode.

Related Topics

Adding a VTP Client Switch to a VTP Domain, on page 383

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 controllers must share the same password and you must configure the password on each controller in the management domain. Controllers without a password or with the wrong password reject VTP advertisements.

If you configure a VTP password for a domain, a controller that is booted without a VTP configuration does not accept VTP advertisements until you configure it with the correct password. After the configuration, the controller accepts the next VTP advertisement that uses the same password and domain name in the advertisement.

If you are adding a new controller to an existing network with VTP capability, the new controller 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 controller in the domain.

Related Topics

Configuring a VTP Version 3 Password, on page 376
Example: Configuring a Switch as the Primary Server, on page 385

VTP Version

Follow these guidelines when deciding which VTP version to implement:

- All controllers 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 controller can operate in the same VTP domain as a controller running VTP version 1 if version 2 is disabled on the version 2-capable controller (version 2 is disabled by default).
- If a controller 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 controller running VTP version 3 is connected to a controller running VTP version 1, the VTP version 1 controller moves to VTP version 2, and the VTP version 3 controller sends scaled-down versions of the VTP packets so that the VTP version 2 controller can update its database.
- A controller running VTP version 3 cannot move to version 1 or 2 if it has extended VLANs.

- Do not enable VTP version 2 on a controller unless all of the controllers in the same VTP domain are version-2-capable. When you enable version 2 on a controller, all of the version-2-capable controllers in the domain enable version 2. If there is a version 1-only controller, it does not exchange VTP information with controllers that have version 2 enabled.
- Cisco recommends placing VTP version 1 and 2 controllers 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.
- 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 controller for VTP client mode, the controller does not create the VLAN database file (vlan.dat). If the controller is then powered off, it resets the VTP configuration to the default. To keep the VTP configuration with VTP client mode after the controller restarts, you must first configure the VTP domain name before the VTP mode.

Caution If all controllers 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 controller as a VTP server.

Related Topics

Enabling the VTP Version, on page 378

How to Configure VTP

Configuring VTP Mode

You can configure VTP mode as one of these:

- When a controller is in VTP server mode, you can change the VLAN configuration and have it propagated throughout the network.
- When a controller is in VTP client mode, you cannot change its VLAN configuration. The client controller receives VTP updates from a VTP server in the VTP domain and then modifies its configuration accordingly.
- When you configure the controller for VTP transparent mode, VTP is disabled on the controller. The controller does not send VTP updates and does not act on VTP updates received from other controller. However, a VTP transparent controller running VTP version 2 does forward received VTP advertisements on its trunk links.
- VTP off mode is the same as VTP transparent mode except that VTP advertisements are not forwarded.

When you configure a domain name, it cannot be removed; you can only reassign a controller to a different domain.

SUMMARY STEPS

1. **configure terminal**
2. **vtp domain** *domain-name*
3. **vtp mode** {**client** | **server** | **transparent** | **off**} {**vlan** | **mst** | **unknown**}
4. **vtp password** *password*
5. **end**
6. **show vtp status**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**vtp domain** *domain-name* **Example:** `Controller(config)# vtp domain eng_group`	Configures the VTP administrative-domain name. The name can be 1 to 32 characters. All controllers 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 controller has a trunk connection to

	Command or Action	Purpose
		a VTP domain, the controller learns the domain name from the VTP server in the domain. You should configure the VTP domain before configuring other VTP parameters.
Step 3	**vtp mode** {**client** | **server** | **transparent** | **off**} {**vlan** | **mst** | **unknown**} **Example:** `Controller(config)# vtp mode server`	Configures the controller 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.
Step 4	**vtp password** *password* **Example:** `Controller(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 controller in the domain.
Step 5	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 6	**show vtp status** **Example:** `Controller# show vtp status`	Verifies your entries in the *VTP Operating Mode* and the *VTP Domain Name* fields of the display.
Step 7	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves the configuration in the startup configuration file. Only VTP mode and domain name are saved in the controller running configuration and can be copied to the startup configuration file.

Related Topics

VTP Modes, on page 367

Configuring a VTP Version 3 Password

You can configure a VTP version 3 password on the controller.

SUMMARY STEPS

1. **configure terminal**
2. **vtp password** *password* [**hidden** | **secret**]
3. **end**
4. **show vtp password**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**vtp password** *password* [**hidden**	**secret**] **Example:** `Controller(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 nvam:vlan.dat file. If you configure a takeover by configuring a VTP primary server, you are prompted to reenter the password. • (Optional) **secret—**Directly configures the password. The secret password must contain 32 hexadecimal characters.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 4	**show vtp password** **Example:** `Controller# show vtp password`	Verifies your entries. The output appears like this: VTP password: 89914640C8D90868B6A0D8103847A733
Step 5	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves the configuration in the startup configuration file.

Related Topics

Passwords for the VTP Domain, on page 373

Example: Configuring a Switch as the Primary Server, on page 385

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:** `Controller# vtp primary vlan force`	Changes the operational state of a controller from a secondary server (the default) to a primary server and advertises the configuration to the domain. If the controller 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.

Related Topics

VTP Settings, on page 372

Enabling the VTP Version

VTP version 2 and version 3 are disabled by default.

- When you enable VTP version 2 on a controller , every VTP version 2-capable controller in the VTP domain enables version 2. To enable VTP version 3, you must manually configure it on each controller
- With VTP versions 1 and 2, you can configure the version only on controllers in VTP server or transparent mode. If a controller is running VTP version 3, you can change to version 2 when the controller is in client mode if no extended VLANs exist, and no hidden password was configured.

Caution VTP version 1 and VTP version 2 are not interoperable on controllers in the same VTP domain. Do not enable VTP version 2 unless every controller 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. **configure terminal**
2. **vtp version** {**1** | **2** | **3**}
3. **end**
4. **show vtp status**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**vtp version** {**1**	**2**	**3**} **Example:** `Controller(config)# vtp version 2`	Enables the VTP version on the controller . The default is VTP version 1.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 4	**show vtp status** **Example:** `Controller# show vtp status`	Verifies that the configured VTP version is enabled.
Step 5	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves the configuration in the startup configuration file.

Related Topics

VTP Version, on page 373

VTP Version 2, on page 369

VTP Version 3, on page 369

Enabling VTP Pruning

Pruning increases available bandwidth by restricting flooded traffic to those trunk links that the traffic must use to access the destination devices. You can only enable VTP pruning on a controller in VTP server mode.

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 controller in the domain.

Only VLANs included in the pruning-eligible list can be pruned. By default, VLANs 2 through 1001 are pruning-eligible on trunk ports. Reserved VLANs and extended-range VLANs cannot be pruned.

Before You Begin

VTP pruning is not designed to function in VTP transparent mode. If one or more controllers in the network are in VTP transparent mode, you should do one of these:

- Turn off VTP pruning in the entire network.
- Turn off VTP pruning by making all VLANs on the trunk of the controller upstream to the VTP transparent controller 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. **configure terminal**
2. **vtp pruning**
3. **end**
4. **show vtp status**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**vtp pruning** **Example:** `Controller(config)# vtp pruning`	Enables pruning in the VTP administrative domain. By default, pruning is disabled. You need to enable pruning on only one controller in VTP server mode.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 4	**show vtp status** **Example:** `Controller# show vtp status`	Verifies your entries in the *VTP Pruning Mode* field of the display.

Related Topics

VTP Pruning, on page 370

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.

To enable VTP:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **vtp**
4. **end**
5. **show running-config interface** *interface-id*
6. **show vtp status**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet1/0/1`	Identifies an interface, and enters interface configuration mode.
Step 3	**vtp** **Example:** `Controller(config)# vtp`	Enables VTP on the specified port.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 5	**show running-config interface** *interface-id* **Example:** `Controller# show running-config interface gigabitethernet1/0/1`	Verifies the change to the port.
Step 6	**show vtp status** **Example:** `Controller# show vtp status`	Verifies the configuration.

Related Topics

VTP Settings, on page 372

Adding a VTP Client Switch to a VTP Domain

Follow these steps to verify and reset the VTP configuration revision number on a controller *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 controllers in the VTP domain. Controllers in a VTP domain always use the VLAN configuration of the controller with the highest VTP configuration revision number. With VTP versions 1 and 2, adding a controller 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 controller and then to change its VLAN information without affecting the other controllers in the VTP domain.

SUMMARY STEPS

1. **show vtp status**
2. **configure terminal**
3. **vtp domain** *domain-name*
4. **end**
5. **show vtp status**
6. **configure terminal**
7. **vtp domain** *domain-name*
8. **end**
9. **show vtp status**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show vtp status** **Example:** `Controller# show vtp status`	Checks the VTP configuration revision number. If the number is 0, add the controller to the VTP domain. If the number is greater than 0, follow these sub-steps: • Write down the domain name. • Write down the configuration revision number. • Continue with the next steps to reset the controller configuration revision number.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 3	**vtp domain** *domain-name* **Example:** `Controller(config)# vtp domain domain123`	Changes the domain name from the original one displayed in Step 1 to a new name.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. The VLAN information on the controller is updated and the configuration revision number is reset to 0.
Step 5	**show vtp status** **Example:** `Controller# show vtp status`	Verifies that the configuration revision number has been reset to 0.
Step 6	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 7	**vtp domain** *domain-name* **Example:** `Controller(config)# vtp domain domain012`	Enters the original domain name on the controller
Step 8	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. The VLAN information on the controller is updated.
Step 9	**show vtp status** **Example:** `Controller# show vtp status`	(Optional) Verifies that the domain name is the same as in Step 1 and that the configuration revision number is 0.

Related Topics

VTP Domain, on page 366
Prerequisites for VTP, on page 365
Domain Names for Configuring VTP, on page 372

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 controller.

Table 46: 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 controller 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 controller.
show vtp status	Displays the VTP controller configuration information.

Configuration Examples for VTP

Example: Configuring a Switch as the Primary Server

This example shows how to configure a controller as the primary server for the VLAN database (the default) when a hidden or secret password was configured:

```
Controller# 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
```

Related Topics

Configuring a VTP Version 3 Password, on page 376
Passwords for the VTP Domain, on page 373

Where to Go Next

After configuring VTP, you can configure the following:

- VLANs
- VLAN Trunking

Additional References

Standards and RFCs

Standard/RFC	Title
RFC 1573	Evolution of the Interfaces Group of MIB-II
RFC 1757	Remote Network Monitoring Management
RFC 2021	SNMPv2 Management Information Base for the Transmission Control Protocol using SMIv2

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

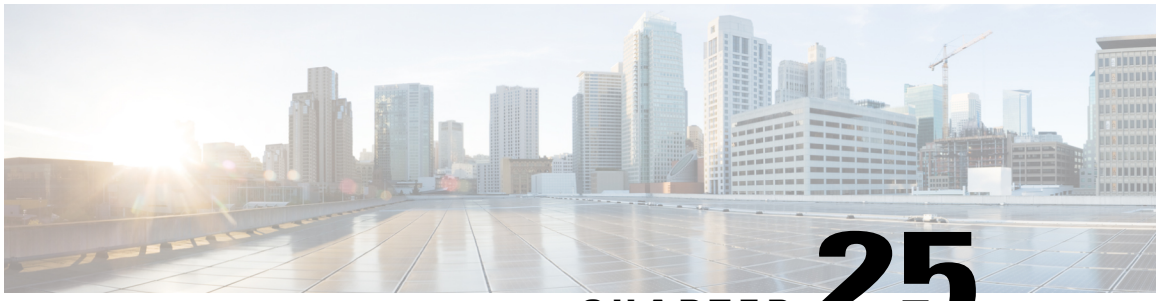
Technical Assistance

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 Information for VTP

Table 47: Feature Information for VTP

Feature Name	Releases	Feature Information
VTP functionality		The following VTP functionality is supported: • VTP Domain • VTP Mode • VTP Advertisements



CHAPTER 25

Configuring VLANs

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 VLANs

The following are prerequisites and considerations for configuring VLANs:

- Before you create VLANs, you must decide whether to use VLAN Trunking Protocol (VTP) to maintain global VLAN configuration for your network.
- If you plan to configure many VLANs on the controller and to not enable routing, you can set the Switch Database Management (SDM) feature to the VLAN template, which configures system resources to support the maximum number of unicast MAC addresses.
- Controllers running the LAN Base feature set support only static routing on SVIs.

- A VLAN should be present in the controller to be able to add it to the VLAN group.

Restrictions for VLANs

The following are restrictions for VLANs:

- The controller 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 controller supports IEEE 802.1Q trunking methods for sending VLAN traffic over Ethernet ports.
- Configuring an interface VLAN router's MAC address is not supported. The interface VLAN already has an MAC address assigned by default.
- The number of VLANs mapped to a VLAN group is not limited by IOS. But if the number of VLANs in a VLAN group exceed the recommended value of 128, the mobility can be unexpected. So it is the responsibility of the administrator to configure feasible number of VLANs in a VLAN group. When a WLAN is mapped to a VLAN group which has more number of VLANs, an error is generated.
- The static IP client behavior is not supported.
- Private VLANs are not supported on the controller.

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 controller 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 controller supporting fallback bridging. 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 controller is assigned manually on an interface-by-interface basis. When you assign controller interfaces to VLANs by using this method, it is known as interface-based, or static, VLAN membership.

Traffic between VLANs must be routed .

The controller can route traffic between VLANs by using controller virtual interfaces (SVIs). An SVI must be explicitly configured and assigned an IP address to route traffic between VLANs.

Supported VLANs

The controller supports VLANs in VTP client, server, and transparent modes. VLANs are identified by a number from 1 to 4094. VLAN 1 is the default VLAN and is created during system initialization. VLAN IDs

1002 through 1005 are reserved for Token Ring and FDDI VLANs. All of the VLANs except 1002 to 1005 are available for user configuration.

There are 3 VTP versions. VTP version 1 and version 2 support only normal-range VLANs (VLAN IDs 1 to 1005). In these versions, the controller 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 can configure up to 4094 VLANs on the controller.

Related Topics

Creating or Modifying an Ethernet VLAN, on page 396
Deleting a VLAN, on page 399
Assigning Static-Access Ports to a VLAN, on page 402
Monitoring VLANs, on page 407
Creating an Extended-Range VLAN, on page 405
Creating an Extended-Range VLAN with an Internal VLAN ID
Monitoring VLANs, on page 407

Creating or Modifying an Ethernet VLAN, on page 396
Deleting a VLAN, on page 399
Assigning Static-Access Ports to a VLAN, on page 402
Monitoring VLANs, on page 407
Creating an Extended-Range VLAN, on page 405
Creating an Extended-Range VLAN with an Internal VLAN ID
Monitoring VLANs, on page 407
Creating or Modifying an Ethernet VLAN, on page 396
Deleting a VLAN, on page 399
Assigning Static-Access Ports to a VLAN, on page 402
Monitoring VLANs, on page 407
Creating an Extended-Range VLAN, on page 405
Creating an Extended-Range VLAN with an Internal VLAN ID
Monitoring VLANs, on page 407
Creating or Modifying an Ethernet VLAN, on page 396
Deleting a VLAN, on page 399
Assigning Static-Access Ports to a VLAN, on page 402
Monitoring VLANs, on page 407
Creating an Extended-Range VLAN, on page 405
Creating an Extended-Range VLAN with an Internal VLAN ID
Monitoring VLANs, on page 407

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 controller learns and manages the addresses associated with the port on a per-VLAN basis.

Table 48: 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 controller connected to a trunk port of a second controller.
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 controllers 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 controller, but you must connect the dynamic-access port to an end station or hub and not to another controller.	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 controller must be connected to a trunk port of a second controller .
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 controller running configuration file.

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 controller, the controller 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 configuration for VLANs 1 to 1005 are always saved in the VLAN database. If the VTP mode is transparent, VTP and VLAN configuration are also saved in the controller running configuration file.
- If the controller 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.)
- 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 and transparent mode.
- Before you can create a VLAN, the controller must be in VTP server mode or VTP transparent mode. If the controller is a VTP server, you must define a VTP domain or VTP will not function.
- The controller does not support Token Ring or FDDI media. The controller does not forward FDDI, FDDI-Net, TrCRF, or TrBRF traffic, but it does propagate the VLAN configuration through VTP.
- The controller supports 128 spanning tree instances. If a controller 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 controller, adding another VLAN anywhere in the VTP domain creates a VLAN on that controller that is not running spanning-tree. If you have the default allowed list on the trunk ports of that controller (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 controllers that all have run out of spanning-tree instances. You can prevent this possibility by setting allowed lists on the trunk ports of controllers that have used up their allocation of spanning-tree instances.

If the number of VLANs on the controller exceeds the number of supported spanning-tree instances, we recommend that you configure the IEEE 802.1s Multiple STP (MSTP) on your controller to map multiple VLANs to a single spanning-tree instance.

Related Topics

Creating or Modifying an Ethernet VLAN, on page 396

Monitoring VLANs, on page 407

Creating or Modifying an Ethernet VLAN, on page 396

Deleting a VLAN, on page 399

Assigning Static-Access Ports to a VLAN, on page 402
Monitoring VLANs, on page 407

Creating or Modifying an Ethernet VLAN, on page 396

Deleting a VLAN, on page 399

Assigning Static-Access Ports to a VLAN, on page 402
Monitoring VLANs, on page 407

Creating or Modifying an Ethernet VLAN, on page 396

Deleting a VLAN, on page 399

Assigning Static-Access Ports to a VLAN, on page 402
Monitoring VLANs, on page 407

Creating or Modifying an Ethernet VLAN, on page 396

Deleting a VLAN, on page 399

Assigning Static-Access Ports to a VLAN, on page 402
Monitoring VLANs, on page 407

Extended-Range VLAN Configuration Guidelines

VTP 3 only supports extended-range VLANs. Extended-range VLANs are VLANs with IDs from 1006 to 4094.

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 controller 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 controller boots up in VTP transparent mode. Otherwise, you lose the extended-range VLAN configuration if the controller resets. If you create extended-range VLANs in VTP version 3, you cannot convert to VTP version 1 or 2.

- STP is enabled by default on extended-range VLANs, but you can disable it by using the **no spanning-tree vlan** *vlan-id* global configuration command. When the maximum number of spanning-tree instances are on the controller, spanning tree is disabled on any newly created VLANs. If the number of VLANs on the controller exceeds the maximum number of spanning-tree instances, we recommend that you configure the IEEE 802.1s Multiple STP (MSTP) on your controller to map multiple VLANs to a single spanning-tree instance.

Related Topics

Creating an Extended-Range VLAN, on page 405

Monitoring VLANs, on page 407

Creating an Extended-Range VLAN, on page 405

Creating an Extended-Range VLAN with an Internal VLAN ID
Monitoring VLANs, on page 407

Creating an Extended-Range VLAN, on page 405

Creating an Extended-Range VLAN with an Internal VLAN ID
Monitoring VLANs, on page 407

Creating an Extended-Range VLAN, on page 405

Creating an Extended-Range VLAN with an Internal VLAN ID
Monitoring VLANs, on page 407

Creating an Extended-Range VLAN, on page 405

Creating an Extended-Range VLAN with an Internal VLAN ID
Monitoring VLANs, on page 407

Information About VLAN Group

Whenever a wireless client connects to a wireless network (WLAN), the client is placed in a VLAN that is associated with the WLAN. In a large venue such as an auditorium, a stadium, or a conference where there are numerous wireless clients, having only a single WLAN to accommodate many clients might be a challenge.

The VLAN group feature enables in using a single WLAN that can support multiple VLANs. The clients can get assigned to one of the configured VLANs. This feature enables to map a WLAN to a single VLAN or multiple VLANs using the VLAN groups. When a wireless client associates to the WLAN, the VLAN is derived by an algorithm based on the MAC address of the wireless client. A VLAN is assigned to the client and the client gets the IP address from the assigned VLAN. This feature also extends the current AP group architecture and AAA override architecture, where the AP groups and AAA override can override a VLAN or a VLAN group to which the WLAN is mapped.

Related Topics

Creating VLAN groups (CLI), on page 401

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)
- Maximum transmission unit (MTU) for the VLAN
- Security Association Identifier (SAID)
- Bridge identification number for TrBRF VLANs
- Ring number for FDDI and TrCRF VLANs
- Parent VLAN number for TrCRF VLANs
- Spanning Tree Protocol (STP) type for TrCRF VLANs
- VLAN number to use when translating from one VLAN type to another

You 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

Before You Begin

With VTP version 1 and 2, if the controller is in VTP transparent mode, you can assign VLAN IDs greater than 1006, but they are not added to the VLAN database.

The controller supports only Ethernet interfaces. 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 controllers.

Although the controller does not support Token Ring connections, a remote device with Token Ring connections could be managed from one of the supported controllers. Controllers running VTP Version 2 advertise information about these Token Ring VLANs:

- Token Ring TrBRF VLANs
- Token Ring TrCRF VLANs

SUMMARY STEPS

1. **configure terminal**
2. **vlan** *vlan-id*
3. **name** *vlan-name*
4. **media** { **ethernet** | **fd-net** | **fddi** | **tokenring** | **trn-net** }
5. **remote-span**
6. **end**
7. **show vlan** {**name** *vlan-name* | **id** *vlan-id*}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**vlan** *vlan-id* **Example:** `Controller(config)# vlan 20`	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. **Note** The available VLAN ID range for this command is 1 to 4094. Additional **vlan** command options include: • **access-map—**Creates VLAN access-maps or enters the vlan access map command mode. • **configuration—**Enters the vlan feature configuration mode. • **dot1q—**Configures VLAN dot1q tag native parameters. • **filter—**Applies a VLAN filter map to a VLAN list. • **group—**Creates a VLAN group.
Step 3	**name** *vlan-name* **Example:** `Controller(config-vlan)# name test20`	(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. The following additional VLAN configuration command options are available:

	Command or Action	Purpose
		• **are—**Sets the maximum number of All Router Explorer (ARE) hops for the VLAN. • **backupcrf—**Enables or disables the backup concentrator relay function (CRF) mode for the VLAN. • **bridge—**Sets the value of the bridge number for the FDDI net or Token Ring net type VLANs. • **exit—**Applies changes, bumps the revision number, and exits. • **media—**Sets the media type of the VLAN. • **no—**Negates the command or default. • **parent—**Sets the value of the ID for the parent VLAN for FDDI or Token Ring type VLANs. • **remote-span—**Configures a remote SPAN VLAN. • **ring—**Sets the ring number value for FDDI or Token Ring type VLANs. • **said—**Sets the IEEE 802.10 SAID value. • **shutdown—**Shuts down the VLAN switching. • **state—**Sets the operational VLAN state to active or suspended. • **ste—**Sets the maximum number of Spanning Tree Explorer (STE) hops for the VLAN. • **stp—**Sets the Spanning Tree characteristics of the VLAN.
Step 4	**media** { **ethernet** | **fd-net** | **fddi** | **tokenring** | **trn-net** } **Example:** `Controller(config-vlan)# media ethernet`	Configures the VLAN media type. Command options include: • **ethernet—**Sets the VLAN media type as Ethernet. • **fd-net—**Sets the VLAN media type as FDDI net. • **fddi—**Sets the VLAN media type as FDDI. • **tokenring—**Sets the VLAN media type as Token Ring. • **trn-net—**Sets the VLAN media type as Token Ring net.
Step 5	**remote-span** **Example:** `Controller(config-vlan)# remote-span`	(Optional) Configures the VLAN as the RSPAN VLAN for a remote SPAN session. For more information on remote SPAN, see the *Catalyst 3850 Network Management Configuration Guide*.

	Command or Action	Purpose
Step 6	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 7	**show vlan** {**name** *vlan-name* | **id** *vlan-id*} **Example:** `Controller# show vlan name test20 id 20`	Verifies your entries.

Related Topics

Normal-Range VLAN Configuration Guidelines, on page 393
Monitoring VLANs, on page 407
Supported VLANs, on page 390
Normal-Range VLAN Configuration Guidelines, on page 393
Monitoring VLANs, on page 407
Supported VLANs, on page 390
Normal-Range VLAN Configuration Guidelines, on page 393
Monitoring VLANs, on page 407
Supported VLANs, on page 390
Normal-Range VLAN Configuration Guidelines, on page 393
Monitoring VLANs, on page 407
Supported VLANs, on page 390
Normal-Range VLAN Configuration Guidelines, on page 393
Monitoring VLANs, on page 407

Deleting a VLAN

When you delete a VLAN from a controller that is in VTP server mode, the VLAN is removed from the VLAN database for all controllers in the VTP domain. When you delete a VLAN from a controller that is in VTP transparent mode, the VLAN is deleted only on that specific controller .

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. **configure terminal**
2. **no vlan** *vlan-id*
3. **end**
4. **show vlan brief**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**no vlan** *vlan-id* **Example:** `Controller(config)# no vlan 4`	Removes the VLAN by entering the VLAN ID.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 4	**show vlan brief** **Example:** `Controller# show vlan brief`	Verifies the VLAN removal.

Related Topics

Monitoring VLANs, on page 407
Supported VLANs, on page 390
Normal-Range VLAN Configuration Guidelines, on page 393
Monitoring VLANs, on page 407
Supported VLANs, on page 390

Normal-Range VLAN Configuration Guidelines, on page 393
Monitoring VLANs, on page 407
Supported VLANs, on page 390
Normal-Range VLAN Configuration Guidelines, on page 393
Monitoring VLANs, on page 407
Supported VLANs, on page 390
Normal-Range VLAN Configuration Guidelines, on page 393
Monitoring VLANs, on page 407

Creating VLAN groups (CLI)

SUMMARY STEPS

1. **configure terminal**
2. **vlan group groupname vlan-list 1-256**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global command mode.
Step 2	**vlan group groupname vlan-list 1-256** **Example:** `Controller#vlan group vlangrp1 vlan-list 91-95`	Creates a VLAN group with the given group name and adds all the VLANs listed in the command. The recommended number of VLANs in a group is 32.
Step 3	**end** **Example:** `Controller(vlan)#end`	Exits the global configuration mode. Alternatively, press CTRL +Z to exit the global configuration mode.

Related Topics

Information About VLAN Group, on page 395

Adding VLAN Group to WLAN (CLI)

SUMMARY STEPS

1. **configure terminal**
2. **wlan wlanname 1 wlanname**
3. **client vlan vlangrp1**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global command mode.
Step 2	**wlan wlanname 1 wlanname** **Example:** `Controller(config)#wlan wlanA 1 wlanA`	Enables the WLAN to map a VLAN group.
Step 3	**client vlan vlangrp1** **Example:** `Controller(config-wlan)#client vlan vlangrp1`	Maps the VLAN group to the WLAN.
Step 4	**end** **Example:** `Controller(config-wlan)#end`	Exits the global configuration mode. Alternatively, press CTRL +Z to exit the global configuration mode.

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). For more information on static-access ports, see VLAN Port Membership Modes.

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:** `Controller# configure terminal`	Enters global configuration mode
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet2/0/1`	Enters the interface to be added to the VLAN.
Step 3	**switchport mode access** **Example:** `Controller(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:** `Controller(config-if)# switchport access vlan 2`	Assigns the port to a VLAN. Valid VLAN IDs are 1 to 4094.
Step 5	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 6	**show running-config interface** *interface-id* **Example:** `Controller# copy running-config startup-config`	Verifies the VLAN membership mode of the interface.
Step 7	**show interfaces** *interface-id* **switchport** **Example:** `Controller# show interfaces gigabitethernet2/0/1`	Verifies your entries in the *Administrative Mode* and the *Access Mode VLAN* fields of the display.

Related Topics

Monitoring VLANs, on page 407

Supported VLANs, on page 390

Normal-Range VLAN Configuration Guidelines, on page 393

Monitoring VLANs, on page 407

Supported VLANs, on page 390

Normal-Range VLAN Configuration Guidelines, on page 393

Monitoring VLANs, on page 407

Supported VLANs, on page 390

Normal-Range VLAN Configuration Guidelines, on page 393

Monitoring VLANs, on page 407

Supported VLANs, on page 390

Normal-Range VLAN Configuration Guidelines, on page 393

Monitoring VLANs, on page 407

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 controller 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 and the remote SPAN configuration state on extended-range VLANs; all other characteristics must remain at the default state.

Creating an Extended-Range VLAN

SUMMARY STEPS

1. **configure terminal**
2. **vlan** *vlan-id*
3. **remote-span**
4. **exit**
5. **interface vlan**
6. **ip mtu** *mtu-size*
7. **end**
8. **show vlan id** *vlan-id*
9. **copy running-config startup config**

DETAILED STEPS

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

Step 2	**vlan** *vlan-id*
	
	Example:
	`Controller(config)# vlan 2000`
	`Controller(config-vlan)#`
		Enters an extended-range VLAN ID and enters VLAN configuration mode. The range is 1006 to 4094.

Step 3	**remote-span** **Example:** `Controller(config-vlan)# remote-span`	(Optional) Configures the VLAN as the RSPAN VLAN.

Step 4	**exit**
	
	Example:
	`Controller(config-vlan)# exit`
	`Controller(config)#`
		Returns to configuration mode.

Step 5	**interface vlan**
	
	Example:
	`Controller(config)# interface vlan 200`
	`Controller(config-if)#`
		Enters the interface configuration mode for the selected VLAN.

	Command or Action	Purpose

Step 6	**ip mtu** *mtu-size*
	
	Example:
	
	`Controller(config-if)# ip mtu 1024`
	`Controller(config-if)#`
		(Optional) Modifies the VLAN by changing the MTU size. You can configure the MTU size between 68 to 1500 bytes.
		
		Note Although all VLAN commands appear in the CLI help, only the **ip mtu** *mtu-size* and **remote-span** commands are supported for extended-range VLANs.

Step 7	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 8	**show vlan id** *vlan-id* **Example:** `Controller# show vlan id 2000`	Verifies that the VLAN has been created.
Step 9	**copy running-config startup config** **Example:** `Controller# copy running-config startup-config`	Saves your entries in the controller 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 controller startup configuration file. Otherwise, if the controller 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

Extended-Range VLAN Configuration Guidelines, on page 394

Monitoring VLANs, on page 407

Supported VLANs, on page 390

Extended-Range VLAN Configuration Guidelines, on page 394

Monitoring VLANs, on page 407

Supported VLANs, on page 390

Extended-Range VLAN Configuration Guidelines, on page 394

Monitoring VLANs, on page 407

Supported VLANs, on page 390

Extended-Range VLAN Configuration Guidelines, on page 394

Monitoring VLANs, on page 407

Supported VLANs, on page 390

Extended-Range VLAN Configuration Guidelines, on page 394
Monitoring VLANs, on page 407

Monitoring VLANs

Table 49: Privileged EXEC show Commands

Command	Purpose
show interfaces [**vlan** *vlan-id*]	Displays characteristics for all interfaces or for the specified VLAN configured on the controller
show vlan [**access-map** *name* | **brief** | **dot1q** { **tag native** } | **filter** [**access-map** | **vlan**] | **group** [**group-name** *name*] | **id** *vlan-id* | **ifindex** | **mtu** | **name** *name* | **remote-span** | **summary**]	Displays parameters for all VLANs or the specified VLAN on the controller . The following command options are available: • **access-map—**Displays the VLAN access-maps. • **brief—**Displays VTP VLAN status in brief. • **dot1q—**Displays the dot1q parameters. • **filter—**Displays VLAN filter information. • **group—**Displays the VLAN group with its name and the connected VLANs that are available. • **id—**Displays VTP VLAN status by identification number. • **ifindex—**Displays SNMP ifIndex. • **mtu—**Displays VLAN MTU information. • **name—**Display the VTP VLAN information by specified name. • **remote-span–**Displays the remote SPAN VLANs. • **summary—**Displays a summary of VLAN information.

Related Topics

Creating or Modifying an Ethernet VLAN, on page 396
Normal-Range VLAN Configuration Guidelines, on page 393
Deleting a VLAN, on page 399
Assigning Static-Access Ports to a VLAN, on page 402
Creating an Extended-Range VLAN, on page 405

Extended-Range VLAN Configuration Guidelines, on page 394

Supported VLANs, on page 390

Normal-Range VLAN Configuration Guidelines, on page 393

Creating or Modifying an Ethernet VLAN, on page 396

Deleting a VLAN, on page 399

Assigning Static-Access Ports to a VLAN, on page 402

Supported VLANs, on page 390

Extended-Range VLAN Configuration Guidelines, on page 394

Creating an Extended-Range VLAN, on page 405

Creating an Extended-Range VLAN with an Internal VLAN ID

Supported VLANs, on page 390

Normal-Range VLAN Configuration Guidelines, on page 393

Creating or Modifying an Ethernet VLAN, on page 396

Deleting a VLAN, on page 399

Assigning Static-Access Ports to a VLAN, on page 402

Supported VLANs, on page 390

Extended-Range VLAN Configuration Guidelines, on page 394

Creating an Extended-Range VLAN, on page 405

Creating an Extended-Range VLAN with an Internal VLAN ID

Supported VLANs, on page 390

Normal-Range VLAN Configuration Guidelines, on page 393

Creating or Modifying an Ethernet VLAN, on page 396

Deleting a VLAN, on page 399

Assigning Static-Access Ports to a VLAN, on page 402

Supported VLANs, on page 390

Extended-Range VLAN Configuration Guidelines, on page 394

Creating an Extended-Range VLAN, on page 405

Creating an Extended-Range VLAN with an Internal VLAN ID

Supported VLANs, on page 390

Normal-Range VLAN Configuration Guidelines, on page 393

Creating or Modifying an Ethernet VLAN, on page 396

Deleting a VLAN, on page 399

Assigning Static-Access Ports to a VLAN, on page 402

Supported VLANs, on page 390

Extended-Range VLAN Configuration Guidelines, on page 394

Creating an Extended-Range VLAN, on page 405

Creating an Extended-Range VLAN with an Internal VLAN ID

Where to Go Next

After configuring VLANs, you can configure the following:

- VLAN trunks
- VLAN Trunking Protocol (VTP)

Additional References

Related Documents

Related Topic	Document Title
CLI commands	

Standards and RFCs

Standard/RFC	Title
RFC 1573	Evolution of the Interfaces Group of MIB-II
RFC 1757	Remote Network Monitoring Management
RFC 2021	SNMPv2 Management Information Base for the Transmission Control Protocol using SMIv2

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

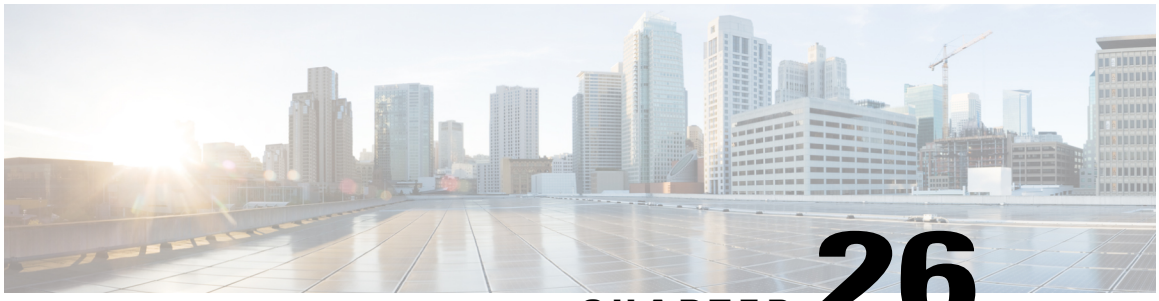
Technical Assistance

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 Information for VLANs

Table 50: Feature Information for VLANs

Feature Name	Releases	Feature Information
VLAN functionality		The controller supports VLANs in VTP client, server, and transparent modes. VLANs are identified by a number from 1 to 4094. VLAN 1 is the default VLAN and is created during system initialization. VLAN IDs 1002 through 1005 are reserved for Token Ring and FDDI VLANs. All of the VLANs except 1002 to 1005 are available for user configuration.



CHAPTER 26

Configuring VLAN Trunks

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 VLAN Trunks

The IEEE 802.1Q trunks impose these limitations on the trunking strategy for a network:

- In a network of Cisco controllers connected through IEEE 802.1Q trunks, the controllers 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 controller to a non-Cisco device through an IEEE 802.1Q trunk, the Cisco controller combines the spanning-tree instance of the VLAN of the trunk with the spanning-tree instance of the non-Cisco IEEE 802.1Q controller. However, spanning-tree information for each VLAN is maintained by Cisco controllers separated by a cloud of non-Cisco IEEE 802.1Q controllers. The

non-Cisco IEEE 802.1Q cloud separating the Cisco controllers is treated as a single trunk link between the controllers.

- 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

Dynamic Trunking Protocol (DTP) is not supported on tunnel ports.

The controller does not support Layer 3 trunks; you cannot configure subinterfaces or use the **encapsulation** keyword on Layer 3 interfaces. The controller 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 controller interfaces and another networking device such as a router or a controller. 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.

Related Topics

Configuring a Trunk Port, on page 415

Layer 2 Interface Modes, on page 413

Layer 2 Interface Modes

Table 51: 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.

Related Topics

Configuring a Trunk Port, on page 415

Trunking Modes, on page 412

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.

Related Topics

Defining the Allowed VLANs on a Trunk, on page 418

Load Sharing on Trunk Ports

Load sharing divides the bandwidth supplied by parallel trunks connecting controllers. To avoid loops, STP normally blocks all but one parallel link between controllers. 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 controller. For load sharing using STP path costs, each load-sharing link can be connected to the same controller or to two different controllers.

Network Load Sharing Using STP Priorities

When two ports on the same controller form a loop, the controller 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.

Related Topics

Configuring Load Sharing Using STP Port Priorities, on page 422

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.

Related Topics

Configuring Load Sharing Using STP Path Cost, on page 426

Feature Interactions

Trunking interacts with other features in these ways:

- A trunk port cannot be a secure 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 controller 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.

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 controller and that this trunk port is connected to the trunk port of a second controller. Otherwise, the controller cannot receive any VTP advertisements.

Before You Begin

By default, an interface is in Layer 2 mode. The default mode for Layer 2 interfaces is **switchport mode dynamic auto**. If the neighboring interface supports trunking and is configured to allow trunking, the link is

a Layer 2 trunk or, if the interface is in Layer 3 mode, it becomes a Layer 2 trunk when you enter the **switchport** interface configuration command.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode** {**dynamic** {**auto** | **desirable**} | **trunk**}
4. **switchport access vlan** *vlan-id*
5. **switchport trunk native vlan** *vlan-id*
6. **end**
7. **show interfaces** *interface-id* **switchport**
8. **show interfaces** *interface-id* **trunk**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet1/0/2`	Specifies the port to be configured for trunking, and enters interface configuration mode.
Step 3	**switchport mode** {**dynamic** {**auto**	**desirable**}	**trunk**} **Example:** `Controller(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.

	Command or Action	Purpose
Step 4	**switchport access vlan** *vlan-id* **Example:** `Controller(config-if)# switchport access vlan 200`	(Optional) Specifies the default VLAN, which is used if the interface stops trunking.
Step 5	**switchport trunk native vlan** *vlan-id* **Example:** `Controller(config-if)# switchport trunk native vlan 200`	Specifies the native VLAN for IEEE 802.1Q trunks.
Step 6	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 7	**show interfaces** *interface-id* **switchport** **Example:** `Controller# 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.
Step 8	**show interfaces** *interface-id* **trunk** **Example:** `Controller# show interfaces gigabitethernet1/0/2 trunk`	Displays the trunk configuration of the interface.
Step 9	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Trunking Modes, on page 412

Layer 2 Interface Modes, on page 413

Defining the Allowed VLANs on a Trunk

VLAN 1 is the default VLAN on all trunk ports in all Cisco controllers, 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** | **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:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet1/0/1`	Specifies the port to be configured, and enters interface configuration mode.
Step 3	**switchport mode trunk** **Example:** `Controller(config-if)# switchport mode trunk`	Configures the interface as a VLAN trunk port.
Step 4	**switchport trunk allowed vlan** {**add** | **all** | **except** | **remove**} *vlan-list* **Example:** `Controller(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.

	Command or Action	Purpose
Step 5	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 6	**show interfaces** *interface-id* **switchport** **Example:** `Controller# show interfaces gigabitethernet1/0/1`	Verifies your entries in the *Trunking VLANs Enabled* field of the display.
Step 7	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Allowed VLANs on a Trunk, on page 413

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. **configure terminal**
2. **interface** *interface-id*
3. **switchport trunk pruning vlan** {**add** | **except** | **none** | **remove**} *vlan-list* [,*vlan* [,*vlan* [,,,]]
4. **end**
5. **show interfaces** *interface-id* **switchport**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet2/0/1`	Selects the trunk port for which VLANs should be pruned, and enter interface configuration mode.
Step 3	**switchport trunk pruning vlan** {**add** | **except** | **none** | **remove**} *vlan-list* [,*vlan* [,*vlan* [,,,]]	Configures the list of VLANs allowed to be pruned from the trunk. 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.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 5	**show interfaces** *interface-id* **switchport** **Example:** `Controller# show interfaces gigabitethernet2/0/1 switchport`	Verifies your entries in the *Pruning VLANs Enabled* field of the display.
Step 6	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

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 controller 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 controller sends the packet with a tag.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport trunk native vlan** *vlan-id*
4. **end**
5. **show interfaces** *interface-id* **switchport**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet1/0/2`	Defines the interface that is configured as the IEEE 802.1Q trunk, and enters interface configuration mode.
Step 3	**switchport trunk native vlan** *vlan-id* **Example:** `Controller(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.
Step 4	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	**show interfaces** *interface-id* **switchport** **Example:** `Controller# show interfaces gigabitethernet1/0/2 switchport`	Verifies your entries in the *Trunking Native Mode VLAN* field.
Step 6	**copy running-config startup-config** **Example:** `Controller# 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

These steps describe how to configure a network with load sharing using STP port priorities.

SUMMARY STEPS

1. **configure terminal**
2. **vtp domain** *domain-name*
3. **vtp mode server**
4. **end**
5. **show vtp status**
6. **show vlan**
7. **configure terminal**
8. **interface** *interface-id*
9. **switchport mode trunk**
10. **end**
11. **show interfaces** *interface-id* **switchport**
12. Repeat Steps 7 through 10 on Controller A for a second port in the controller.
13. Repeat Steps 7 through 10 on Controller B to configure the trunk ports that connect to the trunk ports configured on Controller A.
14. **show vlan**
15. **configure terminal**
16. **interface** *interface-id*
17. **spanning-tree vlan** *vlan-range* **port-priority** *priority-value*
18. **exit**
19. **interface** *interface-id*
20. **spanning-tree vlan** *vlan-range* **port-priority** *priority-value*
21. **end**
22. **show running-config**
23. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode on Controller A.
Step 2	**vtp domain** *domain-name* **Example:** `Controller(config)# vtp domain workdomain`	Configures a VTP administrative domain. The domain name can be 1 to 32 characters.

	Command or Action	Purpose
Step 3	**vtp mode server** **Example:** `Controller(config)# vtp mode server`	Configures Controller A as the VTP server.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 5	**show vtp status** **Example:** `Controller# show vtp status`	Verifies the VTP configuration on both Controller A and Controller B. In the display, check the *VTP Operating Mode* and the *VTP Domain Name* fields.
Step 6	**show vlan** **Example:** `Controller# show vlan`	Verifies that the VLANs exist in the database on Controller A.
Step 7	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 8	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet1/0/1`	Defines the interface to be configured as a trunk, and enter interface configuration mode.
Step 9	**switchport mode trunk** **Example:** `Controller(config-if)# switchport mode trunk`	Configures the port as a trunk port.
Step 10	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.
Step 11	**show interfaces** *interface-id* **switchport** **Example:** `Controller# show interfaces gigabitethernet1/0/1`	Verifies the VLAN configuration.

	Command or Action	Purpose
Step 12	Repeat Steps 7 through 10 on Controller A for a second port in the controller.	
Step 13	Repeat Steps 7 through 10 on Controller B to configure the trunk ports that connect to the trunk ports configured on Controller A.	
Step 14	**show vlan** **Example:** `Controller# show vlan`	When the trunk links come up, VTP passes the VTP and VLAN information to Controller B. This command verifies that Controller B has learned the VLAN configuration.
Step 15	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode on Controller A.
Step 16	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet1/0/1`	Defines the interface to set the STP port priority, and enter interface configuration mode.
Step 17	**spanning-tree vlan** *vlan-range* **port-priority** *priority-value* **Example:** `Controller(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 18	**exit** **Example:** `Controller(config-if)# exit`	Returns to global configuration mode.
Step 19	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet1/0/2`	Defines the interface to set the STP port priority, and enter interface configuration mode.
Step 20	**spanning-tree vlan** *vlan-range* **port-priority** *priority-value* **Example:** `Controller(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.

	Command or Action	Purpose
Step 21	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.
Step 22	**show running-config** **Example:** `Controller# show running-config`	Verifies your entries.
Step 23	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Network Load Sharing Using STP Priorities, on page 414

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. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode trunk**
4. **exit**
5. Repeat Steps 2 through 4 on a second interface in Controller A .
6. **end**
7. **show running-config**
8. **show vlan**
9. **configure terminal**
10. **interface** *interface-id*
11. **spanning-tree vlan** *vlan-range* **cost** *cost-value*
12. **end**
13. Repeat Steps 9 through 13 on the other configured trunk interface on Controller A, and set the spanning-tree path cost to 30 for VLANs 8, 9, and 10.
14. **exit**
15. **show running-config**
16. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode on Controller A.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet1/0/1`	Defines the interface to be configured as a trunk, and enter interface configuration mode.
Step 3	**switchport mode trunk** **Example:** `Controller(config-if)# switchport mode trunk`	Configures the port as a trunk port.
Step 4	**exit** **Example:** `Controller(config-if)# exit`	Returns to global configuration mode.
Step 5	Repeat Steps 2 through 4 on a second interface in Controller A .	
Step 6	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 7	**show running-config** **Example:** `Controller# show running-config`	Verifies your entries. In the display, make sure that the interfaces are configured as trunk ports.
Step 8	**show vlan** **Example:** `Controller# show vlan`	When the trunk links come up, Controller A receives the VTP information from the other controllers. This command verifies that Controller A has learned the VLAN configuration.

	Command or Action	Purpose
Step 9	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 10	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet1/0/1`	Defines the interface on which to set the STP cost, and enters interface configuration mode.
Step 11	**spanning-tree vlan** *vlan-range* **cost** *cost-value* **Example:** `Controller(config-if)# spanning-tree vlan 2-4 cost 30`	Sets the spanning-tree path cost to 30 for VLANs 2 through 4.
Step 12	**end** **Example:** `Controller(config-if)# end`	Returns to global configuration mode.
Step 13	Repeat Steps 9 through 13 on the other configured trunk interface on Controller A, and set the spanning-tree path cost to 30 for VLANs 8, 9, and 10.	
Step 14	**exit** **Example:** `Controller(config)# exit`	Returns to privileged EXEC mode.
Step 15	**show running-config** **Example:** `Controller# show running-config`	Verifies your entries. In the display, verify that the path costs are set correctly for both trunk interfaces.
Step 16	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Network Load Sharing Using STP Path Cost, on page 414

Where to Go Next

After configuring VLAN trunks, you can configure the following:

- VLANs

Additional References

Related Documents

Related Topic	Document Title
CLI commands	*VLAN Command Reference (Cisco WLC 5700 Series)*

Standards and RFCs

Standard/RFC	Title
RFC 1573	Evolution of the Interfaces Group of MIB-II
RFC 1757	Remote Network Monitoring Management
RFC 2021	SNMPv2 Management Information Base for the Transmission Control Protocol using SMIv2

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

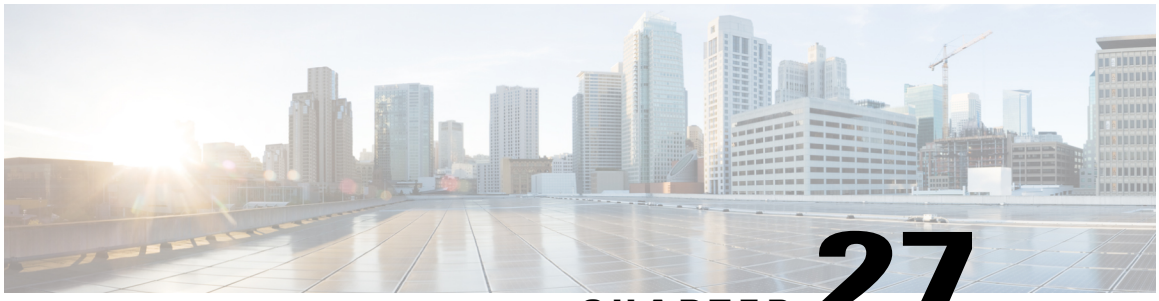
Feature Information for VLAN Trunks

Table 52: Feature Information for VLAN Trunks

Feature Name	Releases	Feature Information
VLAN Trunk Functionality		A trunk is a point-to-point link between one or more Ethernet controller interfaces and another networking device such as a router or a controller. 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.

PART V

VideoStream



CHAPTER 27

Configuring VideoStream

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 VideoStream

Make sure that the multicast feature is enabled. We recommend configuring IP multicast on the controller with multicast-multicast mode.

Check for the IP address on the client machine. The machine should have an IP address from the respective VLAN.

Verify that the access points have joined the controllers.

Restrictions for Configuring VideoStream

Information about VideoStream

The IEEE 802.11 wireless multicast delivery mechanism does not provide a reliable way to acknowledge lost or corrupted packets. As a result, if any multicast packet is lost in the air, it is not sent again which may cause an IP multicast stream unviewable.

The VideoStream feature makes the IP multicast stream delivery reliable over the air, by converting the multicast frame to a unicast frame over the air. Each VideoStream client acknowledges receiving a video IP multicast stream.

How to Configure VideoStream

Configuring Multicast-Direct Globally for Media-Stream

SUMMARY STEPS

1. **configure terminal**
2. **wireless media-stream multicast-direct**
3. **wireless media-stream message**
4. **wireless media-stream group**<*name*><*startIp*><*endIp*>
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**wireless media-stream multicast-direct** **Example:** `Controller(config)#wireless media-stream multicast-direct`	Configures the global multicast-direct feature for the controller.

Step 3	**wireless media-stream message**
	
	Example:
	`Controller(config)#wireless media-stream message ?`
	`  Email  Configure Session Announcement Email`
	`  Notes  Configure Session Announcement notes`
	`  URL    Configure Session Announcement URL`
		Configures various message configuration parameters like phone, URL, email and notes. That is, when a media stream is refused (due to bandwidth constraints), a message can be sent to the user. These parameters configure the messages to send IT support email address, notes (message to display explaining why the stream was refused), URL to which the user can be redirected

	Command or Action	Purpose
	phone Configure Session Announcement Phone number <cr>	and the phone number that the user can call about the refused stream.
Step 4	**wireless media-stream group**<*name*><*startIp*><*endIp*> **Example:** Controller(config)#**wireless media-stream group grp1 231.1.1.1 239.1.1.3** Controller(config-media-stream)#**?** admit Allow traffic for the media stream avg-packet-size Configure Average Packet Size default Set a command to its defaults exit Exit sub-mode max-bandwidth Configure maximum Expected Stream Bandwidth in Kbps no Negate a command or set its defaults priority Configure Media Stream Priotity, <1:Lowest - 8:Highest> qos Configure Over the AIR QoS class, <'video'> ONLY rrc-evaluation RRC re-evaluation admission violation Configure Stream violation policy on periodic re-evaluation <cr>	configures each media stream and its parameters like expected multicast destination addresses, stream bandwidth consumption and stream priority parameters.
Step 5	**end** **Example:** Controller(config)# **end**	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring Media-Stream for 802.11 bands

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 24ghz | 5ghz media-stream multicast-direct**
3. **ap dot11 24ghz | 5ghz media-stream video-redirect**
4. **ap dot11 24ghz | 5ghz media-stream multicast-direct admission-besteffort**
5. **ap dot11 24ghz | 5ghz media-stream multicast-direct client-maximum [<value >]**
6. **ap dot11 24ghz | 5ghz cac multimedia max-bandwidth [<bandwidth>]**
7. **ap dot11 24ghz | 5ghz cac media-stream multicast-direct min_client_rate [<dot11_rate>]**
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 24ghz | 5ghz media-stream multicast-direct** **Example:** `Controller(config)#ap dot11 24ghz media-stream multicast-direct`	Configures if media stream (mc2uc) is allowed for 802.11 band
Step 3	**ap dot11 24ghz | 5ghz media-stream video-redirect** **Example:** `Controller(config)#ap dot11 24ghz media-stream video-redirect`	Configures to redirect unicast video traffic to best effort queue.
Step 4	**ap dot11 24ghz | 5ghz media-stream multicast-direct admission-besteffort** **Example:** `Controller(config)#ap dot11 24ghz media-stream multicast-direct admission-besteffort`	Configures the media stream to still be sent through the best effort queue if a media stream cannot be prioritized due to bandwidth availability limitations. Add **no** in the command to drop the stream if the media stream cannot be prioritized due to bandwidth availability limitations.
Step 5	**ap dot11 24ghz | 5ghz media-stream multicast-direct client-maximum [<value >]** **Example:** `Controller(config)#ap dot11 24ghz media-stream multicast-direct client-max 15`	Configures maximum number of allowed media streams per individual client. The maximum is 15 and the default is 0.
Step 6	**ap dot11 24ghz | 5ghz cac multimedia max-bandwidth [<bandwidth>]** **Example:** `Controller(config)#ap dot11 24ghz cac multimedia max-bandwidth 60`	Configure maximum media (voice + video) bandwidth in %. The range is between 5% and 85%.
Step 7	**ap dot11 24ghz | 5ghz cac media-stream multicast-direct min_client_rate [<dot11_rate>]** **Example:** `Controller(config)#ap dot11 24ghz cac media-stream multicast-direct min_client_rate`	Configures the minimum PHY rate needed for a client to send media-stream as unicast. Clients communicating below this rate will not receive the media stream as a unicast flow. Typically, this PHY rate is equal to or higher than the rate at which multicast frames are sent.
Step 8	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring WLAN to Stream Video

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan_name*
3. **shutdown**
4. **media-stream multicast-direct**
5. **no shutdown**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *wlan_name* **Example:** `Controller(config)#wlan wlan50`	Enables the WLAN configuration mode.
Step 3	**shutdown** **Example:** `Controller(config-wlan)#shutdown`	Disables the WLAN for configuring it parameters.
Step 4	**media-stream multicast-direct** **Example:** `Controller(config)#media-stream multicast-direct`	Configures the multicast-direct feature on media-stream for the WLAN.
Step 5	**no shutdown** **Example:** `Controller(config-wlan)#no shutdown`	Enables the WLAN.
Step 6	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Deleting a Media-Stream

Before You Begin

The media-stream should be enabled and configured for it to be deleted.

SUMMARY STEPS

1. **configure terminal**
2. **no wireless media-stream** <*media_stream_name*>
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**no wireless media-stream** <*media_stream_name*> **Example:** `Controller(config)#no wireless media-stream grp1`	Deletes the media-stream which bears the name mentioned in the command.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Monitoring Media Streams

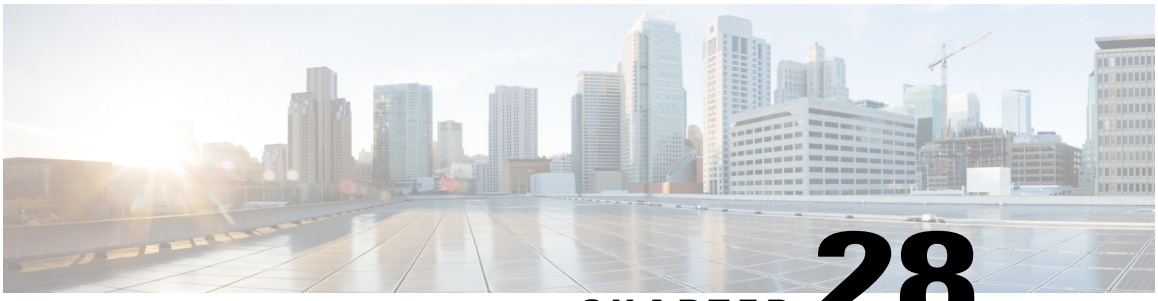
Table 53: Commands for monitoring media streams

Commands	Description
show wireless media-stream client detail *client name*	Displays media stream configuration details of the particular client.
show wireless media-stream client summary	Displays the media stream information of all the clients.
show wireless media-stream group detail *group name*	Displays the media stream configuration details of the particular group.

Commands	Description
show wireless media-stream group summary	Displays the media stream configuration details of all the groups.
show wireless media-stream message details	Displays the session announcement message details.
show wireless multicast	Displays the multicast-direct configuration state.
show ap dot11 24ghz | 5ghz media-stream rrc	Displays 802.11 media Resource-Reservation-Control configurations.

PART VI

Multicast



CHAPTER 28

Configuring IGMP

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 IGMP

The following are the restrictions for configuring IGMP:

- The controller supports IGMP Versions 1, 2 , and 3.

Note For IGMP Version 3, only IGMP Version 3 BISS (Basic IGMPv3 Snooping Support) is supported.

- IGMP Version 3 uses new membership report messages that might not be correctly recognized by older IGMP snooping controllers.
- IGMP filtering and throttling is not supported under the WLAN.

Information About IGMP

To participate in IP multicasting, multicast hosts, routers, and multilayer controllers must have the Internet Group Management Protocol (IGMP) operating. This protocol defines the querier and host roles:

- A querier is a network device that sends query messages to discover which network devices are members of a given multicast group.
- A host is a receiver that sends report messages (in response to query messages) to inform a querier of a host membership.

A set of queriers and hosts that receive multicast data streams from the same source is called a multicast group. Queriers and hosts use IGMP messages to join and leave multicast groups.

Any host, regardless of whether it is a member of a group, can send to a group. However, only the members of a group receive the message. Membership in a multicast group is dynamic; hosts can join and leave at any time. There is no restriction on the location or number of members in a multicast group. A host can be a member of more than one multicast group at a time. How active a multicast group is and what members it has can vary from group to group and from time to time. A multicast group can be active for a long time, or it can be very short-lived. Membership in a group can constantly change.

IP Multicast Group Addresses

IP multicast traffic uses group addresses, which are class D addresses. The high-order bits of a Class D address are 1110. Therefore, host group addresses can be in the range 224.0.0.0 through 239.255.255.255. Multicast addresses in the range 224.0.0.0 to 224.0.0.255 are reserved for use by routing protocols and other network control traffic. The address 224.0.0.0 is guaranteed not to be assigned to any group.

IGMP packets are sent using these IP multicast group addresses:

- IGMP general queries are destined to the address 224.0.0.1 (all systems on a subnet).
- IGMP group-specific queries are destined to the group IP address for which the controller is querying.
- IGMP group membership reports are destined to the group IP address for which the controller is reporting.
- IGMP Version 2 (IGMPv2) leave messages are destined to the address 224.0.0.2 (all multicast routers on a subnet). In some old host IP stacks, leave messages might be destined to the group IP address rather than to the all-routers address.

Related Topics

Configuring the Controller as a Member of a Group, on page 452

Example: Configuring the Controller as a Member of a Multicast Group, on page 490

IGMP Versions

The controller supports IGMP version 1, IGMP version 2, and IGMP version 3. These versions are interoperable on the controller. For example, if IGMP snooping is enabled and the querier's version is IGMPv2. and the controller receives an IGMPv3 report from a host, then the controller can forward the IGMPv3 report to the multicast router.

IGMP Version 1

IGMP version 1 (IGMPv1) primarily uses a query-response model that enables the multicast router and multilayer controller to find which multicast groups are active (have one or more hosts interested in a multicast group) on the local subnet. IGMPv1 has other processes that enable a host to join and leave a multicast group. For more information, see RFC 1112.

IGMP Version 2

IGMPv2 extends IGMP functionality by providing such features as the IGMP leave process to reduce leave latency, group-specific queries, and an explicit maximum query response time. IGMPv2 also adds the capability for routers to elect the IGMP querier without depending on the multicast protocol to perform this task. For more information, see RFC 2236.

Note IGMP version 2 is the default version for the controller.

IGMP Version 3

The controller supports IGMP version 3. The following are considerations for the controller and IGMP version 3:

- An IGMPv3 controller supports Basic IGMPv3 Snooping Support (BISS), which includes support for the snooping features on IGMPv1 and IGMPv2 switches and for IGMPv3 membership report messages. BISS constrains the flooding of multicast traffic when your network includes IGMPv3 hosts. It constrains traffic to approximately the same set of ports as the IGMP snooping feature on IGMPv2 or IGMPv1 hosts.
- The controller supports IGMPv3 snooping based only on the destination multicast IP address. It does not support snooping based on a source IP address or proxy report.
- IGMPv3 join and leave messages are not supported on controllers running IGMP filtering or Multicast VLAN registration (MVR).
- An IGMPv3 controller can receive messages from and forward messages to a device running the Source Specific Multicast (SSM) feature.

IGMPv3 Host Signalling

In IGMPv3, hosts signal membership to last hop routers of multicast groups. Hosts can signal group membership with filtering capabilities with respect to sources. A host can either signal that it wants to receive traffic from

all sources sending to a group except for some specific sources (called exclude mode), or that it wants to receive traffic only from some specific sources sending to the group (called include mode).

IGMPv3 can operate with both Internet Standard Multicast (ISM) and Source Specific Multicast (SSM). In ISM, both exclude and include mode reports are applicable. In SSM, only include mode reports are accepted by the last-hop router. Exclude mode reports are ignored.

IGMP Snooping

Layer 2 controllers can use IGMP snooping to constrain the flooding of multicast traffic by dynamically configuring Layer 2 interfaces so that multicast traffic is forwarded to only those interfaces associated with IP multicast devices. As the name implies, IGMP snooping requires the LAN controller to snoop on the IGMP transmissions between the host and the router and to keep track of multicast groups and member ports. When the controller receives an IGMP report from a host for a particular multicast group, the controller adds the host port number to the forwarding table entry; when it receives an IGMP Leave Group message from a host, it removes the host port from the table entry. It also periodically deletes entries if it does not receive IGMP membership reports from the multicast clients.

Note For more information on IP multicast and IGMP, see RFC 1112 and RFC 2236.

The multicast router (which could be a controller with the IP services feature set on the active switch) sends out periodic general queries to all VLANs. All hosts interested in this multicast traffic send join requests and are added to the forwarding table entry. The controller creates one entry per VLAN in the IGMP snooping IP multicast forwarding table for each group from which it receives an IGMP join request.

The controller supports IP multicast group-based bridging, instead of MAC-addressed based groups. With multicast MAC address-based groups, if an IP address being configured translates (aliases) to a previously configured MAC address or to any reserved multicast MAC addresses (in the range 224.0.0.xxx), the command fails. Because the controller uses IP multicast groups, there are no address aliasing issues.

The IP multicast groups learned through IGMP snooping are dynamic. However, you can statically configure multicast groups by using the **ip igmp snooping vlan***vlan-id***static** *ip_address* **interface** *interface-id* global configuration command. If you specify group membership for a multicast group address statically, your setting supersedes any automatic manipulation by IGMP snooping. Multicast group membership lists can consist of both user-defined and IGMP snooping-learned settings.

You can configure an IGMP snooping querier to support IGMP snooping in subnets without multicast interfaces because the multicast traffic does not need to be routed.

If a port spanning-tree, a port group, or a VLAN ID change occurs, the IGMP snooping-learned multicast groups from this port on the VLAN are deleted.

Related Topics

Enabling IGMP Snooping on a Controller, on page 469

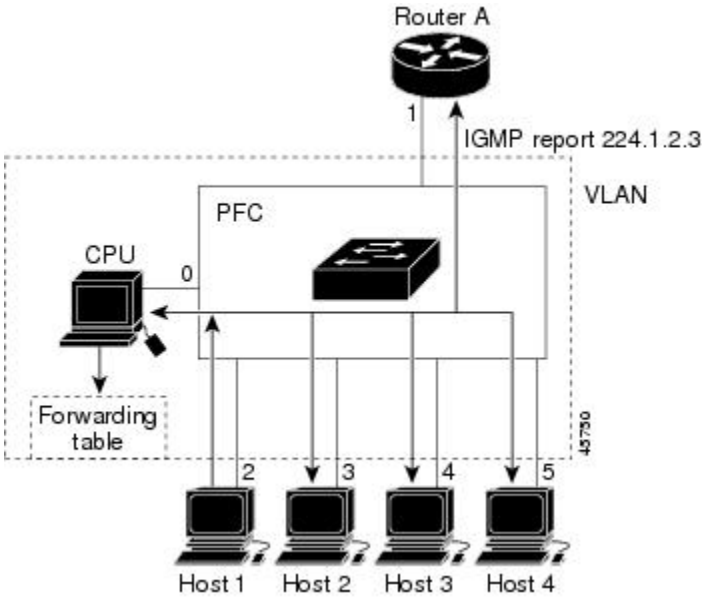
Examples: Configuring IGMP Snooping, on page 491

Joining a Multicast Group

When a host connected to the controller wants to join an IP multicast group and it is an IGMP version 2 client, it sends an unsolicited IGMP join message, specifying the IP multicast group to join. Alternatively, when the

controller receives a general query from the router, it forwards the query to all ports in the VLAN. IGMP version 1 or version 2 hosts wanting to join the multicast group respond by sending a join message to the controller. The controller CPU creates a multicast forwarding-table entry for the group if it is not already present. The CPU also adds the interface where the join message was received to the forwarding-table entry. The host associated with that interface receives multicast traffic for that multicast group.

Figure 10: Initial IGMP Join Message



Router A sends a general query to the controller, which forwards the query to ports 2 through 5, which are all members of the same VLAN. Host 1 wants to join multicast group 224.1.2.3 and multicasts an IGMP membership report (IGMP join message) to the group. The controller CPU uses the information in the IGMP report to set up a forwarding-table entry that includes the port numbers connected to Host 1 and to the router.

Table 54: IGMP Snooping Forwarding Table

Destination Address	Type of Packet	Ports
224.1.2.3	IGMP	1, 2

The controller hardware can distinguish IGMP information packets from other packets for the multicast group. The information in the table tells the switching engine to send frames addressed to the 224.1.2.3 multicast IP address that are not IGMP packets to the router and to the host that has joined the group.

If another host (for example, Host 4) sends an unsolicited IGMP join message for the same group, the CPU receives that message and adds the port number of Host 4 to the forwarding table. Because the forwarding

table directs IGMP messages only to the CPU, the message is not flooded to other ports on the controller. Any known multicast traffic is forwarded to the group and not to the CPU.

Figure 11: Second Host Joining a Multicast Group

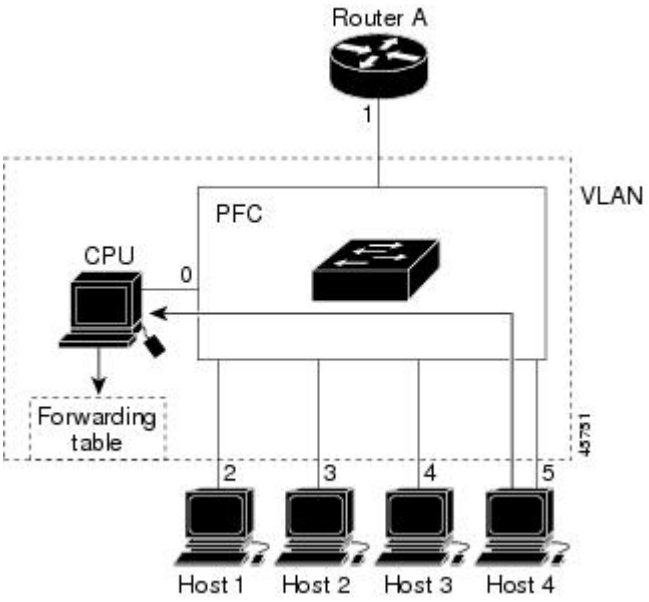


Table 55: Updated IGMP Snooping Forwarding Table

Destination Address	Type of Packet	Ports
224.1.2.3	IGMP	1, 2, 5

Related Topics

Configuring the Controller as a Member of a Group, on page 452

Example: Configuring the Controller as a Member of a Multicast Group, on page 490

Leaving a Multicast Group

The router sends periodic multicast general queries, and the controller forwards these queries through all ports in the VLAN. Interested hosts respond to the queries. If at least one host in the VLAN wants to receive multicast traffic, the router continues forwarding the multicast traffic to the VLAN. The controller forwards multicast group traffic only to those hosts listed in the forwarding table for that IP multicast group maintained by IGMP snooping.

When hosts want to leave a multicast group, they can silently leave, or they can send a leave message. When the controller receives a leave message from a host, it sends a group-specific query to learn if any other devices connected to that interface are interested in traffic for the specific multicast group. The controller then updates the forwarding table for that MAC group so that only those hosts interested in receiving multicast traffic for the group are listed in the forwarding table. If the router receives no reports from a VLAN, it removes the group for the VLAN from its IGMP cache.

Immediate Leave

The controller uses IGMP snooping Immediate Leave to remove from the forwarding table an interface that sends a leave message without the controller sending group-specific queries to the interface. The VLAN interface is pruned from the multicast tree for the multicast group specified in the original leave message. Immediate Leave ensures optimal bandwidth management for all hosts on a switched network, even when multiple multicast groups are simultaneously in use.

Immediate Leave is only supported on IGMP version 2 hosts. IGMP version 2 is the default version for the controller.

Note You should only use the Immediate Leave feature on VLANs where a single host is connected to each port. If Immediate Leave is enabled in VLANs where more than one host is connected to a port, some hosts might inadvertently be dropped.

IGMP Configurable-Leave Timer

You can configure the time that the controller waits after sending a group-specific query to determine if hosts are still interested in a specific multicast group. The IGMP leave response time can be configured from 100 to 5000 milliseconds. The timer can be set either globally or on a per-VLAN basis. The VLAN configuration of the leave time overrides the global configuration.

Related Topics

Configuring the IGMP Leave Timer, on page 476

IGMP Report Suppression

Note IGMP report suppression is supported only when the multicast query has IGMPv1 and IGMPv2 reports. This feature is not supported when the query includes IGMPv3 reports.

The controller uses IGMP report suppression to forward only one IGMP report per multicast router query to multicast devices. When IGMP router suppression is enabled (the default), the controller sends the first IGMP report from all hosts for a group to all the multicast routers. The controller does not send the remaining IGMP reports for the group to the multicast routers. This feature prevents duplicate reports from being sent to the multicast devices.

If the multicast router query includes requests only for IGMPv1 and IGMPv2 reports, the controller forwards only the first IGMPv1 or IGMPv2 report from all hosts for a group to all the multicast routers.

If the multicast router query also includes requests for IGMPv3 reports, the controller forwards all IGMPv1, IGMPv2, and IGMPv3 reports for a group to the multicast devices.

If you disable IGMP report suppression, all IGMP reports are forwarded to the multicast routers.

IGMP Filtering and Throttling Overview

In some environments, for example, metropolitan or multiple-dwelling unit (MDU) installations, you might want to control the set of multicast groups to which a user on a controller port can belong. You can control

the distribution of multicast services, such as IP/TV, based on some type of subscription or service plan. You might also want to limit the number of multicast groups to which a user on a controller port can belong.

With the IGMP filtering feature, you can filter multicast joins on a per-port basis by configuring IP multicast profiles and associating them with individual controller ports. An IGMP profile can contain one or more multicast groups and specifies whether access to the group is permitted or denied. If an IGMP profile denying access to a multicast group is applied to a controller port, the IGMP join report requesting the stream of IP multicast traffic is dropped, and the port is not allowed to receive IP multicast traffic from that group. If the filtering action permits access to the multicast group, the IGMP report from the port is forwarded for normal processing. You can also set the maximum number of IGMP groups that a Layer 2 interface can join.

IGMP filtering controls only group-specific query and membership reports, including join and leave reports. It does not control general IGMP queries. IGMP filtering has no relationship with the function that directs the forwarding of IP multicast traffic. The filtering feature operates in the same manner whether CGMP or MVR is used to forward the multicast traffic.

IGMP filtering applies only to the dynamic learning of IP multicast group addresses, not static configuration.

With the IGMP throttling feature, you can set the maximum number of IGMP groups that a Layer 2 interface can join. If the maximum number of IGMP groups is set, the IGMP snooping forwarding table contains the maximum number of entries, and the interface receives an IGMP join report, you can configure an interface to drop the IGMP report or to replace the randomly selected multicast entry with the received IGMP report.

Note IGMPv3 join and leave messages are not supported on controllers running IGMP filtering.

Related Topics

Configuring the IGMP Throttling Action, on page 467

Displaying IGMP Filtering and Throttling Configuration, on page 490

Examples: Configuring Filtering and Throttling, on page 492

Default IGMP Configuration

This table displays the default IGMP configuration for the controller.

Table 56: Default IGMP Configuration

Feature	Default Setting
Multilayer controller as a member of a multicast group	No group memberships are defined.
Access to multicast groups	All groups are allowed on an interface.
IGMP version	Version 2 on all interfaces.
IGMP host-query message interval	60 seconds on all interfaces.
IGMP query timeout	60 seconds on all interfaces.
IGMP maximum query response time	10 seconds on all interfaces.

Feature	Default Setting
Multilayer controller as a statically connected member	Disabled.

Default IGMP Snooping Configuration

This table displays the default IGMP snooping configuration for the controller.

Table 57: Default IGMP Snooping Configuration

Feature	Default Setting
IGMP snooping	Enabled globally and per VLAN
Multicast routers	None configured
IGMP snooping Immediate Leave	Disabled
Static groups	None configured
TCN[3] flood query count	2
TCN query solicitation	Disabled
IGMP snooping querier	Disabled
IGMP report suppression	Enabled

3 (1) TCN = Topology Change Notification

Default IGMP Filtering and Throttling Configuration

This table displays the default IGMP filtering and throttling configuration for the controller.

Table 58: Default IGMP Filtering Configuration

Feature	Default Setting
IGMP filters	None applied.
IGMP maximum number of IGMP groups	No maximum set. **Note** When the maximum number of groups is in the forwarding table, the default IGMP throttling action is to deny the IGMP report.
IGMP profiles	None defined.

Feature	Default Setting
IGMP profile action	Deny the range addresses.

How to Configure IGMP

Configuring the Controller as a Member of a Group

You can configure the controller as a member of a multicast group and discover multicast reachability in a network. If all the multicast-capable routers and multilayer controllers that you administer are members of a multicast group, pinging that group causes all these devices to respond. The devices respond to ICMP echo-request packets addressed to a group of which they are members. Another example is the multicast trace-route tools provided in the software.

Caution Performing this procedure might impact the CPU performance because the CPU will receive all data traffic for the group address.

This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip igmp join-group** *group-address*
4. **end**
5. **show ip igmp interface** [*interface-id*]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet 1/0/1`	Specifies the Layer 3 interface on which you want to enable multicast 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.

	Command or Action	Purpose
		You will also need to enable IP PIM sparse-dense-mode on the interface, and join the interface as a statically connected member to an IGMP static group. For a configuration example, see Example: Interface Configuration as a Routed Port, on page 492 • 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-dense-mode on the VLAN, join the VLAN as a statically connected member to an IGMP static group, and then enable IGMP snooping on the VLAN, the IGMP static group, and physical interface. For a configuration example, see Example: Interface Configuration as an SVI, on page 493 These interfaces must have IP addresses assigned to them.
Step 3	**ip igmp join-group** *group-address* **Example:** `Controller(config-if)# ip igmp join-group 225.2.2.2`	Configures the controller to join a multicast group. By default, no group memberships are defined. For *group-address*, specify the multicast IP address in dotted decimal notation.
Step 4	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show ip igmp interface** [*interface-id*] **Example:** `Controller# show ip igmp interface`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Joining a Multicast Group, on page 446
Example: Configuring the Controller as a Member of a Multicast Group, on page 490
IP Multicast Group Addresses, on page 444
Example: Configuring the Controller as a Member of a Multicast Group, on page 490

Controlling Access to IP Multicast Group

The controller sends IGMP host-query messages to find which multicast groups have members on attached local networks. The controller then forwards to these group members all packets addressed to the multicast group. You can place a filter on each interface to restrict the multicast groups that hosts on the subnet serviced by the interface can join.

To limit the number of joins on the interface, configure the port for the filter which associates with the IGMP profile.

This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp profile**
3. **permit**
4. **exit**
5. **interface** *interface-id*
6. **ip igmp filter** *filter_number*
7. **end**
8. **show ip igmp interface** [*interface-id*]

DETAILED STEPS

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

Step 2	**ip igmp profile**
	
	Example:
	
	`Controller(config)# ip igmp profile 10`
	`Controller(config-igmp-profile)# ?`
		Enters an IGMP filter profile number from 1 to 4294967295.
		
		For additional information about configuring IGMP filter profiles, see Configuring IGMP Profiles, on page 462.

Step 3	**permit**
	
	Example:
	
	`Controller(config-igmp-profile)#`
	`permit 229.9.9.0`
		Enters an IGMP profile configuration action. The following IGMP profile configuration actions are supported:
		
		• **deny—**Matching IP addresses are denied.
		• **exit—**Exits from the IGMP profile configuration mode.
		• **no—**Negates a command or set its defaults.
		• **permit—**Matching addresses are permitted.
		• **range—**Adds a range to the set.

	Command or Action	Purpose
Step 4	**exit** **Example:** `Controller(config-igmp-profile)# exit`	Returns to global configuration mode.
Step 5	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet 1/0/1`	Specifies the interface to be configured, and enters interface configuration mode.
Step 6	**ip igmp filter** *filter_number* **Example:** `Controller(config-if)# ip igmp filter 10`	Specifies the IGMP filter profile number. For additional information about applying IGMP filter profiles, see Applying IGMP Profiles, on page 464.
Step 7	**end** **Example:** `Controller(config-igmp-profile)# end`	Returns to privileged EXEC mode.
Step 8	**show ip igmp interface** [*interface-id*] **Example:** `Controller# show ip igmp interface`	Verifies your entries.

Modifying the IGMP Host-Query Message Interval

The controller periodically sends IGMP host-query messages to discover which multicast groups are present on attached networks. These messages are sent to the all-hosts multicast group (224.0.0.1) with a time-to-live (TTL) of 1. The controller sends host-query messages to refresh its knowledge of memberships present on the network. If, after some number of queries, the software discovers that no local hosts are members of a multicast group, the software stops forwarding multicast packets to the local network from remote origins for that group and sends a prune message upstream toward the source.

The controller elects a PIM designated router (DR) for the LAN (subnet). The DR is the router or multilayer controller with the highest IP address for IGMPv2. For IGMPv1, the DR is elected according to the multicast routing protocol that runs on the LAN. The designated router is responsible for sending IGMP host-query messages to all hosts on the LAN. In sparse mode, the designated router also sends PIM register and PIM join messages toward the RP router.

This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip igmp query-interval** *seconds*
4. **end**
5. **show ip igmp interface** [*interface-id*]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet 1/0/1`	Specifies the Layer 3 interface on which you want to enable multicast 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-dense-mode on the interface, and join the interface as a statically connected member to an IGMP static group. For a configuration example, see Example: Interface Configuration as a Routed Port, on page 492 • 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-dense-mode on the VLAN, join the VLAN as a statically connected member to an IGMP static group, and then enable IGMP snooping on the VLAN, the IGMP static group, and physical interface. For a configuration example, see Example: Interface Configuration as an SVI, on page 493 These interfaces must have IP addresses assigned to them.
Step 3	**ip igmp query-interval** *seconds* **Example:** `Controller(config-if)# ip igmp query-interval 75`	Configures the frequency at which the designated router sends IGMP host-query messages. By default, the designated router sends IGMP host-query messages every 60 seconds to keep the IGMP overhead very low on hosts and networks. The range is 1 to 65535.

	Command or Action	Purpose
Step 4	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show ip igmp interface** [*interface-id*] **Example:** `Controller# show ip igmp interface`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Changing the IGMP Query Timeout for IGMPv2

If you are using IGMPv2, you can specify the period of time before the controller takes over as the querier for the interface. By default, the controller waits twice the query interval controlled by the **ip igmp query-interval** interface configuration command. After that time, if the controller has received no queries, it becomes the querier.

This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip igmp querier-timeout** *seconds*
4. **end**
5. **show ip igmp interface** [*interface-id*]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet 1/0/1`	Specifies the Layer 3 interface on which you want to enable multicast 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-dense-mode on the interface, and join the interface as a statically connected member to an IGMP static group. For a configuration example, see Example: Interface Configuration as a Routed Port, on page 492 • 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-dense-mode on the VLAN, join the VLAN as a statically connected member to an IGMP static group, and then enable IGMP snooping on the VLAN, the IGMP static group, and physical interface. For a configuration example, see Example: Interface Configuration as an SVI, on page 493 These interfaces must have IP addresses assigned to them.
Step 3	**ip igmp querier-timeout** *seconds* **Example:** `Controller(config-if)# ip igmp querier-timeout 120`	Specifies the IGMP query timeout. The default is 60 seconds (twice the query interval). The range is 60 to 300.
Step 4	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show ip igmp interface** [*interface-id*] **Example:** `Controller# show ip igmp interface`	Verifies your entries.

	Command or Action	Purpose
Step 6	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Changing the Maximum Query Response Time for IGMPv2

If you are using IGMPv2, you can change the maximum query response time advertised in IGMP queries. The maximum query response time enables the controller to quickly detect that there are no more directly connected group members on a LAN. Decreasing the value enables the controller to prune groups faster.

This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip igmp query-max-response-time** *seconds*
4. **end**
5. **show ip igmp interface** [*interface-id*]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet 1/0/1`	Specifies the Layer 3 interface on which you want to enable multicast 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-dense-mode on the interface, and join the interface as a statically connected member to an IGMP static group. For a configuration example, see Example: Interface Configuration as a Routed Port, on page 492

	Command or Action	Purpose
		• 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-dense-mode on the VLAN, join the VLAN as a statically connected member to an IGMP static group, and then enable IGMP snooping on the VLAN, the IGMP static group, and physical interface. For a configuration example, see Example: Interface Configuration as an SVI, on page 493 These interfaces must have IP addresses assigned to them.
Step 3	**ip igmp query-max-response-time** *seconds* **Example:** `Controller(config-if)# ip igmp query-max-response-time 15`	Changes the maximum query response time advertised in IGMP queries. The default is 10 seconds. The range is 1 to 25.
Step 4	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show ip igmp interface** [*interface-id*] **Example:** `Controller# show ip igmp interface`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring the Controller as a Statically Connected Member

At various times, there is either no group member on a network segment or a host cannot report its group membership by using IGMP. However, you may want multicast traffic to be sent to that network segment. The following commands are used to pull multicast traffic down to a network segment:

- **ip igmp join-group**—The controller accepts the multicast packets in addition to forwarding them. Accepting the multicast packets prevents the controller from fast switching.

- **ip igmp static-group**—The controller does not accept the packets itself, but only forwards them. This method enables fast switching. The outgoing interface appears in the IGMP cache, but the controller itself is not a member, as evidenced by lack of an L (local) flag in the multicast route entry.

This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip igmp static-group** *group-address*
4. **end**
5. **show ip igmp interface** [*interface-id*]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet 1/0/1`	Specifies the Layer 3 interface on which you want to enable multicast 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-dense-mode on the interface, and join the interface as a statically connected member to an IGMP static group. For a configuration example, see Example: Interface Configuration as a Routed Port, on page 492 • 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-dense-mode on the VLAN, join the VLAN as a statically connected member to an IGMP static group, and then enable IGMP snooping on the VLAN, the IGMP static group, and physical interface. For a configuration example, see Example: Interface Configuration as an SVI, on page 493 These interfaces must have IP addresses assigned to them.
Step 3	**ip igmp static-group** *group-address*	Configures the controller as a statically connected member of a group.

	Command or Action	Purpose
	Example: `Controller(config-if)# ip igmp static-group 239.100.100.101`	By default, this feature is disabled.
Step 4	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show ip igmp interface** [*interface-id*] **Example:** `Controller# show ip igmp interface gigabitethernet 1/0/1`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring IGMP Profiles

To configure an IGMP profile, use the **ip igmp profile** global configuration command with a profile number to create an IGMP profile and to enter IGMP profile configuration mode. From this mode, you can specify the parameters of the IGMP profile to be used for filtering IGMP join requests from a port. When you are in IGMP profile configuration mode, you can create the profile by using these commands:

- **deny—**Specifies that matching addresses are denied; this is the default.
- **exit—**Exits from igmp-profile configuration mode.
- **no—**Negates a command or returns to its defaults.
- **permit—**Specifies that matching addresses are permitted.
- **range—**Specifies a range of IP addresses for the profile. You can enter a single IP address or a range with a start and an end address.

The default is for the controller to have no IGMP profiles configured. When a profile is configured, if neither the **permit** nor **deny** keyword is included, the default is to deny access to the range of IP addresses.

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp profile** *profile number*
3. **permit** | **deny**
4. **range** *ip multicast address*
5. **end**
6. **show ip igmp profile** *profile number*
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ip igmp profile** *profile number* **Example:** `Controller(config)# ip igmp profile 3`	Assigns a number to the profile you are configuring, and enters IGMP profile configuration mode. The profile number range is 1 to 4294967295.
Step 3	**permit** | **deny** **Example:** `Controller(config-igmp-profile)# permit`	(Optional) Sets the action to permit or deny access to the IP multicast address. If no action is configured, the default for the profile is to deny access.
Step 4	**range** *ip multicast address* **Example:** `Controller(config-igmp-profile)# range 229.9.9.0`	Enters the IP multicast address or range of IP multicast addresses to which access is being controlled. If entering a range, enter the low IP multicast address, a space, and the high IP multicast address. You can use the **range** command multiple times to enter multiple addresses or ranges of addresses.
Step 5	**end** **Example:** `Controller(config-igmp-profile)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 6	**show ip igmp profile** *profile number* **Example:** `Controller# show ip igmp profile 3`	Verifies the profile configuration.
Step 7	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Applying IGMP Profiles

To control access as defined in an IGMP profile, use the **ip igmp filter** interface configuration command to apply the profile to the appropriate interfaces. You can apply IGMP profiles only to Layer 2 access ports; you cannot apply IGMP profiles to routed ports or SVIs. You cannot apply profiles to ports that belong to an EtherChannel port group. You can apply a profile to multiple interfaces, but each interface can have only one profile applied to it.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip igmp filter** *profile number*
4. **end**
5. **show running-config interface** *interface-id*
6. **copy running-config startup-config**

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet1/0/1`	Specifies the physical interface, and enters interface configuration mode. The interface must be a Layer 2 port that does not belong to an EtherChannel port group.
Step 3	**ip igmp filter** *profile number* **Example:** `Controller(config-if)# ip igmp filter 321`	Applies the specified IGMP profile to the interface. The range is 1 to 4294967295.
Step 4	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show running-config interface** *interface-id* **Example:** `Controller# show running-config interface gigabitethernet1/0/1`	Verifies the configuration.
Step 6	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Setting the Maximum Number of IGMP Groups

You can set the maximum number of IGMP groups that a Layer 2 interface can join by using the **ip igmp max-groups** interface configuration command. Use the **no** form of this command to set the maximum back to the default, which is no limit.

This restriction can be applied to Layer 2 ports only; you cannot set a maximum number of IGMP groups on routed ports or SVIs. You also can use this command on a logical EtherChannel interface but cannot use it on ports that belong to an EtherChannel port group.

Beginning in privileged EXEC mode, follow these steps to set the maximum number of IGMP groups in the forwarding table:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip igmp max-groups** *number*
4. **end**
5. **show running-config interface** *interface-id*
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet1/0/2`	Specifies the interface to be configured, and enters interface configuration mode. The interface can be a Layer 2 port that does not belong to an EtherChannel group or a EtherChannel interface.
Step 3	**ip igmp max-groups** *number* **Example:** `Controller(config-if)# ip igmp max-groups 20`	Sets the maximum number of IGMP groups that the interface can join. The range is 0 to 4294967294. The default is to have no maximum set. **Note** The controller supports a maximum number of 4096 Layer 2 IGMP groups and 2048 Layer 3 IGMP groups.
Step 4	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show running-config interface** *interface-id* **Example:** `Controller# show running-config interface gigabitethernet1/0/1`	Verifies your entries.

	Command or Action	Purpose
Step 6	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring the IGMP Throttling Action

After you set the maximum number of IGMP groups that a Layer 2 interface can join, you can configure an interface to replace the existing group with the new group for which the IGMP report was received by using the **ip igmp max-groups action replace** interface configuration command. Use the **no** form of this command to return to the default, which is to drop the IGMP join report.

Follow these guidelines when configuring the IGMP throttling action:

- This restriction can be applied only to Layer 2 ports. You can use this command on a logical EtherChannel interface but cannot use it on ports that belong to an EtherChannel port group.
- When the maximum group limitation is set to the default (no maximum), entering the **ip igmp max-groups action** {**deny** | **replace**} command has no effect.
- If you configure the throttling action and set the maximum group limitation after an interface has added multicast entries to the forwarding table, the forwarding-table entries are either aged out or removed, depending on the throttling action.
  - If you configure the throttling action as **deny**, the entries that were previously in the forwarding table are not removed but are aged out. After these entries are aged out and the maximum number of entries is in the forwarding table, the controller drops the next IGMP report received on the interface.
  - If you configure the throttling action as **replace**, the entries that were previously in the forwarding table are removed. When the maximum number of entries is in the forwarding table, the controller replaces a randomly selected entry with the received IGMP report.

    To prevent the controller from removing the forwarding-table entries, you can configure the IGMP throttling action before an interface adds entries to the forwarding table.

Beginning in privileged EXEC mode, follow these steps to configure the throttling action when the maximum number of entries is in the forwarding table:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip igmp max-groups action** {**deny** | **replace**}
4. **end**
5. **show running-config interface** *interface-id*
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet 1/0/1`	Specifies the physical interface to be configured, and enters interface configuration mode. The interface can be a Layer 2 port that does not belong to an EtherChannel group or an EtherChannel interface. The interface cannot be a trunk port.
Step 3	**ip igmp max-groups action** {**deny**	**replace**} **Example:** `Controller(config-if)# ip igmp max-groups action replace`	When an interface receives an IGMP report and the maximum number of entries is in the forwarding table, specifies the action that the interface takes: • **deny—**Drops the report. • **replace—**Replaces the existing group with the new group for which the IGMP report was received.
Step 4	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show running-config interface** *interface-id* **Example:** `Controller# show running-config interface gigabitethernet1/0/1`	Verifies your entries.

	Command or Action	Purpose
Step 6	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

IGMP Filtering and Throttling Overview, on page 449

Displaying IGMP Filtering and Throttling Configuration, on page 490

Examples: Configuring Filtering and Throttling, on page 492

How to Configure IGMP Snooping

Enabling IGMP Snooping on a Controller

By default, IGMP snooping is globally enabled on the controller. When globally enabled or disabled, it is also enabled or disabled in all existing VLAN interfaces. IGMP snooping is by default enabled on all VLANs, but can be enabled and disabled on a per-VLAN basis.

Global IGMP snooping overrides the VLAN IGMP snooping. If global snooping is disabled, you cannot enable VLAN snooping. If global snooping is enabled, you can enable or disable VLAN snooping.

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

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp snooping**
3. **end**
4. **copy running-config startup-config**

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	**ip igmp snooping** **Example:** `Controller(config)# ip igmp snooping`	Globally enables IGMP snooping in all existing VLAN interfaces.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 4	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

IGMP Snooping, on page 446

Examples: Configuring IGMP Snooping, on page 491

Enabling IGMP Snooping on a VLAN Interface

Beginning in privileged EXEC mode, follow these steps to enable IGMP snooping on a VLAN interface:

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp snooping vlan** *vlan-id*
3. **end**
4. **copy running-config startup-config**

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	**ip igmp snooping vlan** *vlan-id* **Example:** `Controller(config)# ip igmp snooping vlan 7`	Enables IGMP snooping on the VLAN interface. The VLAN ID range is 1 to 1001 and 1006 to 4094. **Note** IGMP snooping must be globally enabled before you can enable VLAN snooping.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 4	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Setting the Snooping Method

Multicast-capable router ports are added to the forwarding table for every Layer 2 multicast entry. The controller learns of the ports through one of these methods:

- Snooping on IGMP queries
- Statically connecting to a multicast router port using the **ip igmp snooping mrouter** global configuration command

.

Beginning in privileged EXEC mode, follow these steps to alter the method in which a VLAN interface accesses a multicast router:

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp snooping vlan** *vlan-id* **mrouter interface** {**GigabitEthernet** | **Port-Channel** | **TenGigabitEthernet**}
3. **end**
4. **show ip igmp snooping**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ip igmp snooping vlan** *vlan-id* **mrouter interface** {**GigabitEthernet** | **Port-Channel** | **TenGigabitEthernet**} **Example:** `Controller(config)# ip igmp snooping vlan 1 mrouter interface GigabitEthernet1/0/3`	Enables IGMP snooping on a VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4094.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 4	**show ip igmp snooping** **Example:** `Controller# show ip igmp snooping`	Verifies the configuration.
Step 5	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring a Multicast Router Port

To add a multicast router port (add a static connection to a multicast router), use the **ip igmp snooping vlan mrouter** global configuration command on the controller.

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

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp snooping vlan** *vlan-id* **mrouter interface** *interface-id*
3. **end**
4. **show ip igmp snooping mrouter** [**vlan** *vlan-id*]
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ip igmp snooping vlan** *vlan-id* **mrouter interface** *interface-id* **Example:** `Controller(config)# ip igmp snooping vlan 5 mrouter interface gigabitethernet1/0/1`	Specifies the multicast router VLAN ID and 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 128.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 4	**show ip igmp snooping mrouter** [**vlan** *vlan-id*] **Example:** `Controller# show ip igmp snooping mrouter vlan 5`	Verifies that IGMP snooping is enabled on the VLAN interface.
Step 5	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring a Host Statically to Join a Group

Hosts or Layer 2 ports normally join multicast groups dynamically, but you can also statically configure a host on an interface.

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp snooping vlan** *vlan-id* **static** *ip_address* **interface** *interface-id*
3. **end**
4. **show ip igmp snooping groups**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ip igmp snooping vlan** *vlan-id* **static** *ip_address* **interface** *interface-id* **Example:** `Controller(config)# ip igmp snooping vlan 105 static 230.0.0.1 interface gigabitethernet1/0/1`	Statically configures a Layer 2 port as a member of a multicast group: • *vlan-id* is the multicast group VLAN ID. The range is 1 to 1001 and 1006 to 4094. • *ip-address* is the group IP address. • *interface-id* is the member port. It can be a physical interface or a port channel (1 to 128).
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 4	**show ip igmp snooping groups** **Example:** `Controller# show ip igmp snooping groups`	Verifies the member port and the IP address.

	Command or Action	Purpose
Step 5	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Enabling IGMP Immediate Leave

When you enable IGMP Immediate Leave, the controller immediately removes a port when it detects an IGMP Version 2 leave message on that port. You should only use the Immediate-Leave feature when there is a single receiver present on every port in the VLAN.

Note Immediate Leave is supported only on IGMP Version 2 hosts. IGMP Version 2 is the default version for the controller .

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

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp snooping vlan** *vlan-id* **immediate-leave**
3. **end**
4. **show ip igmp snooping vlan** *vlan-id*
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ip igmp snooping vlan** *vlan-id* **immediate-leave** **Example:** `Controller(config)# ip igmp snooping vlan 21 immediate-leave`	Enables IGMP Immediate Leave on the VLAN interface.

	Command or Action	Purpose
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 4	**show ip igmp snooping vlan** *vlan-id* **Example:** `Controller# show ip igmp snooping vlan 21`	Verifies that Immediate Leave is enabled on the VLAN interface.
Step 5	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring the IGMP Leave Timer

Follow these guidelines when configuring the IGMP leave timer:

- You can configure the leave time globally or on a per-VLAN basis.
- Configuring the leave time on a VLAN overrides the global setting.
- The default leave time is 1000 milliseconds.
- The IGMP configurable leave time is only supported on hosts running IGMP Version 2. IGMP version 2 is the default version for the controller.
- The actual leave latency in the network is usually the configured leave time. However, the leave time might vary around the configured time, depending on real-time CPU load conditions, network delays and the amount of traffic sent through the interface.

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp snooping last-member-query-interval** *time*
3. **ip igmp snooping vlan** *vlan-id* **last-member-query-interval** *time*
4. **end**
5. **show ip igmp snooping**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ip igmp snooping last-member-query-interval** *time* **Example:** `Controller(config)# ip igmp snooping last-member-query-interval 1000`	Configures the IGMP leave timer globally. The range is 100 to 32768 milliseconds. The default is 1000 seconds.
Step 3	**ip igmp snooping vlan** *vlan-id* **last-member-query-interval** *time* **Example:** `Controller(config)# ip igmp snooping vlan 210 last-member-query-interval 1000`	(Optional) Configures the IGMP leave time on the VLAN interface. The range is 100 to 32768 milliseconds. **Note** Configuring the leave time on a VLAN overrides the globally configured timer.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 5	**show ip igmp snooping** **Example:** `Controller# show ip igmp snooping`	(Optional) Displays the configured IGMP leave time.
Step 6	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

IGMP Configurable-Leave Timer, on page 449

Configuring the IGMP Robustness-Variable

Use the following procedure to configure the IGMP robustness variable on the controller.

The robustness variable is the integer used by IGMP snooping during calculations for IGMP messages. The robustness variable provides fine tuning to allow for expected packet loss.

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp snooping robustness-variable** *count*
3. **ip igmp snooping vlan** *vlan-id* **robustness-variable** *count*
4. **end**
5. **show ip igmp snooping**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ip igmp snooping robustness-variable** *count* **Example:** `Controller(config)# ip igmp snooping robustness-variable 3`	Configures the IGMP robustness variable. The range is 1 to 3 times. The recommended value for the robustness variable is 2. Use this command to change the value of the robustness variable for IGMP snooping from the default (2) to a specified value.
Step 3	**ip igmp snooping vlan** *vlan-id* **robustness-variable** *count* **Example:** `Controller(config)#ip igmp snooping vlan 100 robustness-variable 3`	(Optional) Configures the IGMP robustness variable on the VLAN interface. The range is 1 to 3 times. The recommended value for the robustness variable is 2. **Note** Configuring the robustness variable count on a VLAN overrides the globally configured value
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 5	**show ip igmp snooping** **Example:** `Controller# show ip igmp snooping`	(Optional) Displays the configured IGMP robustness variable count.

	Command or Action	Purpose
Step 6	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring the IGMP Last Member Query Count

To configure the number of times the controller sends IGMP group-specific or group-source-specific (with IGMP version 3) query messages in response to receiving a group-specific or group-source-specific leave message, use this command.

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp snooping last-member-query-count** *count*
3. **ip igmp snooping vlan** *vlan-id* **last-member-query-count** *count*
4. **end**
5. **show ip igmp snooping**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ip igmp snooping last-member-query-count** *count* **Example:** `Controller(config)# ip igmp snooping last-member-query-count 3`	Configures the IGMP last member query count. The range is 1 to 7 messages. The default is 2 messages.
Step 3	**ip igmp snooping vlan** *vlan-id* **last-member-query-count** *count* **Example:** `Controller(config)#ip igmp snooping vlan 100`	(Optional) Configures the IGMP last member query count on the VLAN interface. The range is 1 to 7 messages. **Note** Configuring the last member query count on a VLAN overrides the globally configured timer.

	Command or Action	Purpose
	`last-member-query-count 3`	
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 5	**show ip igmp snooping** **Example:** `Controller# show ip igmp snooping`	(Optional) Displays the configured IGMP last member query count.
Step 6	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring TCN-Related Commands

Controlling the Multicast Flooding Time After a TCN Event

You can control the time that multicast traffic is flooded after a topology change notification (TCN) event by using the **ip igmp snooping tcn flood query count** global configuration command. This command configures the number of general queries for which multicast data traffic is flooded after a TCN event. Some examples of TCN events are when the client changed its location and the receiver is on same port that was blocked but is now forwarding, and when a port went down without sending a leave message.

If you set the TCN flood query count to 1 by using the **ip igmp snooping tcn flood query count** command, the flooding stops after receiving 1 general query. If you set the count to 7, the flooding continues until 7 general queries are received. Groups are relearned based on the general queries received during the TCN event.

Beginning in privileged EXEC mode, follow these steps to configure the TCN flood query count:

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp snooping tcn flood query count** *count*
3. **end**
4. **show ip igmp snooping**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ip igmp snooping tcn flood query count** *count* **Example:** `Controller(config)# ip igmp snooping tcn flood query count 3`	Specifies the number of IGMP general queries for which the multicast traffic is flooded. The range is 1 to 10. By default, the flooding query count is 2.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 4	**show ip igmp snooping** **Example:** `Controller# show ip igmp snooping`	Verifies the TCN settings.
Step 5	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Recovering from Flood Mode

When a topology change occurs, the spanning-tree root sends a special IGMP leave message (also known as global leave) with the group multicast address 0.0.0.0. However, when you enable the **ip igmp snooping tcn query solicit** global configuration command, the controller sends the global leave message whether or not it is the spanning-tree root. When the router receives this special leave, it immediately sends general queries, which expedite the process of recovering from the flood mode during the TCN event. Leaves are always sent if the controller is the spanning-tree root regardless of this configuration command. By default, query solicitation is disabled.

Beginning in privileged EXEC mode, follow these steps to enable the controller to send the global leave message whether or not it is the spanning-tree root:

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp snooping tcn query solicit**
3. **end**
4. **show ip igmp snooping**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ip igmp snooping tcn query solicit** **Example:** `Controller(config)# ip igmp snooping tcn query solicit`	Sends an IGMP leave message (global leave) to speed the process of recovering from the flood mode caused during a TCN event. By default, query solicitation is disabled.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 4	**show ip igmp snooping** **Example:** `Controller# show ip igmp snooping`	Verifies the TCN settings.
Step 5	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Disabling Multicast Flooding During a TCN Event

When the controller receives a TCN, multicast traffic is flooded to all the ports until 2 general queries are received. If the controller has many ports with attached hosts that are subscribed to different multicast groups, this flooding might exceed the capacity of the link and cause packet loss. You can use the **ip igmp snooping tcn flood** interface configuration command to control this operation function.

Beginning in privileged EXEC mode, follow these steps to disable multicast flooding on an interface:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **no ip igmp snooping tcn flood**
4. **end**
5. **show ip igmp snooping**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet 1/0/1`	Specifies the interface to be configured, and enters interface configuration mode.
Step 3	**no ip igmp snooping tcn flood** **Example:** `Controller(config-if)# no ip igmp snooping tcn flood`	Disables the flooding of multicast traffic during a spanning-tree TCN event. By default, multicast flooding is enabled on an interface.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 5	**show ip igmp snooping** **Example:** `Controller# show ip igmp snooping`	Verifies the TCN settings.

	Command or Action	Purpose
Step 6	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring the IGMP Snooping Querier

Follow these guidelines when configuring the IGMP snooping querier:

- Configure the VLAN in global configuration mode.
- Configure an IP address on the VLAN interface. When enabled, the IGMP snooping querier uses the IP address as the query source address.
- If there is no IP address configured on the VLAN interface, the IGMP snooping querier tries to use the configured global IP address for the IGMP querier. If there is no global IP address specified, the IGMP querier tries to use the VLAN controller virtual interface (SVI) IP address (if one exists). If there is no SVI IP address, the controller uses the first available IP address configured on the controller. The first IP address available appears in the output of the **show ip interface** privileged EXEC command. The IGMP snooping querier does not generate an IGMP general query if it cannot find an available IP address on the controller.
- The IGMP snooping querier supports IGMP Versions 1 and 2.
- When administratively enabled, the IGMP snooping querier moves to the nonquerier state if it detects the presence of a multicast router in the network.
- When it is administratively enabled, the IGMP snooping querier moves to the operationally disabled state under these conditions:
  - IGMP snooping is disabled in the VLAN.
  - PIM is enabled on the SVI of the corresponding VLAN.

Beginning in privileged EXEC mode, follow these steps to enable the IGMP snooping querier feature in a VLAN:

SUMMARY STEPS

1. **configure terminal**
2. **ip igmp snooping querier**
3. **ip igmp snooping querier address** *ip_address*
4. **ip igmp snooping querier query-interval** *interval-count*
5. **ip igmp snooping querier tcn query** [**count** *count* | **interval** *interval*]
6. **ip igmp snooping querier timer expiry** *timeout*
7. **ip igmp snooping querier version** *version*
8. **end**
9. **show ip igmp snooping vlan** *vlan-id*
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ip igmp snooping querier** **Example:** `Controller(config)# ip igmp snooping querier`	Enables the IGMP snooping querier.
Step 3	**ip igmp snooping querier address** *ip_address* **Example:** `Controller(config)# ip igmp snooping querier address 172.16.24.1`	(Optional) Specifies an IP address for the IGMP snooping querier. If you do not specify an IP address, the querier tries to use the global IP address configured for the IGMP querier. **Note** The IGMP snooping querier does not generate an IGMP general query if it cannot find an IP address on the controller.
Step 4	**ip igmp snooping querier query-interval** *interval-count* **Example:** `Controller(config)# ip igmp snooping querier query-interval 30`	(Optional) Sets the interval between IGMP queriers. The range is 1 to 18000 seconds.

	Command or Action	Purpose
Step 5	**ip igmp snooping querier tcn query** [**count** *count* | **interval** *interval*] **Example:** `Controller(config)# ip igmp snooping querier tcn query interval 20`	(Optional) Sets the time between Topology Change Notification (TCN) queries. The count range is 1 to 10. The interval range is 1 to 255 seconds.
Step 6	**ip igmp snooping querier timer expiry** *timeout* **Example:** `Controller(config)# ip igmp snooping querier timer expiry 180`	(Optional) Sets the length of time until the IGMP querier expires. The range is 60 to 300 seconds.
Step 7	**ip igmp snooping querier version** *version* **Example:** `Controller(config)# ip igmp snooping querier version 2`	(Optional) Selects the IGMP version number that the querier feature uses. Select 1 or 2.
Step 8	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 9	**show ip igmp snooping vlan** *vlan-id* **Example:** `Controller# show ip igmp snooping vlan 30`	(Optional) Verifies that the IGMP snooping querier is enabled on the VLAN interface. The VLAN ID range is 1 to 1001 and 1006 to 4094.
Step 10	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Disabling IGMP Report Suppression

Note IGMP report suppression is supported only when the multicast query has IGMPv1 and IGMPv2 reports. This feature is not supported when the query includes IGMPv3 reports.

IGMP report suppression is enabled by default. When it is enabled, the controller forwards only one IGMP report per multicast router query. When report suppression is disabled, all IGMP reports are forwarded to the multicast routers.

Beginning in privileged EXEC mode, follow these steps to disable IGMP report suppression:

SUMMARY STEPS

1. **configure terminal**
2. **no ip igmp snooping report-suppression**
3. **end**
4. **show ip igmp snooping**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**no ip igmp snooping report-suppression** **Example:** `Controller(config)# no ip igmp snooping report-suppression`	Disables IGMP report suppression.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 4	**show ip igmp snooping** **Example:** `Controller# show ip igmp snooping`	Verifies that IGMP report suppression is disabled.
Step 5	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Monitoring IGMP

You can display specific statistics, such as the contents of IP routing tables, caches, and databases.

Note This release does not support per-route statistics.

You can display information to learn resource usage and solve network problems. You can also display information about node reachability and discover the routing path that packets of your device are taking through the network.

You can use any of the privileged EXEC commands in the following table to display various routing statistics.

Table 59: Commands for Displaying System and Network Statistics

Command	Purpose
ping [*group-name* | *group-address*]	Sends an ICMP Echo Request to a multicast group address.
show ip igmp filter	Displays IGMP filter information.
show ip igmp groups [*type-number* | *detail*]	Displays the multicast groups that are directly connected to the controller and that were learned through IGMP.
show ip igmp interface [*type number*]	Displays multicast-related information about an interface.
show ip igmp membership [*name/group address* | **all** | **tracked**]	Displays IGMP membership information for forwarding.
show ip igmp profile [profile_number]	Displays IGMP profile information.
show ip igmp ssm-mapping [*hostname/IP address*]	Displays IGMP SSM mapping information.
show ip igmp static-group {**class-map** [**interface** [type]]	Displays static group information.
show ip igmp vrf	Displays the selected VPN routing/forwarding instance by name.

Displaying IGMP Snooping Information

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

Table 60: Commands for Displaying IGMP Snooping Information

Command	Purpose
show ip igmp snooping detail	Displays the operational state information.
show ip igmp snooping groups [**count** | [**vlan** *vlan-id* [*A.B.C.D* | **count**]]	Displays multicast table information for the controller or about a specific parameter: • **count—**Displays the total number of groups. • **vlan—**Displays group information by VLAN ID.
show ip igmp snooping igmpv2-tracking	Displays the IGMP snooping tracking. **Note** This command displays group and IP address entries only for wireless multicast IGMP joins and not for wired IGMP joins. Wireless IP multicast must be enabled for this command to display.
show ip igmp snooping mrouter [**vlan** *vlan-id*]	Displays information on dynamically learned and manually configured multicast router interfaces. **Note** When you enable IGMP snooping, the controller automatically learns the interface to which a multicast router is connected. These are dynamically learned interfaces. (Optional) Enter **vlan** *vlan-id* to display information for a single VLAN.
show ip igmp snooping querier [**detail** | **vlan** *vlan-id*]	Displays information about the IP address and receiving port for the most-recently received IGMP query messages in the VLAN. (Optional) Enter **detail** to display the detailed IGMP querier information in a VLAN. (Optional) Enter **vlan** *vlan-id* to display information for a single VLAN.
show ip igmp snooping [**vlan** *vlan-id* [**detail**]]	Displays the snooping configuration information for all VLANs on the controller or for a specified VLAN. (Optional) Enter **vlan** *vlan-id* to display information for a single VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4094.
show ip igmp snooping wireless mgid	Displays wireless-related events.

Displaying IGMP Filtering and Throttling Configuration

You can display IGMP profile characteristics, and you can display the IGMP profile and maximum group configuration for all interfaces on the controller or for a specified interface. You can also display the IGMP throttling configuration for all interfaces on the controller or for a specified interface.

Table 61: Commands for Displaying IGMP Filtering and Throttling Configuration

Command	Purpose
show ip igmp profile [*profile number*]	Displays the specified IGMP profile or all the IGMP profiles defined on the controller.
show running-config [**interface** *interface-id*]	Displays the configuration of the specified interface or the configuration of all interfaces on the controller, including (if configured) the maximum number of IGMP groups to which an interface can belong and the IGMP profile applied to the interface.

Related Topics

Configuring the IGMP Throttling Action, on page 467

IGMP Filtering and Throttling Overview, on page 449

Configuration Examples for IGMP

Example: Configuring the Controller as a Member of a Multicast Group

This example shows how to enable the controller to join multicast group 255.2.2.2:

```
Controller(config)# interface gigabitethernet1/0/1
Controller(config-if)# ip igmp join-group 255.2.2.2
Controller(config-if)#
```

Related Topics

Configuring the Controller as a Member of a Group, on page 452

Joining a Multicast Group, on page 446

Configuring the Controller as a Member of a Group, on page 452

IP Multicast Group Addresses, on page 444

Example: Controlling Access to Multicast Groups

To limit the number of joins on the interface, configure the port for filter which associates with the IGMP profile.

```
Controller# configure terminal
Controller(config)# ip igmp profile 10
Controller(config-igmp-profile)# ?

IGMP profile configuration commands:
deny matching addresses are denied
exit Exit from igmp profile configuration mode
no Negate a command or set its defaults
permit matching addresses are permitted
range add a range to the set

Controller(config-igmp-profile)# range 172.16.5.1
Controller(config-igmp-profile)# exit
Controller(config)#
Controller(config)# interface gigabitEthernet 2/0/10
Controller(config-if)# ip igmp filter 10
```

Examples: Configuring IGMP Snooping

This example shows how to enable a static connection to a multicast router:

```
Controller# configure terminal
Controller(config)# ip igmp snooping vlan 200 mrouter interface gigabitethernet1/0/2
Controller(config)# end
```

This example shows how to statically configure a host on a port:

```
Controller# configure terminal
Controller(config)# ip igmp snooping vlan 105 static 224.2.4.12 interface gigabitethernet1/0/1
Controller(config)# end
```

This example shows how to enable IGMP Immediate Leave on VLAN 130:

```
Controller# configure terminal
Controller(config)# ip igmp snooping vlan 130 immediate-leave
Controller(config)# end
```

This example shows how to set the IGMP snooping querier source address to 10.0.0.64:

```
Controller# configure terminal
Controller(config)# ip igmp snooping querier 10.0.0.64
Controller(config)# end
```

This example shows how to set the IGMP snooping querier maximum response time to 25 seconds:

```
Controller# configure terminal
Controller(config)# ip igmp snooping querier query-interval 25
Controller(config)# end
```

This example shows how to set the IGMP snooping querier timeout to 60 seconds:

```
Controller# configure terminal
Controller(config)# ip igmp snooping querier timeout expiry 60
```

```
Controller(config)# end
```

This example shows how to set the IGMP snooping querier feature to Version 2:

```
Controller# configure terminal
Controller(config)# no ip igmp snooping querier version 2
Controller(config)# end
```

Related Topics

Enabling IGMP Snooping on a Controller, on page 469

IGMP Snooping, on page 446

Examples: Configuring Filtering and Throttling

This example shows how to create IGMP profile 4 allowing access to the single IP multicast address and how to verify the configuration. If the action was to deny (the default), it would not appear in the **show ip igmp profile** output display.

```
Controller(config)# ip igmp profile 4
Controller(config-igmp-profile)# permit
Controller(config-igmp-profile)# range 229.9.9.0
Controller(config-igmp-profile)# end
Controller# show ip igmp profile 4
IGMP Profile 4
    permit
    range 229.9.9.0 229.9.9.0
```

This example shows how to apply IGMP profile 4 to a port:

```
Controller(config)# interface gigabitethernet1/0/2
Controller(config-if)# ip igmp filter 4
Controller(config-if)# end
```

This example shows how to limit to 25 the number of IGMP groups that a port can join:

```
Controller(config)# interface gigabitethernet1/0/2
Controller(config-if)# ip igmp max-groups 25
Controller(config-if)# end
```

Related Topics

Configuring the IGMP Throttling Action, on page 467

IGMP Filtering and Throttling Overview, on page 449

Example: Interface Configuration as a Routed Port

This example shows how to configure an interface on the controller as a routed port. This configuration is required on the interface for several IP multicast routing configuration procedures that require running the **no switchport** command.

```
Controller configure terminal
Controller(config)#  interface GigabitEthernet1/0/9
Controller(config-if)# description interface to be use as routed port
Controller(config-if)# no switchport
Controller(config-if)# ip address 20.20.20.1 255.255.255.0
```

```
Controller(config-if)# ip pim sparse-dense-mode
Controller(config-if)# ip igmp join-group 224.1.2.3 source 15.15.15.2
Controller(config-if)# end
Controller# configure terminal
Controller# show run interface gigabitEthernet 1/0/9

Current configuration : 166 bytes
!
interface GigabitEthernet1/0/9
 no switchport
 ip address 20.20.20.1 255.255.255.0
 ip pim sparse-dense-mode
 ip igmp static-group 224.1.2.3 source 15.15.15.2
end
```

Example: Interface Configuration as an SVI

This example shows how to configure an interface on the controller as an SVI. This configuration is required on the interface for several IP multicast routing configuration procedures that require running the **no switchport** command.

```
Controller(config)# interface vlan 150
Controller(config-if)# ip address 20.20.20.1 255.255.255.0
Controller(config-if)# ip pim sparse-dense-mode
Controller(config-if)# ip igmp join-group 224.1.2.3 source 15.15.15.2
Controller(config-if)# end
Controller# configure terminal
Controller(config)#  ip igmp snooping vlan 20 static 224.1.2.3
interface gigabitEthernet 1/0/9
Controller# show run interface vlan 150

Current configuration : 137 bytes
!
interface Vlan150
 ip address 20.20.20.1 255.255.255.0
 ip pim sparse-dense-mode
 ip igmp static-group 224.1.2.3 source 15.15.15.2
end
```

Where to Go Next for IGMP

You can configure the following:

- Wireless Multicast

Additional References

Standards and RFCs

Standard/RFC	Title
RFC 1112	*Host Extensions for IP Multicasting*
RFC 2236	*Internet Group Management Protocol, Version 2*

MIBs

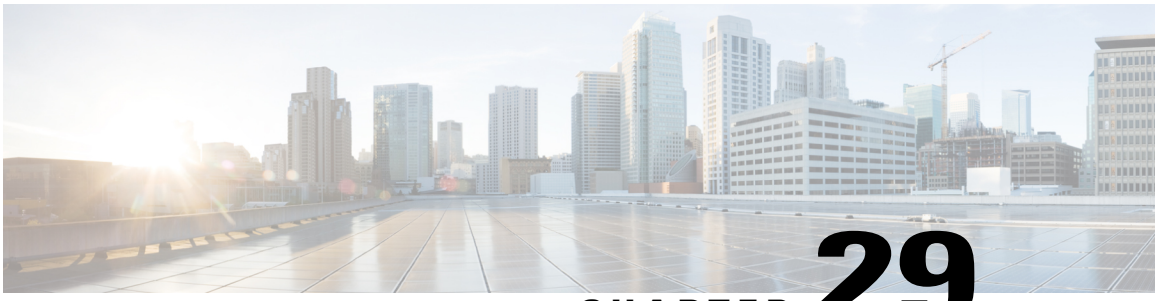
MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

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 Information for IGMP

Feature Name	Releases	Feature Information
IGMP, Versions 1, 2, and 3 support		To participate in IP multicasting, multicast hosts, routers, and multilayer controllers must have the Internet Group Management Protocol (IGMP) operating. This protocol defines both querier and host roles. **Note** IGMP, Version 2 is the default version for the controller.
IGMP Snooping support		Layer 2 controllers use IGMP snooping to constrain the flooding of multicast traffic by dynamically configuring Layer 2 interfaces, so that multicast traffic is forwarded to only those interfaces associated with IP multicast devices.



CHAPTER 29

Configuring Wireless Multicast

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Wireless Multicast

- The IP multicast routing must be enabled. The default routes should be available in the the device. After performing these tasks, the device can then forward multicast packets and can populate its multicast routing table. The network should be multicast enabled to configure mutlicast mode.
- To participate in IP multicasting, the multicast hosts, routers, and multilayer switches must have IGMP operating.
- When enabling multicast mode on the controller, a CAPWAP multicast group address should also be configured. Access points listens to the CAPWAP multicast group using IGMP.

Restrictions for Configuring Wireless Multicast

The following are the restrictions for configuring IP multicast routing:

- Access points in monitor mode, sniffer mode, or rogue detector mode do not join the CAPWAP multicast group address.
- The CAPWAP multicast group configured on the controllers should be different for different controllers.
- Multicast routing should not be enabled for the management interface.

Information about Wireless Multicast

If the network supports packet multicasting, the multicast method can be configured that the controller uses. The controller performs multicasting in two modes:

- Unicast mode - In this mode, the controller unicasts every multicast packet to every access point associated to the controller. This mode is inefficient but might be required on networks that do not support multicasting.
- Multicast mode - In the multicast mode, the controller sends multicast packets to a CAPWAP multicast group. This method reduces overhead on the controller processor and shifts the work of packet replication to the network, which is much more efficient than the unicast method.

When the multicast mode is enabled and the controller receives a multicast packet from the wired LAN, the controller encapsulates the packet using CAPWAP and forwards the packet to the CAPWAP multicast group address. The controller always uses the management VLAN for sending multicast packets. Access points in the multicast group receive the packet and forward it to all the BSSIDs mapped to the VLAN on which clients receive multicast traffic.

The controller supports all the capabilities of v1 including Multicast Listener Discovery (MLD) v1 snooping but the v2 and v3 capabilities are limited. This feature keeps track and delivers IPv6 multicast flows to the clients that request them. To support IPv6 multicast, the Global Multicast Mode should be enabled.

Internet Group Management Protocol (IGMP) snooping is introduced to better direct multicast packets. When this feature is enabled, the controller snooping gathers IGMP reports from the clients, processes them, creates unique multicast group IDs (MGIDs) based on the Layer 3 multicast address and the VLAN number, and sends the IGMP reports to the IGMP courier. The controller then updates the access point MGID table on the access point with the client MAC address. When the controller receives multicast traffic for a particular multicast group, it forwards it to all the access points, but only those access points that have active clients listening or subscribed to that multicast group send multicast traffic on that particular WLAN. IP packets are forwarded with an MGID that is unique for an ingress VLAN and the destination multicast group. Layer 2 multicast packets are forwarded with an MGID that is unique for the ingress VLAN.

MGID is a 14 bit value filled in the 16 bit reserved field of wireless info in capwap header. The remaining 2 bits should be set to zero.

Information about Multicast Optimization

Multicast used to be based on the group of the multicast addresses and the VLAN as one entity, MGID. With VLAN group, there is a possibility that duplicate packets might increase. Using the VLAN group feature, every client listens to the multicast stream on a different VLAN. As a result, the controller creates different

MGIDs for each multicast address and VLAN. Therefore, in a worst case situation, the upstream router sends one copy for each VLAN which results in as many copies as the number of VLANs in the group. Since the WLAN is still the same for all clients, multiple copies of the multicast packet are sent over the air. To suppress the duplication of a multicast stream on the wireless medium, between the controller and the access points, the multicast optimization feature can be used.

Multicast optimization enables in creating a multicast VLAN which can be used for multicast traffic. One of the VLANs in the controller can be configured as a multicast VLAN where multicast groups are registered. The clients are allowed to listen to a multicast stream on the multicast VLAN. The MGID is generated using the mulicast VLAN and multicast IP addresses. If multiple clients on different VLANs of the same WLAN are listening to a single multicast IP address, a single MGID is generated. The controller makes sure that all multicast streams from the clients on this VLAN group always go out on the multicast VLAN to ensure that the upstream router has one entry for all the VLANs of the VLAN group. Only one multicast stream hits the VLAN group even if the clients are on different VLANs. Therefore, the multicast packets that are sent out over the air is just one stream.

How to Configure Wireless Multicast

Configuring Wireless Multicast-MCMC Mode

SUMMARY STEPS

1. **configure terminal**
2. **wireless multicast**
3. **ap capwap multicast ipaddr**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global command mode.

Step 2	**wireless multicast**
	
	Example:
	`Controller(config)# wireless multicast`
	
	`Controller(config)# no wireless multicast`
		Enables the multicast traffic for wireless clients. The default value is disable. Add **no** in the command to disable the multicast traffic for wireless clients.

Step 3	**ap capwap multicast ipaddr**
	
	Example:
	`Controller(config)# ap capwap multicast 231.1.1.1`
	
	`Controller(config)# no ap capwap multicast 231.1.1.1`
		Enables the forwarding mode in Multicast. Add **no** in the command to disable the multicast mode.

	Command or Action	Purpose
Step 4	**end** **Example:** `Controller(config)# end`	Exits the configuration mode. Alternatively, press CTRL+Z to exit the configuration mode.

Configuring Wireless Multicast-MCUC Mode

SUMMARY STEPS

1. **configure terminal**
2. **wireless multicast**
3. **no ap capwap multicast** *ipaddr*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global command mode.
Step 2	**wireless multicast** **Example:** `Controller(config)# wireless multicast`	Enables the multicast traffic for wireless clients. The default value is disable. Add **no** in the command to disable the multicast traffic for wireless clients.
Step 3	**no ap capwap multicast** *ipaddr* **Example:** `Controller(config)# no ap capwap multicast 231.1.1.1`	Enables forwarding mode in Multicast. Add **no** in the command to disable the multicast mode.
Step 4	**end** **Example:** `Controller(config)# end`	Exits the configuration mode. Alternatively, press CTRL+Z to exit the configuration mode.

Configuring Non-IP Wireless Multicast

SUMMARY STEPS

1. **configure terminal**
2. **wireless multicast non-ip**
3. **wireless multicast non-ip** *vlanid*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global command mode.

Step 2	**wireless multicast non-ip**
	
	Example:
	`Controller(config)# wireless multicast non-ip`
	
	`Controller(config)# no wireless multicast non-ip`
		Enables non-ip multicast in all VLANs. Default value is enable. This requires `wireless multicast` to be enabled for the traffic to pass. Add **no** in the command to disable all the non-ip multicast in VLANs.

Step 3	**wireless multicast non-ip** *vlanid*
	
	Example:
	`Controller(config)# wireless multicast non-ip 5`
	
	`Controller(config)# no wireless multicast non-ip 5`
		Enables non-ip multicast per VLAN. Default value is enable. This requires both `wireless multicast` and `wireless multicast non-ip` to be enabled for traffic to pass. Add **no** in the command to disable the non-ip multicast per VLAN.

Step 4	**end** **Example:** `Controller(config)# end`	Exits the configuration mode. Alternatively, press CTRL+Z to exit the configuration mode.

Configuring Wireless Broadcast

SUMMARY STEPS

1. **configure terminal**
2. **wireless broadcast**
3. **wireless broadcast vlan** *vlanid*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global command mode.

Step 2	**wireless broadcast**
	
	Example:
	`Controller(config)# wireless broadcast`
	
	`Controller(config)# no wireless broadcast`
		Enables broadcast packets for wireless clients. Default value is disable. Enabling **wireless broadcast** enables broadcast traffic for each VLAN. Add **no** in the command to disable broadcasting packets.

Step 3	**wireless broadcast vlan** *vlanid*
	
	Example:
	`Controller(config)# wireless broadcast vlan 3`
	
	`Controller(config)# no wireless broadcast vlan 3`
		Enables broadcast packets for single VLAN. Default value is enable. This requires `wireless broadcast` to be enabled for broadcasting. Add **no** in the command to disable the broadcast traffic for each VLAN.

Step 4	**end** **Example:** `Controller(config)# end`	Exits the configuration mode. Alternatively, press CTRL+Z to exit the configuration mode.

Configuring IP Multicast VLAN for WLAN

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan_name*
3. **shutdown**
4. **ip multicast vlan** {*vlan_name vlan_id*}
5. **no shutdown**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global command mode.
Step 2	**wlan** *wlan_name* **Example:** `Controller(config)# wlan test1`	Enters the configuration mode to configure various parameters in the wireless-LAN network.
Step 3	**shutdown** **Example:** `Controller(config-wlan)# shutdown`	Disables WLAN.

Step 4	**ip multicast vlan** {*vlan_name vlan_id*}
	
	Example:
	`Controller(config-wlan)# ip multicast vlan 5`
	
	`Controller(config-wlan)# no ip multicast vlan 5`
		Configures multicast VLAN for WLAN. Add **no** in the command to disable the multicast VLAN for WLAN.

Step 5	**no shutdown** **Example:** `Controller(config-wlan)# no shutdown`	Enables the disabled WLAN.
Step 6	**end** **Example:** `Controller(config)# end`	Exits the configuration mode. Alternatively, press CTRL+Z to exit the configuration mode.

Monitoring Wireless Multicast

Table 62: Commands for Monitoring Wireless Multicast

Commands	Description
show wireless multicast	Displays the multicast status and ap multicast mode, each vlans broadcast/non-ip multicast status.
show wireless multicast group summary	Displays all (Source, Group and VLAN) list and the corresponding mgid value.
show wireless multicast [**source** *source*] **group** *group* **vlan** *vlanid*	Displays details of the given (S,G,V) and shows all the clients associated to it and their mc2uc status.
show ip igmp snooping wireless mcast-spi-count	Displays the statistics of number of multicast SPIs Per MGID sent to Controller.
show ip igmp snooping wireless mgid	Displays the MGID mappings.
show ip igmp snooping igmpv2-tracking	Displays the client-to-SGV mappings and SGV-to-client mappings.
show ip igmp snooping querier vlan *vlanid*	Displays IGMP querier information for the specified VLAN.
show ip igmp snooping querier detail	Displays detailed IGMP querier information of all the VLANs.
show ipv6 mld snooping querier vlan *vlanid*	Displays MLD querier information for the specified VLAN.

Where to Go Next for Wireless Multicast

- IP Multicast feature support

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

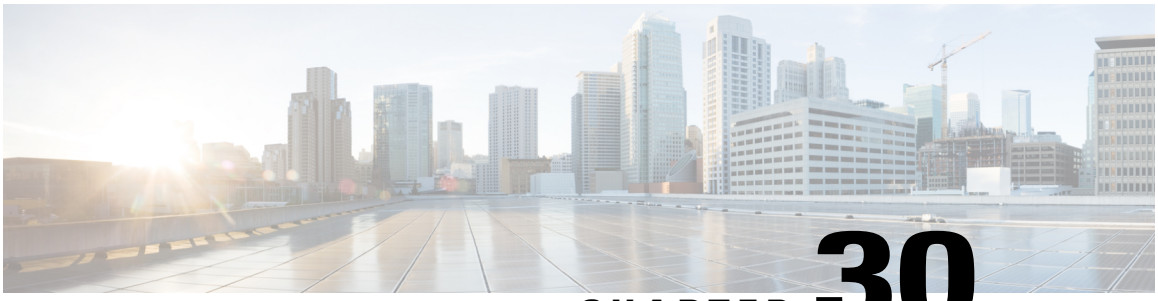
MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

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

PART VII

Security



CHAPTER 30

Preventing Unauthorized Access

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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.

Preventing Unauthorized Access

You can prevent unauthorized users from reconfiguring your Catalyst 3850 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.

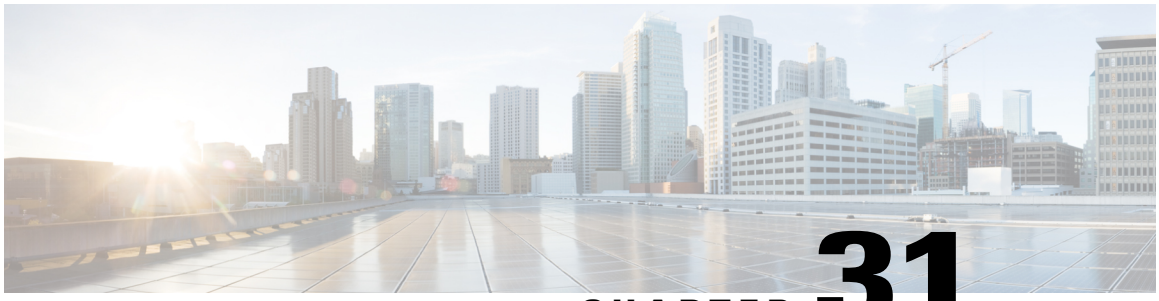
To prevent unauthorized access into your switch, you should configure one or more of these security features:

- 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.

Related Topics

Configuring Username and Password Pairs, on page 518
TACACS+ and Switch Access, on page 527
Setting a Telnet Password for a Terminal Line, on page 516



CHAPTER 31

Controlling Switch Access with Passwords and Privilege Levels

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Passwords and Privileges

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.

Related Topics

Disabling Password Recovery, on page 515
Password Recovery, on page 511

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 63: 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.

Related Topics

Protecting Enable and Enable Secret Passwords with Encryption, on page 513
Example: Protecting Enable and Enable Secret Passwords with Encryption, on page 523

Password Recovery

By default, any end user with physical access to the Catalyst 3850 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.

Related Topics

Disabling Password Recovery, on page 515
Restrictions for Controlling Switch Access with Passwords and Privileges, on page 509

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. For more information on doing this, see Related Topics.

Related Topics

Setting a Telnet Password for a Terminal Line, on page 516
Example: Setting a Telnet Password for a Terminal Line, on page 523

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.

Related Topics

Configuring Username and Password Pairs, on page 518

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.

Related Topics

Setting the Privilege Level for a Command, on page 519
Example: Setting the Privilege Level for a Command, on page 523
Changing the Default Privilege Level for Lines, on page 520
Logging into and Exiting a Privilege Level, on page 521

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. Beginning in privileged EXEC mode, follow these steps to set or change a static enable password:

SUMMARY STEPS

1. **configure terminal**
2. **enable password** *password*
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**enable password** *password* **Example:** `Controller(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: Enter **abc**. Enter **Crtl-v**. 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 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Related Topics

Example: Setting or Changing a Static Enable Password, on page 522

Protecting Enable and Enable Secret Passwords with Encryption

Beginning in privileged EXEC mode, 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. **configure terminal**
2. Use one of the following:
   - **enable password** [**level** *level*] {*password* | *encryption-type encrypted-password*}
   - **enable secret** [**level** *level*] {*password* | *encryption-type encrypted-password*}
3. **service password-encryption**
4. **end**

DETAILED STEPS

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

Step 2	Use one of the following:
	• **enable password** [**level** *level*] {*password* | *encryption-type encrypted-password*}
	• **enable secret** [**level** *level*] {*password* | *encryption-type encrypted-password*}
	
	Example:
	`Controller(config)# enable password example102`
	or
	`Controller(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.

	Command or Action	Purpose
Step 3	**service password-encryption** **Example:** `Controller(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.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Related Topics

Additional Password Security, on page 510

Example: Protecting Enable and Enable Secret Passwords with Encryption, on page 523

Disabling Password Recovery

Beginning in privileged EXEC mode, 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. **configure terminal**
2. **no service password-recovery**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**no service password-recovery** **Example:** `Controller(config)# no service password-recovery`	Disables password recovery. This setting is saved in an area of the flash memory that is accessible by the boot loader and the Cisco IOS image, but it is not part of the file system and is not accessible by any user.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

What to Do Next

To re-enable password recovery, use the **service password-recovery** global configuration command.

Related Topics

Password Recovery, on page 511

Restrictions for Controlling Switch Access with Passwords and Privileges, on page 509

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**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Controller> 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:** `Controller# configure terminal`	Enters global configuration mode.
Step 3	**line vty 0 15** **Example:** `Controller(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:** `Controller(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:** `Controller(config-line)# end`	Returns to privileged EXEC mode.

Related Topics

Preventing Unauthorized Access, on page 507
Terminal Line Telnet Configuration, on page 511

Example: Setting a Telnet Password for a Terminal Line, on page 523

Configuring Username and Password Pairs

Beginning in privileged EXEC mode, follow these steps to configure username and password pairs:

SUMMARY STEPS

1. **configure terminal**
2. **username** *name* [**privilege** *level*] {**password** *encryption-type password*}
3. Use one of the following:
   - **line console 0**
   - **line vty 0 15**
4. **login local**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**username** *name* [**privilege** *level*] {**password** *encryption-type password*} **Example:** `Controller(config)# username adamsample privilege 1 password secret456`	Sets the username, privilege level, and password 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. 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.
Step 3	Use one of the following: • **line console 0** • **line vty 0 15**	Enters line configuration mode, and configures the console port (line 0) or the VTY lines (line 0 to 15).

	Command or Action	Purpose

	Example:
	`Controller(config)# line console 0`
	or
	`Controller(config)# line vty 15`	

Step 4	**login local** **Example:** `Controller(config-line)# login local`	Enables local password checking at login time. Authentication is based on the username specified in Step 2.
Step 5	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Related Topics

Preventing Unauthorized Access, on page 507
Username and Password Pairs, on page 511

Setting the Privilege Level for a Command

Beginning in privileged EXEC mode, follow these steps to set the privilege level for a command:

SUMMARY STEPS

1. **configure terminal**
2. **privilege** *mode* **level** *level command*
3. **enable password level** *level password*
4. **end**

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	**privilege** *mode* **level** *level command* **Example:** `Controller(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 3	**enable password level** *level password* **Example:** `Controller(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 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Related Topics

Privilege Levels, on page 512

Example: Setting the Privilege Level for a Command, on page 523

Changing the Default Privilege Level for Lines

Beginning in privileged EXEC mode, follow these steps to change the default privilege level for the specified line:

SUMMARY STEPS

1. **configure terminal**
2. **line vty** *line*
3. **privilege level** *level*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**line vty** *line* **Example:** `Controller(config)# line vty 10`	Selects the virtual terminal line on which to restrict access.
Step 3	**privilege level** *level* **Example:** `Controller(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 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

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.

Related Topics

Privilege Levels, on page 512

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:** `Controller> 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:** `Controller# 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.

Related Topics

Privilege Levels, on page 512

Monitoring Switch Access

Table 64: 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):

```
Controller(config)# enable password l1u2c3k4y5
```

Related Topics

Setting or Changing a Static Enable Password, on page 512

Example: Protecting Enable and Enable Secret Passwords with Encryption

This example shows how to configure the encrypted password *1FaD0$Xyti5Rkls3LoyxzS8* for privilege level 2:

```
Controller(config)# enable secret level 2 5 $1$FaD0$Xyti5Rkls3LoyxzS8
```

Related Topics

Protecting Enable and Enable Secret Passwords with Encryption, on page 513
Additional Password Security, on page 510

Example: Setting a Telnet Password for a Terminal Line

This example shows how to set the Telnet password to *let45me67in89*:

```
Controller(config)# line vty 10
Controller(config-line)# password let45me67in89
```

Related Topics

Setting a Telnet Password for a Terminal Line, on page 516
Terminal Line Telnet Configuration, on page 511

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:

```
Controller(config)# privilege exec level 14 configure
Controller(config)# enable password level 14 SecretPswd14
```

Related Topics

Setting the Privilege Level for a Command, on page 519
Privilege Levels, on page 512

Additional References

Related Documents

Related Topic	Document Title

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

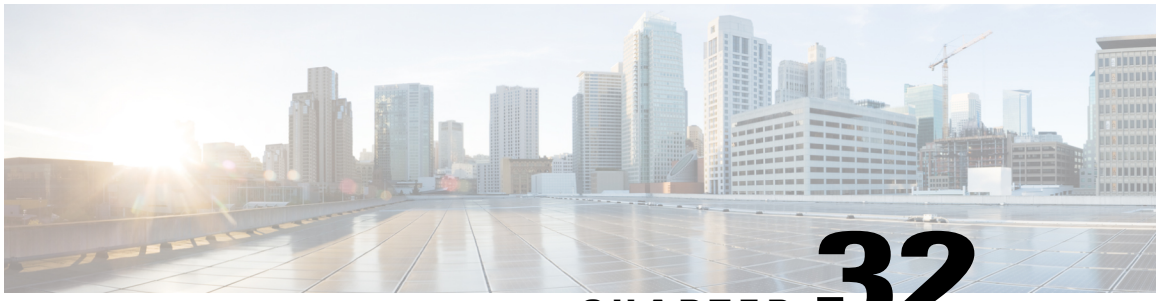
Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, 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 for Setting Passwords and Privilege Levels

Table 65: Feature Information for Setting Passwords and Privilege Levels

Feature Name	Releases	Feature Information

-



CHAPTER 32

Configuring TACACS+

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Catalyst 3850 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+.

Related Topics

TACACS+ Overview, on page 527
TACACS+ Operation, on page 529
How to Configure TACACS+, on page 531
Method List Description, on page 530
Configuring TACACS+ Login Authentication, on page 533
TACACS+ Login Authentication, on page 530
Configuring TACACS+ Authorization for Privileged EXEC Access and Network Services, on page 535
TACACS+ Authorization for Privileged EXEC Access and Network Services, on page 530

Restrictions for Controlling Switch Access with TACACS+

The following are restrictions for controlling switch access with 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.

The switch supports TACACS+ for IPv6. Information is in the "TACACS+ Over an IPv6 Transport" section of the "Implementing ADSL for IPv6" chapter in the *Cisco IOS XE IPv6 Configuration Guide, Release 2*.

For information about configuring this feature, see the "Configuring TACACS+ over IPv6" section of the "Implementing ADSL for IPv6" chapter in the *Cisco IOS XE IPv6 Configuration Guide, Release 2*.

Note For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Security Command Reference, Release 12.4* and the *Cisco IOS IPv6 Command Reference*.

Related Topics

Preventing Unauthorized Access, on page 507
Configuring the Switch for Local Authentication and Authorization, on page 582
SSH Servers, Integrated Clients, and Supported Versions, on page 589

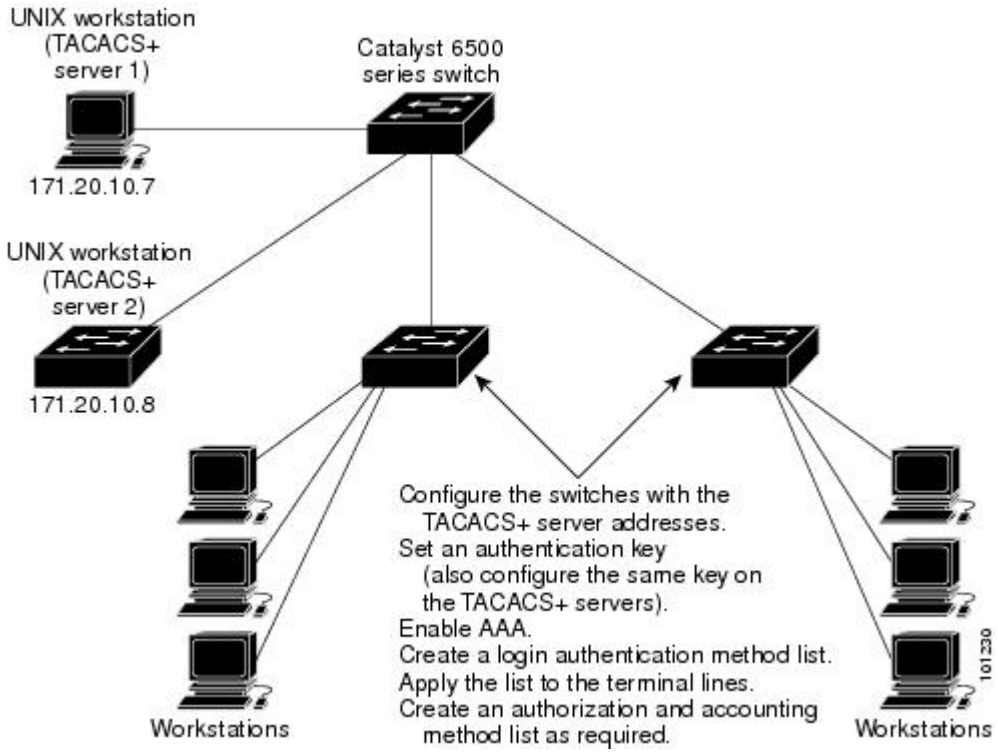
TACACS+ Overview

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 12: 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.

Related Topics

Prerequisites for Controlling Switch Access with Terminal Access Controller Access Control System Plus (TACACS+), on page 525

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

Related Topics

Prerequisites for Controlling Switch Access with Terminal Access Controller Access Control System Plus (TACACS+), on page 525

Method List Description

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.

Related Topics

How to Configure TACACS+, on page 531
Prerequisites for Controlling Switch Access with Terminal Access Controller Access Control System Plus (TACACS+), on page 525

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.

Related Topics

Identifying the TACACS+ Server Host and Setting the Authentication Key, on page 531

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.

Related Topics

Configuring TACACS+ Login Authentication, on page 533
Prerequisites for Controlling Switch Access with Terminal Access Controller Access Control System Plus (TACACS+), on page 525

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.

Related Topics

Configuring TACACS+ Authorization for Privileged EXEC Access and Network Services, on page 535
Prerequisites for Controlling Switch Access with Terminal Access Controller Access Control System Plus (TACACS+), on page 525

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.

Related Topics

Starting TACACS+ Accounting, on page 536

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+.

Related Topics

Method List Description, on page 530
Prerequisites for Controlling Switch Access with Terminal Access Controller Access Control System Plus (TACACS+), on page 525

Identifying the TACACS+ Server Host and Setting the Authentication Key

Beginning in privileged EXEC mode, follow these steps to identify the TACACS+ server host and set the authentication key:

SUMMARY STEPS

1. **configure terminal**
2. **tacacs-server host** *hostname*
3. **aaa new-model**
4. **aaa group server tacacs+** *group-name*
5. **server** *ip-address*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**tacacs-server host** *hostname* **Example:** `Controller(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 3	**aaa new-model** **Example:** `Controller(config)# aaa new-model`	Enables AAA.
Step 4	**aaa group server tacacs+** *group-name* **Example:** `Controller(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 5	**server** *ip-address* **Example:** `Controller(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 2.
Step 6	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Related Topics

TACACS+ Configuration Options, on page 530

Configuring TACACS+ Login Authentication

Beginning in privileged EXEC mode, 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. **configure terminal**
2. **aaa new-model**
3. **aaa authentication login** {**default** | *list-name*} *method1* [*method2...*]
4. **line** [**console** | **tty** | **vty**] *line-number* [*ending-line-number*]
5. **login authentication** {**default** | *list-name*}
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**aaa new-model** **Example:** `Controller(config)# aaa new-model`	Enables AAA.

	Command or Action	Purpose
Step 3	**aaa authentication login** {**default** | *list-name*} *method1* [*method2...*] **Example:** `Controller(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: • *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. For more information, see the Identifying the TACACS+ Server Host and Setting the Authentication Key, on page 531. • *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 4	**line** [**console** | **tty** | **vty**] *line-number* [*ending-line-number*] **Example:** `Controller(config)# line 2 4`	Enters line configuration mode, and configures the lines to which you want to apply the authentication list.
Step 5	**login authentication** {**default** | *list-name*} **Example:** `Controller(config-line)# login`	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.

	Command or Action	Purpose
	`authentication default`	
Step 6	**end** **Example:** `Controller(config-line)# end`	Returns to privileged EXEC mode.

Related Topics

TACACS+ Login Authentication, on page 530

Prerequisites for Controlling Switch Access with Terminal Access Controller Access Control System Plus (TACACS+), on page 525

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.

The **aaa authorization exec tacacs+ local** command sets these authorization parameters:

- Use TACACS+ for privileged EXEC access authorization if authentication was performed by using TACACS+.
- Use the local database if authentication was not performed by using TACACS+.

Note Authorization is bypassed for authenticated users who log in through the CLI even if authorization has been configured.

SUMMARY STEPS

1. **configure terminal**
2. **aaa authorization network tacacs+**
3. **aaa authorization exec tacacs+**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**aaa authorization network tacacs+** **Example:** `Controller(config)# aaa authorization network tacacs+`	Configures the switch for user TACACS+ authorization for all network-related service requests.
Step 3	**aaa authorization exec tacacs+** **Example:** `Controller(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 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Related Topics

TACACS+ Authorization for Privileged EXEC Access and Network Services, on page 530

Prerequisites for Controlling Switch Access with Terminal Access Controller Access Control System Plus (TACACS+), on page 525

Starting TACACS+ Accounting

Beginning in privileged EXEC mode, follow these steps to start TACACS+ Accounting:

SUMMARY STEPS

1. **configure terminal**
2. **aaa accounting network start-stop tacacs+**
3. **aaa accounting exec start-stop tacacs+**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**aaa accounting network start-stop tacacs+** **Example:** `Controller(config)# aaa accounting network start-stop tacacs+`	Enables TACACS+ accounting for all network-related service requests.
Step 3	**aaa accounting exec start-stop tacacs+** **Example:** `Controller(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 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

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.

Related Topics

TACACS+ Accounting, on page 531

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 66: Commands for Displaying TACACS+ Information

show tacacs	Displays TACACS+ server statistics.

Additional References

Related Documents

Related Topic	Document Title

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

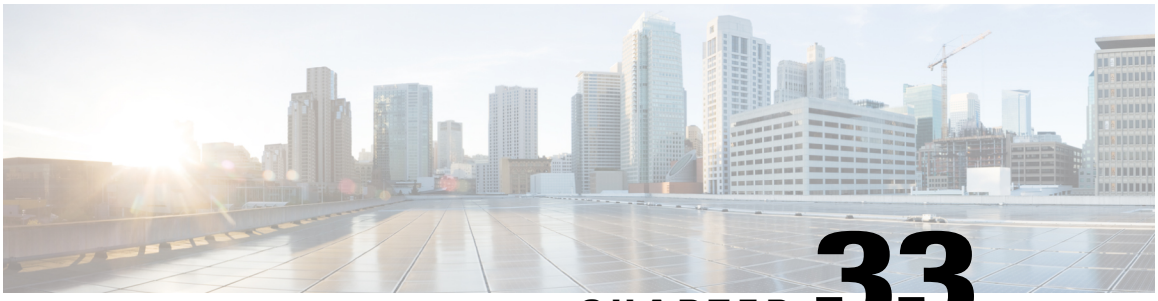
Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, 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 for TACACS+

This table lists the features in this module and provides links to specific configuration information.

Table 67: Feature Information for TACACS+

Feature Name	Releases	Feature Information



CHAPTER 33

Configuring RADIUS

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Catalyst 3850 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.
- A redundant connection between a switch stack and the RADIUS server is recommended. This is to help ensure that the RADIUS server remains accessible in case one of the connected stack members is removed from the switch stack.

For RADIUS operation:

- Users must first successfully complete RADIUS authentication before proceeding to RADIUS authorization, if it is enabled.

Related Topics

RADIUS and Switch Access, on page 543
RADIUS Operation, on page 544

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

RADIUS Overview, on page 543

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.

The switch supports RADIUS for IPv6. Information is in the "RADIUS Over IPv6" section of the "Implementing ADSL for IPv6" chapter in the *Cisco IOS XE IPv6 Configuration Guide, Release 2*. For information about configuring this feature, see the "Configuring the NAS" section in the "Implementing ADSL for IPv6" chapter in the *Cisco IOS XE IPv6 Configuration Guide, Release 2*.

Note For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Security Command Reference, Release 12.4* and the *Cisco IOS IPv6 Command Reference*.

Related Topics

Prerequisites for Controlling Switch Access with RADIUS, on page 541

Configuring the Switch for Local Authentication and Authorization, on page 582

SSH Servers, Integrated Clients, and Supported Versions, on page 589

RADIUS Overview

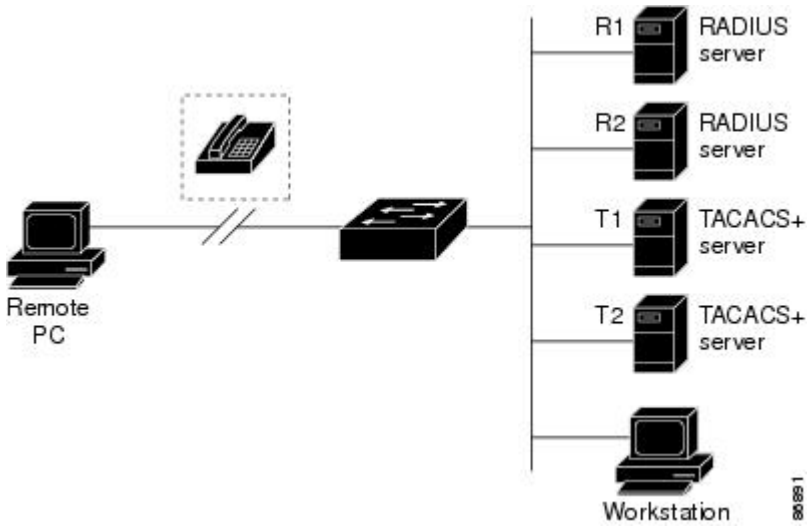
RADIUS is a distributed client/server system that secures networks against unauthorized access. RADIUS clients run on supported Cisco routers and switches. Clients send authentication requests to a central RADIUS server, which contains all user authentication and network service access information.

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 13: Transitioning from RADIUS to TACACS+ Services



Related Topics

Restrictions for Controlling Switch Access with RADIUS, on page 542

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

Prerequisites for Controlling Switch Access with RADIUS, on page 541

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 the 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 the Configuring Switch-Based Authentication chapter in the *Catalyst 3750 Switch Software Configuration Guide, 12.2(50)SE*.
- Accounting—refer to the “Starting RADIUS Accounting” section in the Configuring Switch-Based Authentication chapter in the *Catalyst 3750 Switch Software Configuration Guide, 12.2(50)SE*.

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 68: 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 69: 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

Value	Explanation
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

CoA Request Commands, on page 548

Session Identification

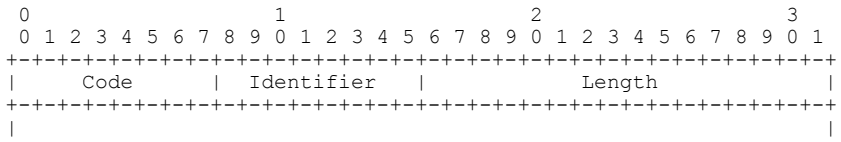
For disconnect and CoA requests targeted at a particular session, the switch locates the session based on one or more of the following attributes:

- 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.



```
|                          Authenticator                          |
|                                                                 |
|                                                                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Attributes ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-
```

The attributes field is used to carry Cisco vendor-specific attributes (VSAs).

Related Topics

CoA Disconnect-Request, on page 549
CoA Request: Disable Host Port, on page 550
CoA Request: Bounce-Port, on page 550

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 70: CoA Commands Supported on the Switch

Command [4]	Cisco VSA
Reauthenticate host	Cisco:Avpair="subscriber:command=reauthenticate"
Terminate session	This is a standard disconnect request that does not require a VSA.
Bounce host port	Cisco:Avpair="subscriber:command=bounce-host-port"
Disable host port	Cisco:Avpair="subscriber:command=disable-host-port"

4 All CoA commands must include the session identifier between the switch and the CoA client.

Related Topics

CoA Request Response Code, on page 547

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 member.
- If the stack master fails before authentication completes, reauthentication is initiated after stack master switch-over based on the original command (which is subsequently removed).
- If the stack master fails before sending an ACK, the new stack master 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. 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 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 547

CoA Request: Disable Host Port

This command is carried in a standard CoA-Request message that has this new VSA:

Cisco:Avpair="subscriber:command=disable-host-port"

Because this command is session-oriented, it must be accompanied by one or more of the session identification attributes. 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 547

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

Session Identification, on page 547

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 stack master 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 stack master.

If the stack master fails before the port-bounce completes, a port-bounce is initiated after stack master change-over based on the original command (which is subsequently removed).

If the stack master fails before sending a CoA-ACK message, the new stack master 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 stack master 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 stack master.

If the stack master fails before the port-disable operation completes, the port is disabled after stack master change-over based on the original command (which is subsequently removed).

If the stack master fails before sending a CoA-ACK message, the new stack master 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

Identifying the RADIUS Server Host, on page 554
Defining AAA Server Groups, on page 558
Configuring Settings for All RADIUS Servers, on page 563
Configuring RADIUS Login Authentication, on page 556

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 556

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 558

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 560

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

Starting RADIUS Accounting, on page 562

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.

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)."

For a complete list of RADIUS attributes or more information about vendor-specific attribute 26, see the "RADIUS Attributes" appendix in the *Cisco IOS Security Configuration Guide*.

Related Topics

Configuring the Switch to Use Vendor-Specific RADIUS Attributes, on page 564

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

Configuring the Switch for Vendor-Proprietary RADIUS Server Communication, on page 565

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.

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. **configure terminal**
2. **radius-server host** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**radius-server host** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*] **Example:** `Controller(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. **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.

	Command or Action	Purpose
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Related Topics

RADIUS Server Host, on page 551

Defining AAA Server Groups, on page 558

Configuring Settings for All RADIUS Servers, on page 563

Configuring RADIUS Login Authentication

Beginning in privileged EXEC mode, 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.

For more information about the **ip http authentication** command, see the *Cisco IOS Security Command Reference, Release 12.4*.

SUMMARY STEPS

1. **configure terminal**
2. **aaa new-model**
3. **aaa authentication login** {**default** | *list-name*} *method1* [*method2...*]
4. **line** [**console** | **tty** | **vty**] *line-number* [*ending-line-number*]
5. **login authentication** {**default** | *list-name*}
6. **end**

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	**aaa new-model** **Example:** `Controller(config)# aaa new-model`	Enables AAA.
Step 3	**aaa authentication login** {**default** | *list-name*} *method1* [*method2...*] **Example:** `Controller(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.
Step 4	**line** [**console** | **tty** | **vty**] *line-number* [*ending-line-number*] **Example:** `Controller(config)# line 1 4`	Enters line configuration mode, and configure the lines to which you want to apply the authentication list.
Step 5	**login authentication** {**default** | *list-name*}	Applies the authentication list to a line or set of lines.

	Command or Action	Purpose
	Example: `Controller(config)# login authentication default`	• 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 6	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Related Topics

RADIUS Login Authentication, on page 552

RADIUS Server Host, on page 551

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.

Beginning in privileged EXEC mode, follow these steps to define AAA server groups:

SUMMARY STEPS

1. **configure terminal**
2. **radius-server host** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]
3. **aaa new-model**
4. **aaa group server radius** *group-name*
5. **server** *ip-address*
6. **end**

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	**radius-server host** {*hostname* | *ip-address*} [**auth-port** *port-number*] [**acct-port** *port-number*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*] **Example:** `Controller(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. **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 3	**aaa new-model** **Example:** `Controller(config)# aaa new-model`	Enables AAA.
Step 4	**aaa group server radius** *group-name* **Example:** `Controller(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.

	Command or Action	Purpose
Step 5	**server** *ip-address* **Example:** `Controller(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 6	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

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.

```
Controller(config)# radius-server host 172.20.0.1 auth-port 1000 acct-port 1001
Controller(config)# radius-server host 172.10.0.1 auth-port 1645 acct-port 1646
Controller(config)# aaa new-model
Controller(config)# aaa group server radius group1
Controller(config-sg-radius)# server 172.20.0.1 auth-port 1000 acct-port 1001
Controller(config-sg-radius)# exit
Controller(config)# aaa group server radius group2
Controller(config-sg-radius)# server 172.20.0.1 auth-port 2000 acct-port 2001
Controller(config-sg-radius)# exit
```

Related Topics

Identifying the RADIUS Server Host, on page 554
RADIUS Server Host, on page 551
AAA Server Groups, on page 553

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.

Beginning in privileged EXEC mode, follow these steps to configure RADIUS authorization for user priviledged access and network services:

SUMMARY STEPS

1. **configure terminal**
2. **aaa authorization network radius**
3. **aaa authorization exec radius**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**aaa authorization network radius** **Example:** `Controller(config)# aaa authorization network radius`	Configures the switch for user RADIUS authorization for all network-related service requests.
Step 3	**aaa authorization exec radius** **Example:** `Controller(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 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

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 553

Starting RADIUS Accounting

Beginning in privileged EXEC mode, follow these steps to start RADIUS accounting:

SUMMARY STEPS

1. **configure terminal**
2. **aaa accounting network start-stop radius**
3. **aaa accounting exec start-stop radius**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**aaa accounting network start-stop radius** **Example:** `Controller(config)# aaa accounting network start-stop radius`	Enables RADIUS accounting for all network-related service requests.
Step 3	**aaa accounting exec start-stop radius** **Example:** `Controller(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 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

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 553

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**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**radius-server key** *string* **Example:** `Controller(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.

	Command or Action	Purpose
Step 3	**radius-server retransmit** *retries* **Example:** `Controller(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:** `Controller(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:** `Controller(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:** `Controller(config)# end`	Returns to privileged EXEC mode.

Related Topics

Identifying the RADIUS Server Host, on page 554
RADIUS Server Host, on page 551

Configuring the Switch to Use Vendor-Specific RADIUS Attributes

Beginning in privileged EXEC mode, follow these steps to configure the switch to use vendor-specific RADIUS attributes:

SUMMARY STEPS

1. **configure terminal**
2. **radius-server vsa send** [**accounting** | **authentication**]
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**radius-server vsa send** [**accounting** | **authentication**] **Example:** `Controller(config)# radius-server vsa send`	Enables the switch to recognize and use VSAs as defined by RADIUS IETF attribute 26. • (Optional) Use the **accounting** keyword to limit the set of recognized vendor-specific attributes to only accounting attributes. • (Optional) Use the **authentication** keyword to limit the set of recognized vendor-specific attributes to only authentication attributes. If you enter this command without keywords, both accounting and authentication vendor-specific attributes are used.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Related Topics

Vendor-Specific RADIUS Attributes, on page 553

Configuring the Switch for Vendor-Proprietary RADIUS Server Communication

Beginning in privileged EXEC mode, follow these steps to configure the switch to use vendor-proprietary RADIUS server communication:

SUMMARY STEPS

1. **configure terminal**
2. **radius-server host** {*hostname* | *ip-address*} **non-standard**
3. **radius-server key** *string*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**radius-server host** {*hostname* | *ip-address*} **non-standard** **Example:** `Controller(config)# radius-server host 172.20.30.15 nonstandard`	Specifies the IP address or hostname of the remote RADIUS server host and identifies that it is using a vendor-proprietary implementation of RADIUS.
Step 3	**radius-server key** *string* **Example:** `Controller(config)# radius-server key rad124`	Specifies the shared secret text string used between the switch and the vendor-proprietary RADIUS server. The switch and the RADIUS server use this text string to encrypt passwords and exchange responses. **Note** The key is a text string that must match the encryption key used on the RADIUS server. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

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

Vendor-Proprietary RADIUS Server Communication, on page 554

Configuring CoA on the Switch

Beginning in privileged EXEC mode, follow these steps to configure CoA on a switch. This procedure is required.

SUMMARY STEPS

1. **configure terminal**
2. **aaa new-model**
3. **aaa server radius dynamic-author**
4. **client** {*ip-address* | *name*} [**vrf** *vrfname*] [**server-key** *string*]
5. **server-key** [**0** | **7**] *string*
6. **port** *port-number*
7. **auth-type** {**any** | **all** | **session-key**}
8. **ignore session-key**
9. **ignore server-key**
10. **authentication command bounce-port ignore**
11. **authentication command disable-port ignore**
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**aaa new-model** **Example:** `Controller(config)# aaa new-model`	Enables AAA.
Step 3	**aaa server radius dynamic-author** **Example:** `Controller(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 4	**client** {*ip-address* | *name*} [**vrf** *vrfname*] [**server-key** *string*]	Enters dynamic authorization local server configuration mode and specifies a RADIUS client from which a device will accept CoA and disconnect requests.
Step 5	**server-key** [**0** | **7**] *string* **Example:** `Controller(config-sg-radius)# server-key your_server_key`	Configures the RADIUS key to be shared between a device and RADIUS clients.

	Command or Action	Purpose
Step 6	**port** *port-number* **Example:** `Controller(config-sg-radius)# port 25`	Specifies the port on which a device listens for RADIUS requests from configured RADIUS clients.
Step 7	**auth-type** {**any** | **all** | **session-key**} **Example:** `Controller(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 8	**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 9	**ignore server-key** **Example:** `Controller(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 10	**authentication command bounce-port ignore** **Example:** `Controller(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.
Step 11	**authentication command disable-port ignore** **Example:** `Controller(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 12	**end** **Example:** `Controller(config-sg-radius)# end`	Returns to privileged EXEC mode.

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 71: Privileged EXEC show Commands

Command	Purpose
show aaa attributes protocol radius	Displays AAA attributes of RADIUS commands.

Table 72: Global Troubleshooting Commands

Command	Purpose
debug radius	Displays information for troubleshooting RADIUS.
debug aaa coa	Displays information for troubleshooting CoA processing.
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:

```
Controller(config)# radius-server host 172.29.36.49 auth-port 1612 key rad1
Controller(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:

```
Controller(config)# radius-server host host1
```

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:

```
Controller(config)# radius-server host 172.20.30.15 nonstandard
Controller(config)# radius-server key rad124
```

Additional References

Related Documents

Related Topic	Document Title

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

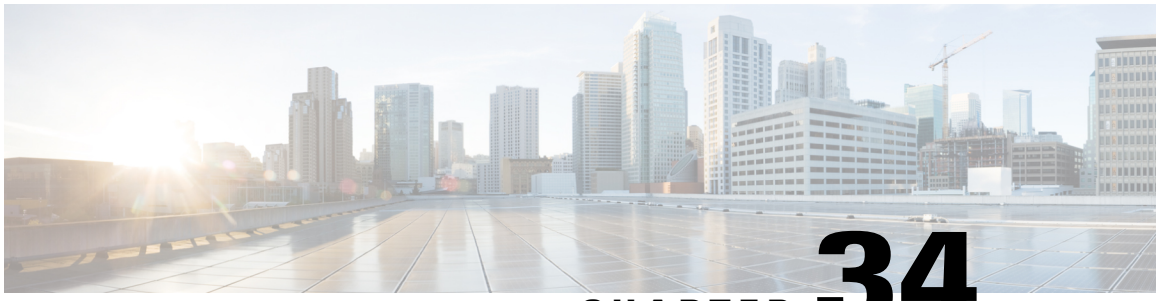
Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, 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 for RADIUS

Table 73: Feature Information for RADIUS

Feature Name	Releases	Feature Information

-



CHAPTER 34

Configuring Kerberos

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 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.

For Kerberos configuration examples, see the "Kerberos Configuration Examples" section in the "Security Server Protocols" chapter of the *Cisco IOS Security Configuration Guide, Release 12.4.*.

For complete syntax and usage information for the commands used in this section, see the "Kerberos Commands" section in the "Security Server Protocols" chapter of the *Cisco IOS Security Command Reference, Release 12.4*.

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).

In this software release, Kerberos supports these network services:

- Telnet
- rlogin
- rsh

This table lists the common Kerberos-related terms and definitions.

Table 74: 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[5] 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.

Term	Definition
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[6]	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[7]	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[8].

Term	Definition
Principal	Also known as a Kerberos identity, this is who you are or what a service is according to the Kerberos server. **Note** The Kerberos principal name *must* be in all lowercase characters.
Service credential	A credential for a network service. When issued from the KDC, this credential is encrypted with the password shared by the network service and the KDC. The password is also shared with the user TGT.
SRVTAB	A password that a network service shares with the KDC. In Kerberos 5 or later Kerberos versions, SRVTAB is referred to as KEYTAB.
TGT	Ticket granting ticket that is a credential that the KDC issues to authenticated users. When users receive a TGT, they can authenticate to network services within the Kerberos realm represented by the KDC.

5 ticket granting ticket
6 key distribution center
7 key table
8 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 577
2 Obtaining a TGT from a KDC, on page 578
3 Authenticating to Network Services, on page 578

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

Standards and RFCs

Standard/RFC	Title

MIBs

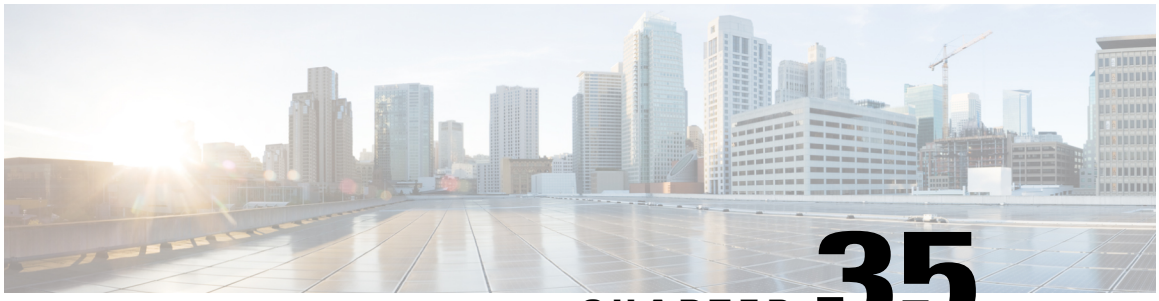
MIB	MIBs Link
	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

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 for Kerberos

Feature Name	Releases	Feature Information



CHAPTER 35

Configuring Local Authentication and Authorization

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Local Authentication and Authorization

This section lists any prerequisites for local authentication and authorization.

Restrictions on Local Authentication and Authorization

This section lists any restrictions for 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 Catalyst 3850 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.

Beginning in privileged EXEC mode, follow these steps to configure AAA to operate without a server by setting the switch to implement AAA in local mode:

SUMMARY STEPS

1. **configure terminal**
2. **aaa new-model**
3. **aaa authentication login default local**
4. **aaa authorization exec local**
5. **aaa authorization network local**
6. **username** *name* [**privilege** *level*] {**password** *encryption-type password*}
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**aaa new-model** **Example:** `Controller(config)# aaa new-model`	Enables AAA.
Step 3	**aaa authentication login default local** **Example:** `Controller(config)# aaa authentication`	Sets the login authentication to use the local username database. The **default** keyword applies the local user database authentication to all ports.

	Command or Action	Purpose
	`login default local`	
Step 4	**aaa authorization exec local** **Example:** `Controller(config)# aaa authorization exec local`	Configures user AAA authorization, check the local database, and allow the user to run an EXEC shell.
Step 5	**aaa authorization network local** **Example:** `Controller(config)# aaa authorization network local`	Configures user AAA authorization for all network-related service requests.
Step 6	**username** *name* [**privilege** *level*] {**password** *encryption-type password*} **Example:** `Controller(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. 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 7	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Related Topics

Setting Up the Switch to Run SSH, on page 591
SSH Configuration Guidelines, on page 589

Monitoring Local Authentication and Authorization

To display Local Authentication and Authorization configuration, use the **show running-config** privileged EXEC command.

Additional References

Related Documents

Related Topic	Document Title

Standards and RFCs

Standard/RFC	Title

MIBs

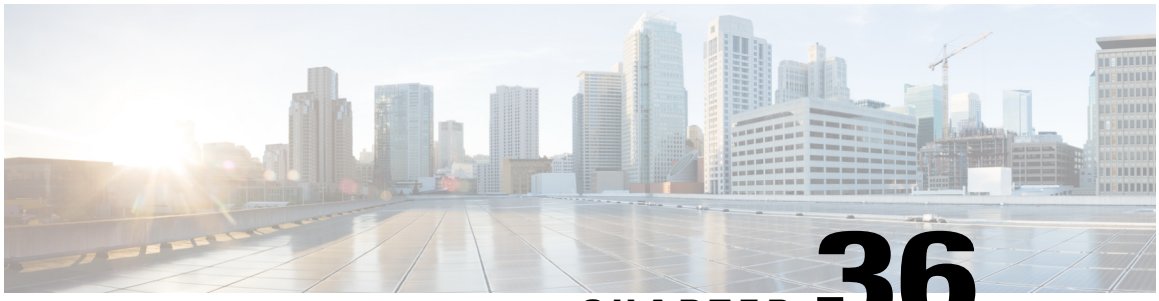
MIB	MIBs Link
	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

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 for Local Authentication and Authorization

Feature Name	Releases	Feature Information



CHAPTER 36

Configuring Secure Shell (SSH)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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

Secure Copy Protocol Concepts, on page 590

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

Secure Copy Protocol Concepts, on page 590

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

For SSH configuration examples, see the “SSH Configuration Examples” section in the “Configuring Secure Shell” section in the “Other Security Features” chapter of the *Cisco IOS Security Configuration Guide, Cisco IOS Release 12.4*.

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.

Note For complete syntax and usage information for the commands used in this section, see the command reference for this release and the "Secure Shell Commands" section of the "Other Security Features" chapter of the *Cisco IOS Security Command Reference, Release 12.4* and the *Cisco IOS IPv6 Command Reference*.

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

Configuring the Switch for Local Authentication and Authorization, on page 582
TACACS+ and Switch Access, on page 527
RADIUS and Switch Access, on page 543

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 a stack master and the stack master fails, the new stack master uses the RSA key pair generated by the previous stack master.
- If you get CLI error messages after entering the **crypto key generate rsa** global configuration command, an RSA key pair has not been generated. Reconfigure the hostname and domain, and then enter the **crypto key generate rsa** command. For more information, see Related Topics below.
- 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

Setting Up the Switch to Run SSH, on page 591
Configuring the Switch for Local Authentication and Authorization, on page 582

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 Concepts

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.

To configure the Secure Copy feature, you should understand the SCP concepts.

The behavior of SCP is similar to that of remote copy (rcp), which comes from the Berkeley r-tools suite, except that SCP relies on SSH for security. SCP also requires that authentication, authorization, and accounting (AAA) authorization be configured so the router can determine whether the user has the correct privilege level.

For information about how to configure and verify SCP, see the “Secure Copy Protocol” section in the *Cisco IOS Security Configuration Guide: Securing User Services, Release 12.4*.

Related Topics

Prerequisites for Configuring the Switch for Secure Shell (SSH) and Secure Copy Protocol (SCP), on page 587
Restrictions for Configuring the Switch for SSH, on page 588

How to Configure SSH

Setting Up the Switch to Run SSH

Beginning in privileged EXEC mode, 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. **configure terminal**
2. **hostname** *hostname*
3. **ip domain-name** *domain_name*
4. **crypto key generate rsa**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**hostname** *hostname* **Example:** `Controller(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 3	**ip domain-name** *domain_name* **Example:** `Controller(config)# ip domain-name your_domain`	Configures a host domain for your switch.
Step 4	**crypto key generate rsa** **Example:** `Controller(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.

	Command or Action	Purpose
		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 5	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Related Topics

SSH Configuration Guidelines, on page 589
Configuring the Switch for Local Authentication and Authorization, on page 582

Configuring the SSH Server

Beginning in privileged EXEC mode, 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. **configure terminal**
2. **ip ssh version** [**1** | **2**]
3. **ip ssh** {**timeout** *seconds* | **authentication-retries** *number*}
4. Use one or both of the following:
   - **line vty***line_number*[*ending_line_number*]
   - **transport input ssh**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ip ssh version** [**1** | **2**] **Example:** `Controller(config)# ip ssh version 1`	(Optional) Configures the switch to run SSH Version 1 or SSH Version 2. • **1—**Configure the switch to run 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 3	**ip ssh** {**timeout** *seconds* | **authentication-retries** *number*} **Example:** `Controller(config)# ip ssh timeout 90 authentication-retries 2`	Configures the SSH control parameters: • 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. 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. • 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 4	Use one or both of the following:
	• **line vty***line_number*[*ending_line_number*]
	• **transport input ssh**
	
	Example:
	`Controller(config)# line vty 1 10`
	
	or
	
	`Controller(config-line)# transport input ssh`
		(Optional) Configures the virtual terminal line settings.
		• 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.
		• Specifies that the switch prevent non-SSH Telnet connections. This limits the router to only SSH connections.

	Command or Action	Purpose
Step 5	**end** **Example:** `Controller(config-line)# end`	Returns to privileged EXEC mode.

Monitoring the SSH Configuration and Status

This table displays the SSH server configuration and status.

Table 75: 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.

For more information about these commands, see the "Secure Shell Commands" section in the "Other Security Features" chapter of the *Cisco IOS Security Command Reference* .

Additional References

Related Documents

Related Topic	Document Title

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

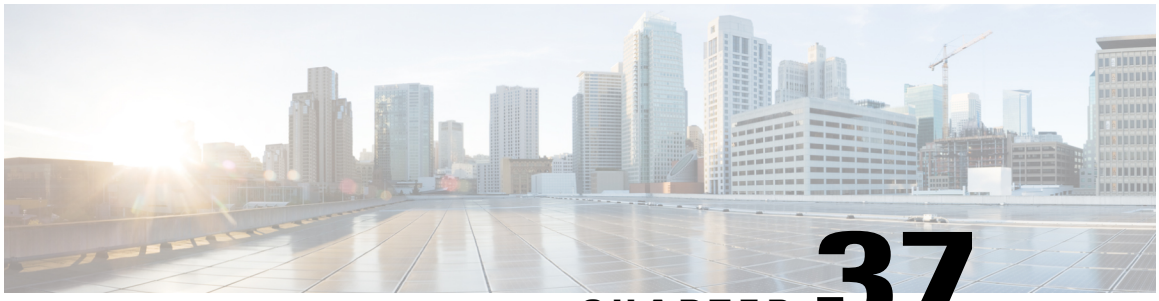
Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, 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 for SSH

Table 76: Feature Information for SSH

Feature Name	Releases	Feature Information

-



CHAPTER 37

Configuring Secure Socket Layer HTTP

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Sockets Layer HTTP

This section lists the prerequisites for configuring the Catalyst 3850 switch for secure socket layer (SSL) HTTP.

Restrictions for Configuring the Switch for Secure Sockets Layer HTTP

This section lists the restrictions for configuring the Catalyst 3850 switch for secure socket layer (SSL) HTTP.

Information about Secure Sockets Layer (SSL) HTTP

This section describes how to configure Secure Sockets Layer (SSL) Version 3.0 support for the HTTP 1.1 server and client. SSL provides server authentication, encryption, and message integrity, as well as HTTP client authentication, to allow secure HTTP communications.

Note SSL evolved into Transport Layer Security (TLS) in 1999, but is still used in this particular context.

On a secure HTTP connection, data to and from an HTTP server is encrypted before being sent over the Internet. HTTP with SSL encryption provides a secure connection to allow such functions as configuring a switch from a Web browser. Cisco's implementation of the secure HTTP server and secure HTTP client uses an implementation of SSL Version 3.0 with application-layer encryption. HTTP over SSL is abbreviated as HTTPS; the URL of a secure connection begins with https:// instead of http://.

The primary role of the HTTP secure server (the switch) is to listen for HTTPS requests on a designated port (the default HTTPS port is 443) and pass the request to the HTTP 1.1 Web server. The HTTP 1.1 server processes requests and passes responses (pages) back to the HTTP secure server, which, 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.

For configuration examples and complete syntax and usage information for the commands used in this section, see the “HTTPS - HTTP Server and Client with SSL 3.0” feature description for Cisco IOS Release 12.2(15)T.

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.

```
Controller# 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_RC4_128_MD5—RSA key exchange with RC4 128-bit encryption and MD5 for message digest
3. SSL_RSA_WITH_RC4_128_SHA—RSA key exchange with RC4 128-bit encryption and SHA for message digest
4. SSL_RSA_WITH_3DES_EDE_CBC_SHA—RSA key exchange with 3DES and DES-EDE3-CBC for message encryption and SHA for message digest

RSA (in conjunction with the specified encryption and digest algorithm combinations) is used for both key generation and authentication on SSL connections. This usage is independent of whether or not a CA trustpoint is configured.

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 stack master.

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://.

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.

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:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**hostname** *hostname* **Example:** `Controller(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:** `Controller(config)# ip domain-name`	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.

	Command or Action	Purpose
	`your_domain`	
Step 4	**crypto key generate rsa** **Example:** `Controller(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:** `Controller(config)# crypto ca trustpoint your_trustpoint`	Specifies a local configuration name for the CA trustpoint and enter CA trustpoint configuration mode.
Step 6	**enrollment url** *url* **Example:** `Controller(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:** `Controller(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:** `Controller(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:** `Controller(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:** `Controller(ca-trustpoint)# exit`	Exits CA trustpoint configuration mode and return to global configuration mode.

	Command or Action	Purpose
Step 11	**crypto ca authentication** *name* **Example:** `Controller(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:** `Controller(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:** `Controller(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:
	
	`Controller# show ip http server status`
		(Optional) Displays the status of the HTTP server to determine if the secure HTTP server feature is supported in the software. You should see one of these lines in the output:
		
		`HTTP secure server capability: Present`
		
		or
		
		`HTTP secure server capability: Not present`

Step 2	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 3	**ip http secure-server** **Example:** `Controller(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:** `Controller(config)# ip http secure-port`	(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.

	Command or Action	Purpose
	`443`	
Step 5	**ip http secure-ciphersuite** {[**3des-ede-cbc-sha**] [**rc4-128-md5**] [**rc4-128-sha**] [**des-cbc-sha**]} **Example:** `Controller(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:** `Controller(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:** `Controller(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:** `Controller(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).
Step 9	**ip http access-class** *access-list-number* **Example:** `Controller(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:** `Controller(config)# ip http max-connections 4`	(Optional) Sets the maximum number of concurrent connections that are allowed to the HTTP server. The range is 1 to 16; the default value is 5.
Step 11	**ip http timeout-policy idle** *seconds* **life** *seconds* **requests** *value*	(Optional) Specifies how long a connection to the HTTP server can remain open under the defined circumstances:

	Command or Action	Purpose
	Example: `Controller(config)# ip http timeout-policy idle 120 life 240 requests 1`	• **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:** `Controller(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:** `Controller# configure terminal`	Enters the global configuration mode.

	Command or Action	Purpose
Step 2	**ip http client secure-trustpoint** *name* **Example:** `Controller(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:** `Controller(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:** `Controller(config)# end`	Returns to privileged EXEC mode.

How to Configure Secure HTTP Servers and Clients

These sections contain this configuration information:

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 77: 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

Related Documents

Related Topic	Document Title

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

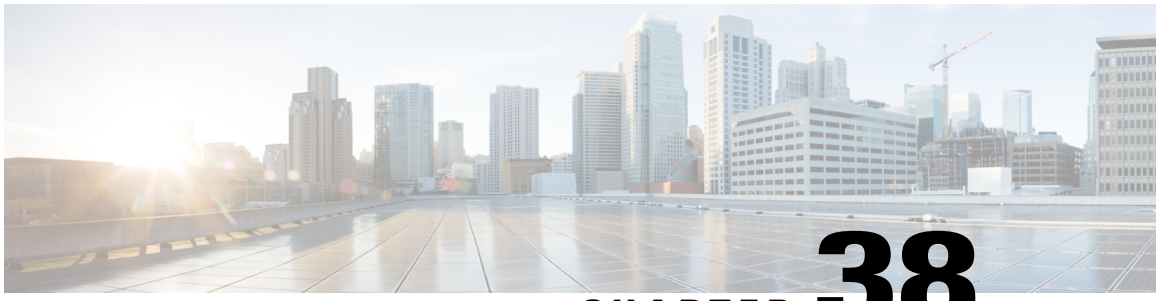
Technical Assistance

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 for SSL HTTP

Table 78: Feature Information for SSL HTTP

Feature Name	Releases	Feature Information



CHAPTER 38

Configuring IPv4 ACLs

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Network Security with ACLs

This section lists the prerequisites for configuring network security with Access Control Lists (ACLs).

- On switches running the LAN base feature set, VLAN maps are not supported.

Restrictions for Configuring Network Security with ACLs

General Network Security

The following are restrictions for configuring network security with ACLs:

- You cannot apply named MAC extended ACLs to Layer 3 interfaces.
- 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.

ACL Filtering

The following are restrictions on ACL filtering:

- If IEEE 802.1Q tunneling is configured on an interface, any IEEE 802.1Q encapsulated IP packets received on the tunnel port can be filtered by MAC ACLs, but not by IP ACLs. This is because the switch does not recognize the protocol inside the IEEE 802.1Q header. This restriction applies to router ACLs, port ACLs, and VLAN maps.

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.
- When private VLANs are configured, you can apply router ACLs only on the primary-VLAN SVIs. The ACL is applied to both primary and secondary VLAN Layer 3 traffic.

Note By default, the router sends Internet Control Message Protocol (ICMP) unreachable messages when a packet is denied by an access group 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:

- 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. Incoming packets received on the Layer 2 port are always filtered by the port ACL.
- You can apply no more than one IP access list and one MAC access list to the same Layer 2 interface. The IP access list filters only IP packets, and the MAC access list filters non-IP packets.
- A Layer 2 interface can have only one MAC access list. If you apply a MAC access list to a Layer 2 interface that has a MAC ACL configured, the new ACL replaces the previously configured one.

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.

Related Topics

Applying an IPv4 ACL to an Interface, on page 637
IPv4 ACL Interface Considerations, on page 626
Creating Named MAC Extended ACLs, on page 638
Applying a MAC ACL to a Layer 2 Interface, on page 640

Information about Network Security with ACLs

This chapter describes how to configure network security on the Catalyst 3850 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 Port ACLs, router ACLs, and VLAN maps 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.

Related Topics

Restrictions for Configuring Network Security with ACLs, on page 612

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 on outbound and inbound interfaces. The following access lists are supported:

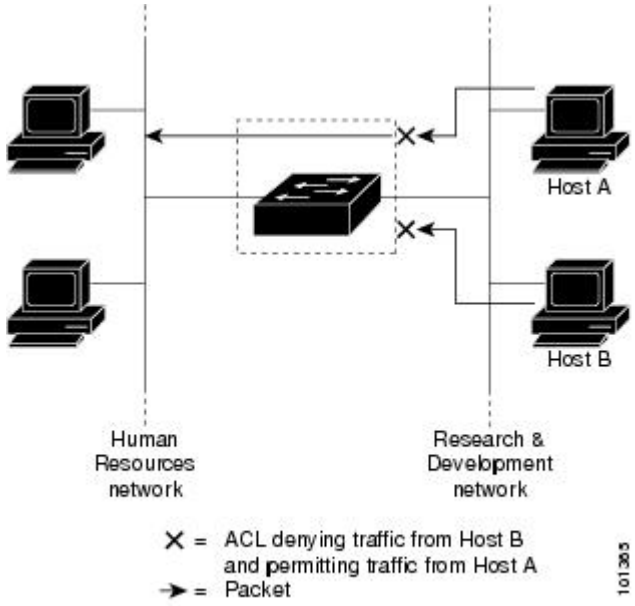
- Standard IP access lists using source addresses
- Extended IP access lists using source and destination addresses and optional protocol type information
- MAC extended access lists using source and destination MAC addresses and optional protocol type information

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.

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.

Figure 14: Using ACLs to Control Traffic in a Network



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).

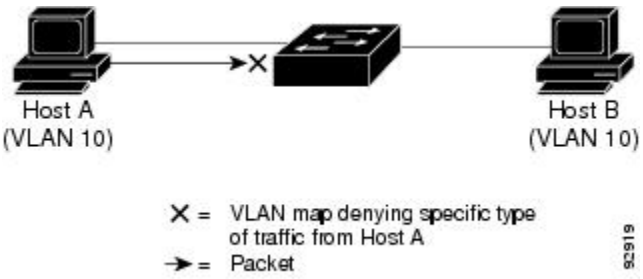
You can configure VLAN maps to match Layer 3 addresses for IPv4 traffic.

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.

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.

Figure 15: Using VLAN Maps to Control Traffic



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:

```
Controller(config)# access-list 102 permit tcp any host 10.1.1.1 eq smtp
Controller(config)# access-list 102 deny tcp any host 10.1.1.2 eq telnet
Controller(config)# access-list 102 permit tcp any host 10.1.1.2
Controller(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.
- ACL logging for port ACLs and VLAN maps

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 79: Access List Numbers

Access List Number	Type	Supported
1–99	IP standard access list	Yes
100–199	IP extended access list	Yes
200–299	Protocol type-code access list	No
300–399	DECnet access list	No
400–499	XNS standard access list	No
500–599	XNS extended access list	No
600–699	AppleTalk access list	No
700–799	48-bit MAC address access list	No
800–899	IPX standard access list	No
900–999	IPX extended access list	No
1000–1099	IPX SAP access list	No
1100–1199	Extended 48-bit MAC address access list	No
1200–1299	IPX summary address access list	No
1300–1999	IP standard access list (expanded range)	Yes
2000–2699	IP extended access list (expanded range)	Yes

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 terminal lines, to interfaces, or to VLANs.

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. The advantage of using named ACLs instead of numbered lists is that you can delete individual entries from a named list.

Consider these guidelines and limitations before configuring named ACLs:

- Not all commands that accept a numbered ACL accept a named ACL. ACLs for packet filters and route filters on interfaces can use a name. VLAN maps also accept a name.
- A standard ACL and an extended ACL cannot have the same name.
- Numbered ACLs are also available.
- You can use standard or extended ACLs (named or numbered) in VLAN maps.
- With IPv4 QoS ACLs, if you enter the **class-map** {**match-all** | **match-any**} *class-map-name* global configuration command, you can enter these **match** commands:
  - **match access-group** *acl-name*

    Note The ACL must be an extended named ACL.

  - **match input-interface** *interface-id-list*
  - **match ip dscp** *dscp-list*
  - **match ip precedence** *ip-precedence-list*

    You cannot enter the **match access-group** *acl-index* command.

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.

For router ACLs, other factors can cause packets to be sent to the CPU:

- Using the **log** keyword
- Generating ICMP unreachable messages

When traffic flows are both logged and forwarded, forwarding is done by hardware, but logging must be done by software. Because of the difference in packet handling capacity between hardware and software, if the sum of all flows being logged (both permitted flows and denied flows) is of great enough bandwidth, not all of the packets that are forwarded can be logged.

When you enter the **show ip access-lists** privileged EXEC command, the match count displayed does not account for packets that are access controlled in hardware. Use the **show platform acl counters hardware** privileged EXEC command to obtain some basic hardware ACL statistics for switched and routed packets.

Router ACLs function as follows:

- The hardware controls permit and deny actions of standard and extended ACLs (input and output) for security access control.
- If **log** has not been specified, the flows that match a *deny* statement in a security ACL are dropped by the hardware if *ip unreachables* is disabled. The flows matching a *permit* statement are switched in hardware.
- Adding the **log** keyword to an ACE in a router ACL causes a copy of the packet to be sent to the CPU for logging only. If the ACE is a *permit* statement, the packet is still switched and routed in hardware.

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.
- 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.

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.

Related Topics

Configuring Time Ranges for ACLs, on page 634

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.

Related Topics

Applying an IPv4 ACL to an Interface, on page 637
Restrictions for Configuring Network Security with ACLs, on page 612

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

Beginning in privileged EXEC mode, follow these steps to create a numbered standard ACL:

SUMMARY STEPS

1. **configure terminal**
2. **access-list** *access-list-number* {**deny** | **permit**} *source source-wildcard* [**log**]
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**access-list** *access-list-number* {**deny** | **permit**} *source source-wildcard* [**log**] **Example:** `Controller(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. (Optional) Enter **log** to cause an informational logging message about the packet that matches the entry to be sent to the console.

	Command or Action	Purpose
		(Optional) Enter **smartlog** to send copies of denied or permitted packets to a NetFlow collector. **Note** Logging is supported only on ACLs attached to Layer 3 interfaces.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Related Topics

Configuring VLAN Maps, on page 641

Creating a Numbered Extended ACL

Beginning in privileged EXEC mode, 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:** `Controller# configure terminal`	Enters the global configuration mode.
Step 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*] **Example:** `Controller(config)# access-list 101 permit ip host 10.1.1.2 any precedence 0 tos 0 log`	Defines an extended IPv4 access list and the access conditions. The *access-list-number* is a decimal number from 100 to 199 or 2000 to 2699. Enter **deny** or **permit** to specify whether to deny or permit the packet if conditions are matched. For *protocol*, enter the name or number of an 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**. 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 the entry or **log-input** to include the input interface in the log entry. • **smartlog—**Enter when smart logging is globally enabled to have a copy of the denied or permitted packet sent to a NetFlow collector.

	Command or Action	Purpose
		• **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:** `Controller(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. 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:** `Controller(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.
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: • *icmp-type***—**Enter to filter by ICMP message type, a number from 0 to 255.

	Command or Action	Purpose
	Example: `Controller(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*] **Example:** `Controller(config)# access-list 101 permit igmp any any 14`	(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. *igmp-type*—To match IGMP message type, enter a number from 0 to 15, or enter the message name: **dvmrp**, **host-query**, **host-report**, **pim**, or **trace**.
Step 7	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Extended IP ACL with the any Keyword

To use an abbreviation for a source and source wildcard of 0.0.0.0 255.255.255.255 and an abbreviation for a destination and destination wildcard of 0.0.0.0 255.255.255.255 when defining an extended IP ACL, use the **any** keyword in place of source and destination address and wildcard:

```
Controller# configure terminal
Controller(config)# access-list 101 permit ip any any precedence 0 tos 0 fragments
log time-range workhours dscp 10
Controller(config)# end
```

Extended IP ACL with the host Keyword

To use an abbreviation for a source and a source wildcard of source 0.0.0.0 and an abbreviation for a destination and destination wildcard of destination 0.0.0.0 when defining an extended IP ACL, use the **host** keyword in place of the source and destination wildcard or mask.

```
Controller# configure terminal
Controller(config)# access-list 101 permit ip host 10.1.1.2 any
Controller(config)# end
```

Related Topics

Configuring VLAN Maps, on page 641

Creating Named Standard ACLs

Beginning in privileged EXEC mode, follow these steps to create a standard ACL using names:

SUMMARY STEPS

1. **configure terminal**
2. **ip access-list standard** *name*
3. Use one of the following:
   - **deny** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**]
   - **permit** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**]]
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ip access-list standard** *name* **Example:** `Controller(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 3	Use one of the following:
	• **deny** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**]
	• **permit** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**]]
	
	Example:
	`Controller(config-std-nacl)# deny 192.168.0.0 0.0.255.255 255.255.0.0 0.0.255.255`
	or
	`Controller(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.

	Command or Action	Purpose
Step 4	**end** **Example:** `Controller(config-std-nacl)# end`	Returns to privileged EXEC mode.

Creating Extended Named ACLs

Beginning in privileged EXEC mode, follow these steps to create an extended ACL using names:

SUMMARY STEPS

1. **configure terminal**
2. **ip access-list extended** *name*
3. {**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*]
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ip access-list extended** *name* **Example:** `Controller(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.
Step 3	{**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:** `Controller(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.

	Command or Action	Purpose
		• **any—**A source and source wildcard or destination and destination wildcard of 0.0.0.0 255.255.255.255.
Step 4	**end** **Example:** `Controller(config-ext-nacl)# end`	Returns to privileged EXEC mode.

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. This example shows how you can delete individual ACEs from the named access list *border-list*:

```
Controller(config)# ip access-list extended border-list
Controller(config-ext-nacl)# no permit ip host 10.1.1.3 any
```

Being able to selectively remove lines from a named ACL is one reason you might use named ACLs instead of numbered ACLs.

What to Do Next

After creating a named ACL, you can apply it to interfaces or to VLANs .

Configuring Time Ranges for ACLs

Beginning in privileged EXEC mode, follow these steps to configure a time-range parameter for an ACL:

SUMMARY STEPS

1. **configure terminal**
2. **time-range** *time-range-name*
3. 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*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**time-range** *time-range-name* **Example:** `Controller(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 3	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:
	
	`Controller(config-time-range)# absolute start 00:00 1 Jan 2006 end 23:59 1 Jan 2006`
	
	or
	
	`Controller(config-time-range)# periodic weekdays 8:00 to 12:00`
		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.

Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

What to Do Next

Repeat the steps if you have multiple items that you want in effect at different times.

Related Topics

Time Ranges for ACLs, on page 625

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.

Beginning in privileged EXEC mode, follow these steps to restrict incoming and outgoing connections between a virtual terminal line and the addresses in an ACL:

SUMMARY STEPS

1. **configure terminal**
2. **line** [**console** | **vty**] *line-number*
3. **access-class** *access-list-number* {**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:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**line** [**console**	**vty**] *line-number* **Example:** `Controller(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 3	**access-class** *access-list-number* {**in**	**out**} **Example:** `Controller(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.

	Command or Action	Purpose
Step 4	**end** **Example:** `Controller(config-line)# end`	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** `Controller# show running-config`	Displays the access list configuration.
Step 6	**copy running-config startup-config** **Example:** `Controller# 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:** `Controller# configure terminal`	Enters the global configuration mode.

	Command or Action	Purpose
Step 2	**interface** *interface-id* **Example:** `Controller(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:** `Controller(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:** `Controller(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** `Controller# show running-config`	Displays the access list configuration.
Step 6	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

IPv4 ACL Interface Considerations, on page 626

Restrictions for Configuring Network Security with ACLs, on page 612

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.

Beginning in privileged EXEC mode, follow these steps to create a named MAC extended ACL:

SUMMARY STEPS

1. **configure terminal**
2. **mac access-list extended** *name*
3. {**deny** | **permit**} {**any** | **host** *source MAC address* | *source MAC address mask*} {**any** | **host** *destination MAC address* | *destination MAC address mask*} [*type mask* | **lsap** *lsap mask* | **aarp** | **amber** | **dec-spanning** | **decnet-iv** | **diagnostic** | **dsm** | **etype-6000** | **etype-8042** | **lat** | **lavc-sca** | **mop-console** | **mop-dump** | **msdos** | **mumps** | **netbios** | **vines-echo** | **vines-ip** | **xns-idp** | 0-*65535*] [**cos** *cos*]
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**mac access-list extended** *name* **Example:** `Controller(config)# mac access-list extended macl`	Defines an extended MAC access list using a name.

Step 3	{**deny** | **permit**} {**any** | **host** *source MAC address* | *source MAC address mask*} {**any** | **host** *destination MAC address* | *destination MAC address mask*} [*type mask* | **lsap** *lsap mask* | **aarp** | **amber** | **dec-spanning** | **decnet-iv** | **diagnostic** | **dsm** | **etype-6000** | **etype-8042** | **lat** | **lavc-sca** | **mop-console** | **mop-dump** | **msdos** | **mumps** | **netbios** | **vines-echo** | **vines-ip** | **xns-idp** | 0-*65535*] [**cos** *cos*]
	
	Example:
	
	`Controller(config-ext-macl)# deny any any decnet-iv`
	
	or
	
	`Controller(config-ext-macl)# permit any any`
		In extended MAC access-list configuration mode, specifies to **permit** or **deny** any source MAC address, a source MAC address with a mask, or a specific **host** source MAC address and **any** destination MAC address, destination MAC address with a mask, or a specific destination MAC address.
		
		(Optional) You can also enter these options:
		
		• *type mask*—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.

	Command or Action	Purpose
Step 4	**end** **Example:** `Controller(config-ext-macl)# end`	Returns to privileged EXEC mode.

Related Topics

Restrictions for Configuring Network Security with ACLs, on page 612

Configuring VLAN Maps, on page 641

Applying a MAC ACL to a Layer 2 Interface

Beginning in privileged EXEC mode, follow these steps to apply a MAC access list to control access to a Layer 2 interface:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **mac access-group** {*name*} {**in** | **out** }
4. **end**
5. **show mac access-group** [**interface** *interface-id*]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(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).

	Command or Action	Purpose
Step 3	**mac access-group** {*name*} {**in** | **out** } **Example:** `Controller(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 4	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show mac access-group** [**interface** *interface-id*] **Example:** `Controller# show mac access-group interface gigabitethernet1/0/2`	Displays the MAC access list applied to the interface or all Layer 2 interfaces.
Step 6	**copy running-config startup-config** **Example:** `Controller# 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.

Related Topics

Restrictions for Configuring Network Security with ACLs, on page 612

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**}

     ```
     Controller(config-access-map)# action forward
     ```

   - **action** { **drop**}

     ```
     Controller(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:** `Controller(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:** `Controller(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):	Sets the action for the map entry.

	Command or Action	Purpose
	• **action** { **forward**} `Controller(config-access-map)#` **`action forward`** • **action** { **drop**} `Controller(config-access-map)#` **`action drop`**	
Step 4	**vlan filter** *mapname* **vlan-list** *list* **Example:** `Controller(config)#` **`vlan filter map 1 vlan-list 20-22`**	Applies the VLAN map to one or more VLAN IDs. The list can be a single VLAN ID (22), a consecutive list (10-22), or a string of VLAN IDs (12, 22, 30). Spaces around the comma and hyphen are optional.

Related Topics

Creating a Numbered Standard ACL, on page 627
Creating a Numbered Extended ACL, on page 628
Creating Named MAC Extended ACLs, on page 638
Creating a VLAN Map, on page 643
Applying a VLAN Map to a VLAN, on page 645

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:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**vlan access-map** *name* [**number**] **Example:** `Controller(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*] **Example:** `Controller(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.
Step 4	**action** {**drop** | **forward**} **Example:** `Controller(config-access-map)# action forward`	(Optional) Sets the action for the map entry. The default is to forward.
Step 5	**end** **Example:** `Controller(config-access-map)# end`	Returns to global configuration mode.
Step 6	**show running-config** **Example:** `Controller# show running-config`	Displays the access list configuration.

	Command or Action	Purpose
Step 7	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Configuring VLAN Maps, on page 641

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:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**vlan filter** *mapname* **vlan-list** *list* **Example:** `Controller(config)# vlan filter map 1 vlan-list 20-22`	Applies the VLAN map to one or more VLAN IDs. The list can be a single VLAN ID (22), a consecutive list (10-22), or a string of VLAN IDs (12, 22, 30). Spaces around the comma and hyphen are optional.

	Command or Action	Purpose
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 4	**show running-config** **Example:** `Controller# show running-config`	Displays the access list configuration.
Step 5	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Configuring VLAN Maps, on page 641

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** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**vlan access-map name** [*number*] **Example:** `Controller(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:** `Controller(config-access-map)# action drop log`	Sets the VLAN access map to drop and log IP packets.
Step 4	**exit** **Example:** `Controller(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:** `Controller(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.

	Command or Action	Purpose
Step 6	**end** **Example:** `Controller(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 80: Commands for Displaying Access Lists and Access Groups

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 81: Commands for Displaying VLAN Map Information

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.

```
Controller# 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.

```
Controller(config)# access-list 188 deny tcp any any time-range new_year_day_2006
Controller(config)# access-list 188 permit tcp any any time-range workhours
Controller(config)# end
Controller# 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.

```
Controller(config)# ip access-list extended deny_access
Controller(config-ext-nacl)# deny tcp any any time-range new_year_day_2006
Controller(config-ext-nacl)# exit
Controller(config)# ip access-list extended may_access
Controller(config-ext-nacl)# permit tcp any any time-range workhours
Controller(config-ext-nacl)# end
Controller# 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:

```
Controller(config)# access-list 1 remark Permit only Jones workstation through
Controller(config)# access-list 1 permit 171.69.2.88
Controller(config)# access-list 1 remark Do not allow Smith through
Controller(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:

```
Controller(config)# ip access-list extended telnetting
Controller(config-ext-nacl)# remark Do not allow Jones subnet to telnet out
Controller(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.

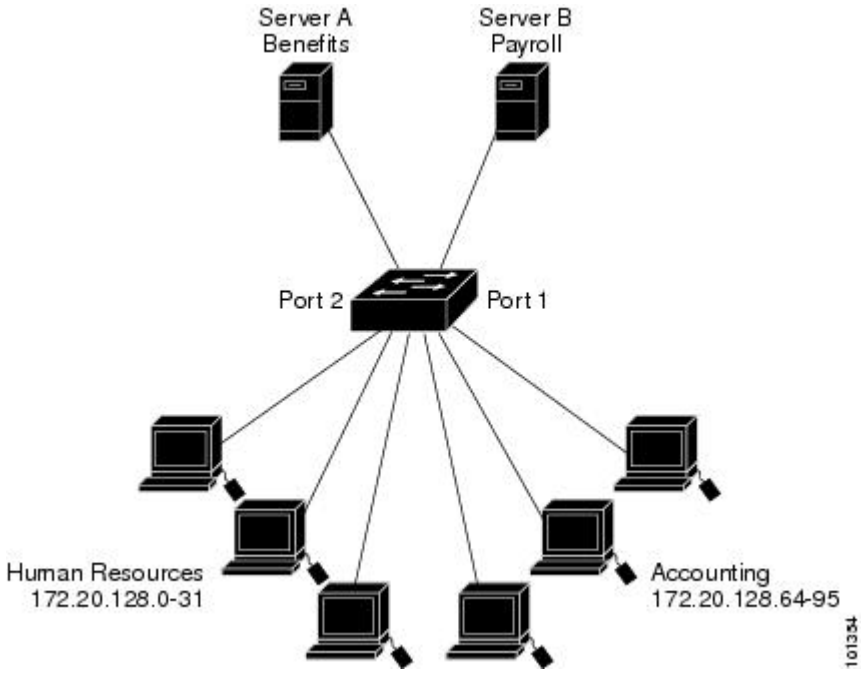
IPv4 ACL Configuration Examples

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

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.

Figure 16: Using Router ACLs to Control Traffic



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.

```
Controller(config)# access-list 6 permit 172.20.128.64 0.0.0.31
Controller(config)# end
Controller# how access-lists
Standard IP access list 6
    10 permit 172.20.128.64, wildcard bits 0.0.0.31
Controller(config)# interface gigabitethernet1/0/1
Controller(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.

```
Controller(config)# access-list 106 permit ip any 172.20.128.64 0.0.0.31
Controller(config)# end
Controller# show access-lists
Extended IP access list 106
    10 permit ip any 172.20.128.64 0.0.0.31
Controller(config)# interface gigabitethernet1/0/1
Controller(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.

```
Controller(config)# access-list 2 permit 36.48.0.3
Controller(config)# access-list 2 deny 36.48.0.0 0.0.255.255
Controller(config)# access-list 2 permit 36.0.0.0 0.255.255.255
Controller(config)# interface gigabitethernet2/0/1
Controller(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.

```
Controller(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 gt 1023
Controller(config)# access-list 102 permit tcp any host 128.88.1.2 eq 25
Controller(config)# access-list 102 permit icmp any any
Controller(config)# interface gigabitethernet2/0/1
Controller(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.

```
Controller(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 eq 23
Controller(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 eq 25
Controller(config)# interface gigabitethernet1/0/1
Controller(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.

```
Controller(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 established
Controller(config)# access-list 102 permit tcp any host 128.88.1.2 eq 25
Controller(config)# interface gigabitethernet1/0/1
Controller(config-if)# ip access-group 102 in
```

Examples: Named 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.

```
Controller(config)# ip access-list standard Internet_filter
Controller(config-ext-nacl)# permit 1.2.3.4
Controller(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.

```
Controller(config)# ip access-list extended marketing_group
Controller(config-ext-nacl)# permit tcp any 171.69.0.0 0.0.255.255 eq telnet
Controller(config-ext-nacl)# deny tcp any any
Controller(config-ext-nacl)# permit icmp any any
Controller(config-ext-nacl)# deny udp any 171.69.0.0 0.0.255.255 lt 1024
Controller(config-ext-nacl)# deny ip any any log
Controller(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.

```
Controller(config)# interface gigabitethernet3/0/2
Controller(config-if)# no switchport
Controller(config-if)# ip address 2.0.5.1 255.255.255.0
Controller(config-if)# ip access-group Internet_filter out
Controller(config-if)# ip access-group marketing_group in
```

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).

```
Controller(config)# time-range no-http
Controller(config)# periodic weekdays 8:00 to 18:00
!
Controller(config)# time-range udp-yes
Controller(config)# periodic weekend 12:00 to 20:00
!
Controller(config)# ip access-list extended strict
Controller(config-ext-nacl)# deny tcp any any eq www time-range no-http
Controller(config-ext-nacl)# permit udp any any time-range udp-yes
!
```

```
Controller(config-ext-nacl)# exit
Controller(config)# interface gigabitethernet2/0/1
Controller(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:

```
Controller(config)# access-list 1 remark Permit only Jones workstation through
Controller(config)# access-list 1 permit 171.69.2.88
Controller(config)# access-list 1 remark Do not allow Smith workstation through
Controller(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:

```
Controller(config)# access-list 100 remark Do not allow Winter to browse the web
Controller(config)# access-list 100 deny host 171.69.3.85 any eq www
Controller(config)# access-list 100 remark Do not allow Smith to browse the web
Controller(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:

```
Controller(config)# ip access-list standard prevention
Controller(config-std-nacl)# remark Do not allow Jones subnet through
Controller(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:

```
Controller(config)# ip access-list extended telnetting
Controller(config-ext-nacl)# remark Do not allow Jones subnet to telnet out
Controller(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.

```
Controller(config)# ip access-list standard stan1
Controller(config-std-nacl)# deny 10.1.1.0 0.0.0.255 log
Controller(config-std-nacl)# permit any log
Controller(config-std-nacl)# exit
Controller(config)# interface gigabitethernet1/0/1
Controller(config-if)# ip access-group stan1 in
Controller(config-if)# end
Controller# 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.

```
Controller(config)# ip access-list extended ext1
Controller(config-ext-nacl)# permit icmp any 10.1.1.0 0.0.0.255 log
Controller(config-ext-nacl)# deny udp any any log
Controller(config-std-nacl)# exit
Controller(config)# interface gigabitethernet1/0/2
Controller(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.

```
Controller(config)# ip access-list extended ip1
Controller(config-ext-nacl)# permit tcp any any
Controller(config-ext-nacl)# exit
Controller(config)# vlan access-map map_1 10
Controller(config-access-map)# match ip address ip1
Controller(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.

```
Controller(config)# ip access-list extended ip2
Controller(config-ext-nacl)# permit udp any any
Controller(config-ext-nacl)# exit
Controller(config)# vlan access-map map_1 20
Controller(config-access-map)# match ip address ip2
Controller(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

```
Controller(config)# access-list 101 permit udp any any
Controller(config)# ip access-list extended igmp-match
Controller(config-ext-nacl)# permit igmp any any
Controller(config)# action forward
Controller(config-ext-nacl)# permit tcp any any
Controller(config-ext-nacl)# exit
Controller(config)# vlan access-map drop-ip-default 10
Controller(config-access-map)# match ip address 101
Controller(config-access-map)# action forward
Controller(config-access-map)# exit
Controller(config)# vlan access-map drop-ip-default 20
Controller(config-access-map)# match ip address igmp-match
Controller(config-access-map)# action drop
Controller(config-access-map)# exit
Controller(config)# vlan access-map drop-ip-default 30
Controller(config-access-map)# match ip address tcp-match
Controller(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

```
Controller(config)# mac access-list extended good-hosts
Controller(config-ext-macl)# permit host 000.0c00.0111 any
Controller(config-ext-macl)# permit host 000.0c00.0211 any
Controller(config-ext-nacl)# exit
Controller(config)# action forward
Controller(config-ext-macl)# mac access-list extended good-protocols
Controller(config-ext-macl)# permit any any vines-ip
Controller(config-ext-nacl)# exit
Controller(config)# vlan access-map drop-mac-default 10
Controller(config-access-map)# match mac address good-hosts
Controller(config-access-map)# action forward
Controller(config-access-map)# exit
Controller(config)# vlan access-map drop-mac-default 20
Controller(config-access-map)# match mac address good-protocols
Controller(config-access-map)# action forward
```

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

```
Controller(config)# vlan access-map drop-all-default 10
Controller(config-access-map)# match ip address tcp-match
Controller(config-access-map)# action forward
Controller(config-access-map)# exit
Controller(config)# vlan access-map drop-all-default 20
Controller(config-access-map)# match mac address good-hosts
Controller(config-access-map)# action forward
```

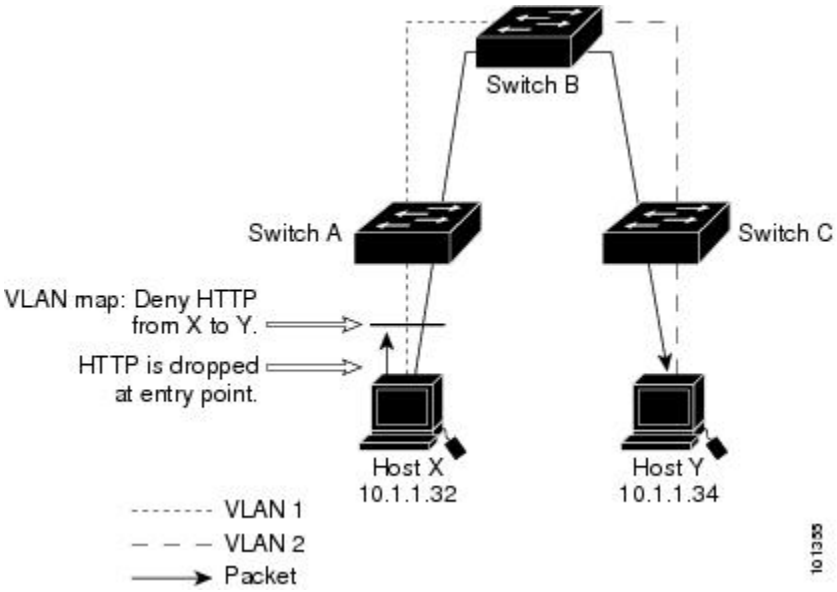
Configuration Examples for Using VLAN Maps in Your Network

Example: 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.

Figure 17: Wiring Closet Configuration



If you do not want HTTP traffic switched from Host X to Host Y, you can configure a VLAN map on Switch A to drop all HTTP traffic from Host X (IP address 10.1.1.32) to Host Y (IP address 10.1.1.34) at Switch A and not bridge it to Switch B.

First, define the IP access list *http* that permits (matches) any TCP traffic on the HTTP port.

```
Controller(config)# ip access-list extended http
Controller(config-ext-nacl)# permit tcp host 10.1.1.32 host 10.1.1.34 eq www
Controller(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.

```
Controller(config)# vlan access-map map2 10
Controller(config-access-map)# match ip address http
Controller(config-access-map)# action drop
Controller(config-access-map)# exit
Controller(config)# ip access-list extended match_all
Controller(config-ext-nacl)# permit ip any any
Controller(config-ext-nacl)# exit
Controller(config)# vlan access-map map2 20
Controller(config-access-map)# match ip address match_all
Controller(config-access-map)# action forward
```

Then, apply VLAN access map *map2* to VLAN 1.

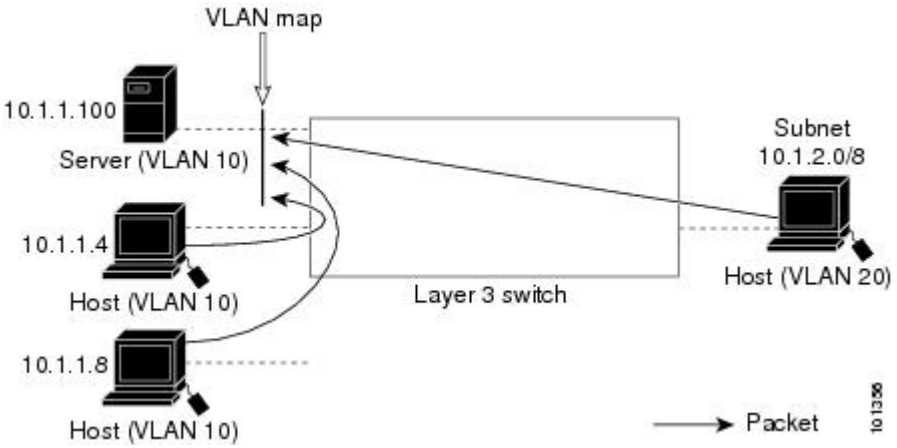
```
Controller(config)# vlan filter map2 vlan 1
```

Example: 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.

Figure 18: Restricting Access to a Server on Another VLAN



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.

```
Controller(config)# ip access-list extended SERVER1_ACL
Controller(config-ext-nacl))# permit ip 10.1.2.0 0.0.0.255 host 10.1.1.100
Controller(config-ext-nacl))# permit ip host 10.1.1.4 host 10.1.1.100
Controller(config-ext-nacl))# permit ip host 10.1.1.8 host 10.1.1.100
Controller(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.

```
Controller(config)# vlan access-map SERVER1_MAP
Controller(config-access-map)# match ip address SERVER1_ACL
Controller(config-access-map)# action drop
Controller(config)# vlan access-map SERVER1_MAP 20
Controller(config-access-map)# action forward
Controller(config-access-map)# exit
```

Apply the VLAN map to VLAN 10.

```
Controller(config)# vlan filter SERVER1_MAP vlan-list 10
```

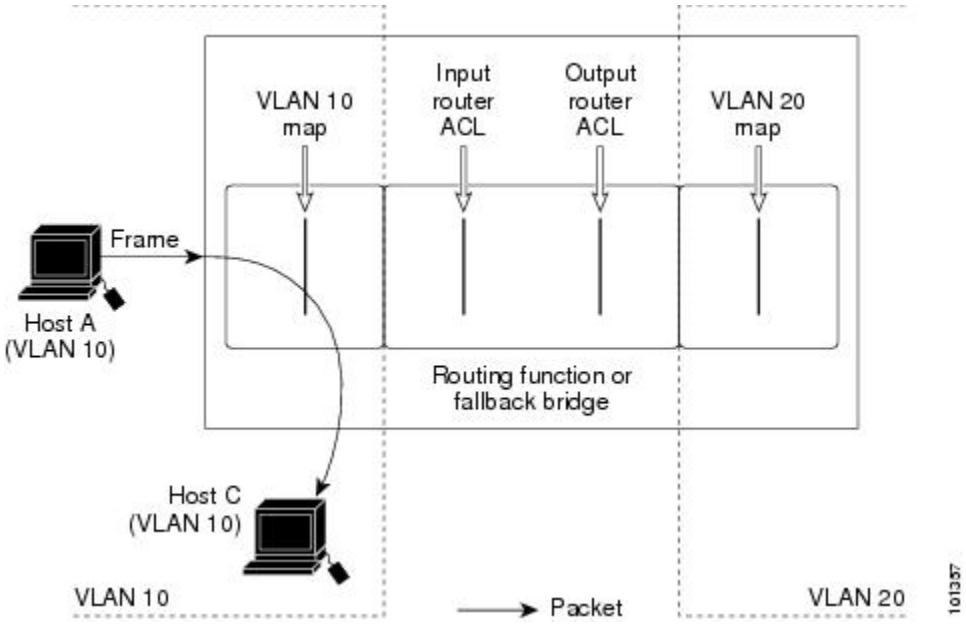
Configuration Examples of Router ACLs and VLAN Maps Applied to VLANs

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

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.

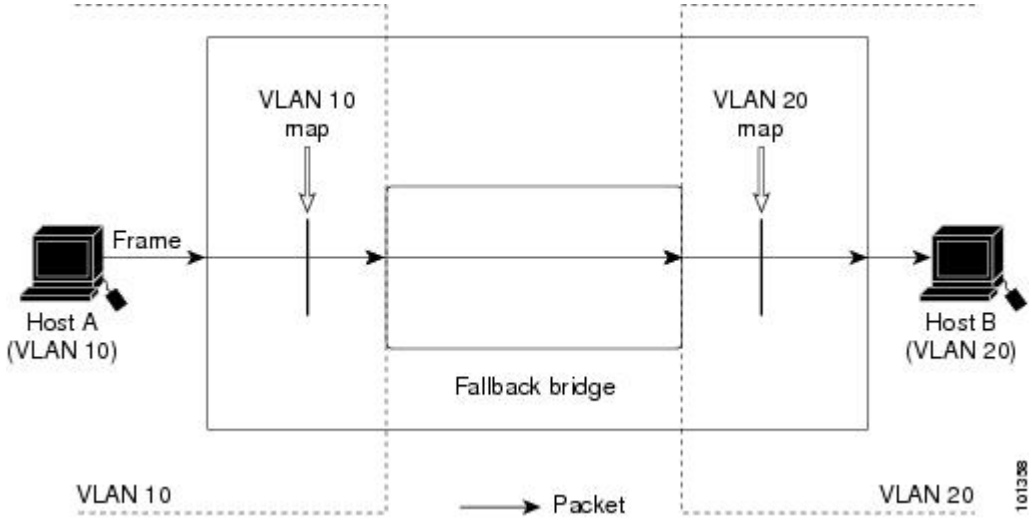
Figure 19: Applying ACLs on Switched Packets



Example: ACLs and 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.

Figure 20: Applying ACLs on Bridged Packets

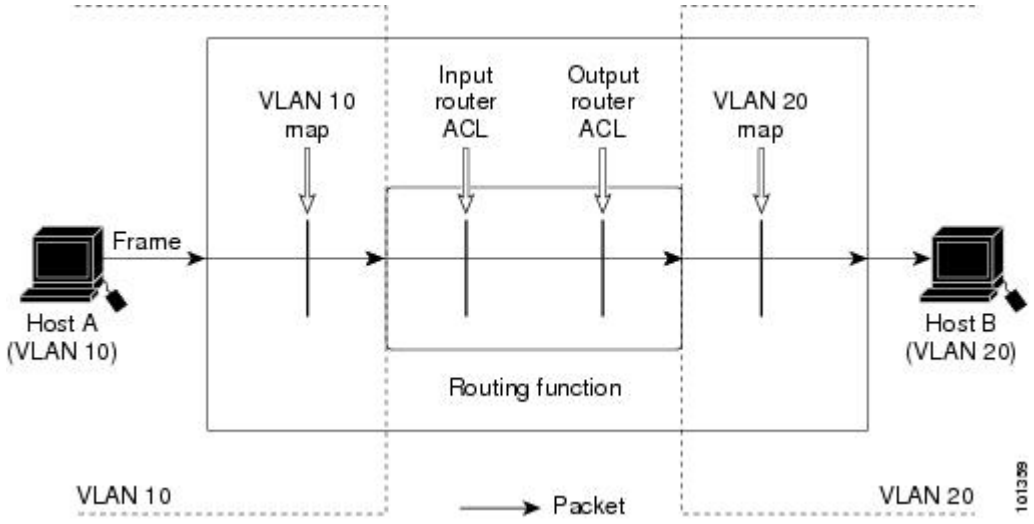


Example: ACLs and Routed Packets

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

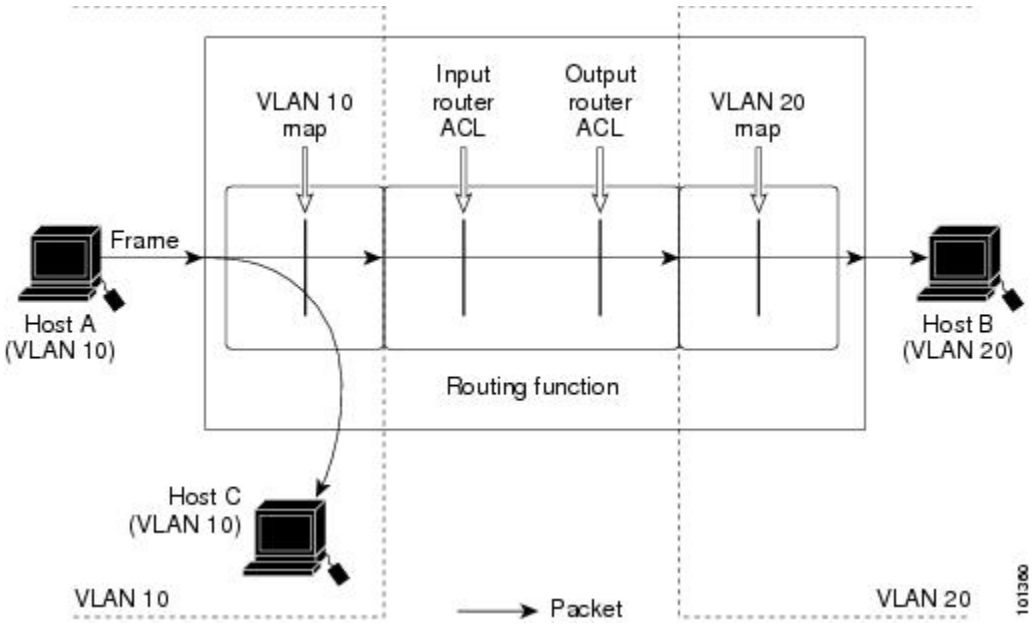
Figure 21: Applying ACLs on Routed Packets



Example: ACLs and 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.

Figure 22: Applying ACLs on Multicast Packets



Additional References

Related Documents

Related Topic	Document Title

Standards and RFCs

Standard/RFC	Title

MIBs

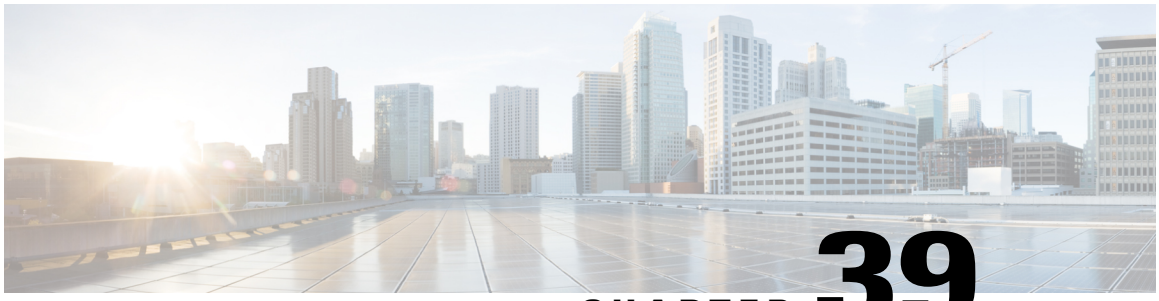
MIB	MIBs Link
	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

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 for ACLs

Feature Name	Releases	Feature Information



CHAPTER 39

Configuring DHCP

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 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.

Related Topics

Prerequisites for Configuring DHCP Snooping and Option 82, on page 676

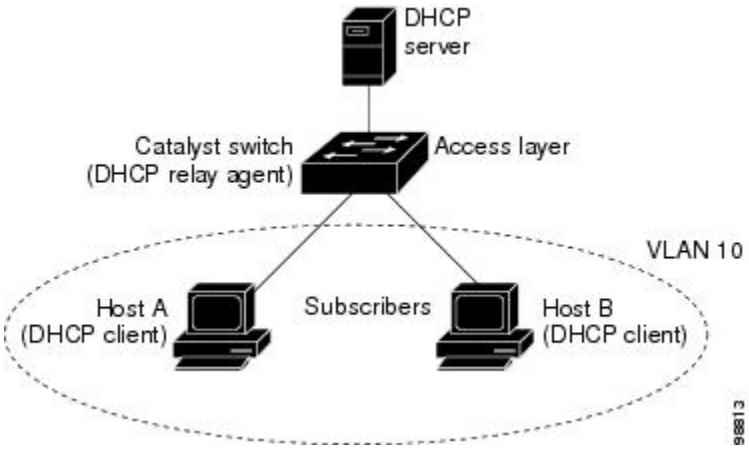
Option-82 Data Insertion

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 23: 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*):

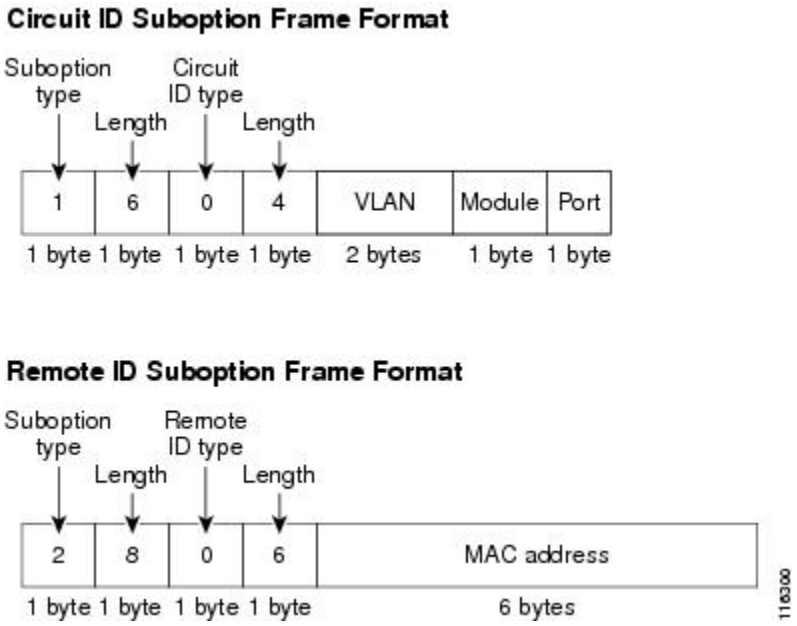
- Circuit-ID suboption fields
  - Suboption type
  - Length of the suboption type
  - Circuit-ID type
  - Length of the circuit-ID type
- Remote-ID suboption fields
  - Suboption type
  - Length of the suboption type
  - Remote-ID type
  - Length of the remote-ID type

In the port field of the circuit ID suboption, the port numbers start at 3. For example, on a Catalyst 3850 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 24: Suboption Packet Formats



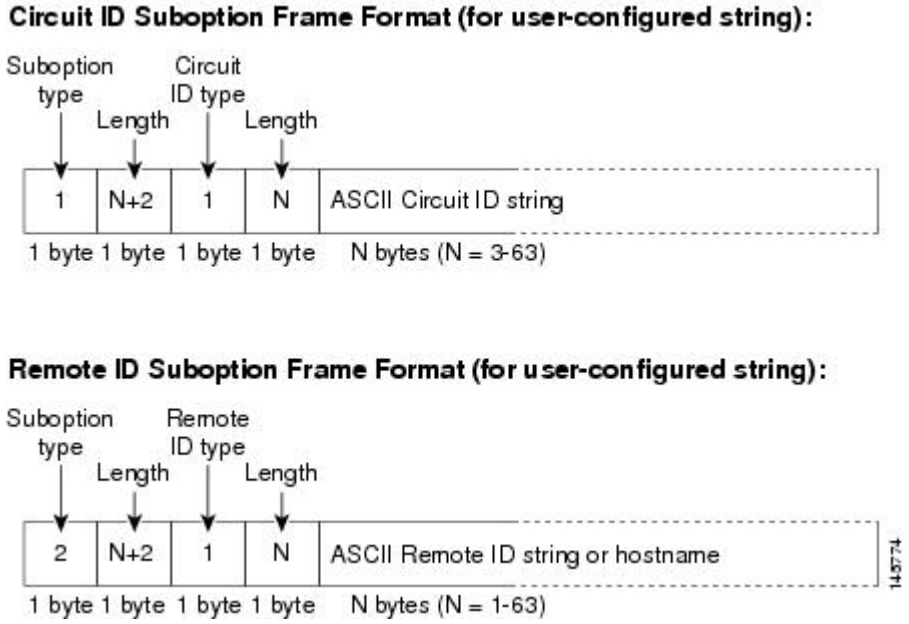
The illustration, *User-Configured Suboption Packet Formats,* shows the packet formats for user-configured remote-ID and circuit-ID suboptions The switch uses these packet formats when DHCP snooping is globally enabled and when the **ip dhcp snooping information option format remote-id** global configuration 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 25: User-Configured Suboption Packet Formats



Cisco IOS DHCP Server Database

During the DHCP-based autoconfiguration process, the designated DHCP server uses the Cisco IOS DHCP server database. It has IP addresses, address bindings, and configuration parameters, such as the boot file.

An address binding is a mapping between an IP address and a MAC address of a host in the Cisco IOS DHCP server database. You can manually assign the client IP address, or the DHCP server can allocate an IP address from a DHCP address pool. 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 abort-timeout values), the update stops.

This is the format of the file with bindings:

```
<initial-checksum>
TYPE DHCP-SNOOPING
VERSION 1
BEGIN
<entry-1> <checksum-1>
<entry-2> <checksum-1-2>
...
...
<entry-n> <checksum-1-2-..-n>
END
```

Each entry in the file is tagged with a checksum value that the switch uses to verify the entries when it reads the file. The initial-checksum entry on the first line distinguishes entries associated with the latest file update from entries associated with a previous file update.

This is an example of a binding file:

```
2bb4c2a1
TYPE DHCP-SNOOPING
VERSION 1
BEGIN
192.1.168.1 3 0003.47d8.c91f 2BB6488E 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 stack master. When a new switch joins the stack, the switch receives the DHCP snooping configuration from the stack master. When a member leaves the stack, all DHCP snooping address bindings associated with the switch age out.

All snooping statistics are generated on the stack master. If a new stack master is elected, the statistics counters reset.

When a stack merge occurs, all DHCP snooping bindings in the stack master are lost if it is no longer the stack master. With a stack partition, the existing stack master is unchanged, and the bindings belonging to the partitioned switches age out. The new master of the partitioned stack begins processing the new incoming DHCP packets.

How to Configure DHCP Features

Default DHCP Snooping Configuration

Table 82: Default DHCP Configuration

Feature	Default Setting
DHCP server	Enabled in Cisco IOS software, requires configuration[9]
DHCP relay agent	Enabled[10]
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[11]	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.

9 The switch responds to DHCP requests only if it is configured as a DHCP server.

10 The switch relays DHCP packets only if the IP address of the DHCP server is configured on the SVI of the DHCP client.
11 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 master. When a new stack master is assigned, the new master downloads the saved binding database from the TFTP server. If the stack master fails, all unsaved bindings are lost. 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.

When a stack merge occurs, the stack master that becomes a stack member loses all of the DHCP lease bindings. With a stack partition, the new master in the partition acts as a new DHCP server without any of the existing DHCP lease bindings.

Configuring the DHCP Relay Agent

Beginning in privileged EXEC mode, follow these steps to enable the DHCP relay agent on the switch:

SUMMARY STEPS

1. **configure terminal**
2. **service dhcp**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**service dhcp** **Example:** `Controller(config)# service dhcp`	Enables the DHCP server and relay agent on your switch. By default, this feature is enabled.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

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. **configure terminal**
2. **interface vlan** *vlan-id*
3. **ip address** *ip-address subnet-mask*
4. **ip helper-address** *address*
5. **end**
6. **interface range** *port-range* or **interface** *interface-id*
7. **switchport mode access**
8. **switchport access vlan** *vlan-id*
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface vlan** *vlan-id* **Example:** `Controller(config)# interface vlan 1`	Creates a switch virtual interface by entering a VLAN ID, and enter interface configuration mode.
Step 3	**ip address** *ip-address subnet-mask* **Example:** `Controller(config-if)# ip address 192.108.1.27 255.255.255.0`	Configures the interface with an IP address and an IP subnet.
Step 4	**ip helper-address** *address* **Example:** `Controller(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 5	**end** **Example:** `Controller(config-if)# end`	Returns to global configuration mode.

	Command or Action	Purpose
Step 6	**interface range** *port-range* or **interface** *interface-id* **Example:** `Controller(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 7	**switchport mode access** **Example:** `Controller(config-if)# switchport mode access`	Defines the VLAN membership mode for the port.
Step 8	**switchport access vlan** *vlan-id* **Example:** `Controller(config-if)# switchport access vlan 1`	Assigns the ports to the same VLAN as configured in Step 2.
Step 9	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.

Prerequisites for Configuring DHCP Snooping and Option 82

The prerequisites for DHCP Snooping and Option 82 are as follows:

- 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.
- 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.
- The following prerequisites apply to DHCP snooping binding database configuration:
  - 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.

- If you want the switch to respond to DHCP requests, it must be configured as a DHCP server.
- 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.
- 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 two list items should be checked for technical accuracy by a subject matter expert:
- You must configure the switch to use the Cisco IOS DHCP server binding database to use it for DHCP snooping.
- You must configure a destination on the DHCP snooping binding database to use the switch for DHCP snooping.
- 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 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.
- When you configure DHCP snooping smart logging, the contents of packets dropped by DHCP are sent to a NetFlow collector. If you configure this feature, make sure that smart logging is globally enabled.

Note Do not enable Dynamic Host Configuration Protocol (DHCP) snooping on RSPAN VLANs. If DHCP snooping is enabled on RSPAN VLANs, DHCP packets might not reach the RSPAN destination port.

Related Topics

DHCP Snooping, on page 666

Enabling DHCP Snooping and Option 82

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

SUMMARY STEPS

1. **configure terminal**
2. **ip dhcp snooping**
3. **ip dhcp snooping vlan** *vlan-range* [**smartlog**]
4. **ip dhcp snooping information option**
5. **ip dhcp snooping information option format remote-id** [**string** *ASCII-string* | **hostname**]
6. **ip dhcp snooping information option allow-untrusted**
7. **interface** *interface-id*
8. **ip dhcp snooping vlan** *vlan* **information option format-type circuit-id** [**override**] **string** *ASCII-string*
9. **ip dhcp snooping trust**
10. **ip dhcp snooping limit rate** *rate*
11. **exit**
12. **ip dhcp snooping verify mac-address**
13. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ip dhcp snooping** **Example:** `Controller(config)# ip dhcp snooping`	Enables DHCP snooping globally.
Step 3	**ip dhcp snooping vlan** *vlan-range* [**smartlog**] **Example:** `Controller(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 hyphens, or a range of VLAN IDs separated by entering the starting and ending VLAN IDs separated by a space. • (Optional) Enter **smartlog** to configure the switch to send the contents of dropped packets to a NetFlow collector.
Step 4	**ip dhcp snooping information option** **Example:** `Controller(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.

	Command or Action	Purpose
Step 5	**ip dhcp snooping information option format remote-id** [**string** *ASCII-string* | **hostname**] **Example:** `Controller(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 6	**ip dhcp snooping information option allow-untrusted** **Example:** `Controller(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 7	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet2/0/1`	Specifies the interface to be configured, and enter interface configuration mode.
Step 8	**ip dhcp snooping vlan** *vlan* **information option format-type circuit-id** [**override**] **string** *ASCII-string* **Example:** `Controller(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). (Optional) Use the **override** keyword when you do not want the circuit-ID suboption inserted in TLV format to define subscriber information.
Step 9	**ip dhcp snooping trust** **Example:** `Controller(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 10	**ip dhcp snooping limit rate** *rate* **Example:** `Controller(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.

	Command or Action	Purpose
Step 11	**exit** **Example:** `Controller(config-if)# exit`	Returns to global configuration mode.
Step 12	**ip dhcp snooping verify mac-address** **Example:** `Controller(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 13	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Enabling DHCP Snooping on Private VLANs

You can enable DHCP snooping on private VLANs. If DHCP snooping is enabled, the configuration is propagated to both a primary VLAN and its associated secondary VLANs. If DHCP snooping is enabled on the primary VLAN, it is also configured on the secondary VLANs.

If DHCP snooping is already configured on the primary VLAN and you configure DHCP snooping with different settings on a secondary VLAN, the configuration for the secondary VLAN does not take effect. You must configure DHCP snooping on the primary VLAN. If DHCP snooping is not configured on the primary VLAN, this message appears when you are configuring DHCP snooping on the secondary VLAN, such as VLAN 200:

```
2w5d:%DHCP_SNOOPING-4-DHCP_SNOOPING_PVLAN_WARNING:DHCP Snooping configuration may not take
 effect on secondary vlan 200. DHCP Snooping configuration on secondary vlan is derived
from its primary vlan.
```

The **show ip dhcp snooping** privileged EXEC command output shows all VLANs, including primary and secondary private VLANs, on which DHCP snooping is enabled.

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 83: 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. **configure terminal**
2. **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*
3. **ip dhcp snooping database timeout** *seconds*
4. **ip dhcp snooping database write-delay** *seconds*
5. **end**
6. **ip dhcp snooping binding** *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id* **expiry** *seconds*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**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*	Specifies the URL for the database agent or the binding file by using one of these forms: • **flash**[*number*]**:**/*filename*

	Command or Action	Purpose
	Example: `Controller(config)# ip dhcp snooping database tftp://10.90.90.90/snooping-rp2`	(Optional) Use the *number* parameter to specify the stack member number of the stack master. 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 3	**ip dhcp snooping database timeout** *seconds* **Example:** `Controller(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 4	**ip dhcp snooping database write-delay** *seconds* **Example:** `Controller(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 5	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 6	**ip dhcp snooping binding** *mac-address* **vlan** *vlan-id* *ip-address* **interface** *interface-id* **expiry** *seconds* **Example:** `Controller# 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. Use this command when you are testing or debugging the switch.

Enabling DHCP Server Port-Based Address Allocation

Beginning in privileged EXEC mode, follow these steps to globally enable port-based address allocation and to automatically generate a subscriber identifier on an interface.

SUMMARY STEPS

1. **configure terminal**
2. **ip dhcp use subscriber-id client-id**
3. **ip dhcp subscriber-id interface-name**
4. **interface** *interface-id*
5. **ip dhcp server use subscriber-id client-id**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ip dhcp use subscriber-id client-id** **Example:** `Controller(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 3	**ip dhcp subscriber-id interface-name** **Example:** `Controller(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 4	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet1/0/1`	Specifies the interface to be configured, and enter interface configuration mode.
Step 5	**ip dhcp server use subscriber-id client-id** **Example:** `Controller(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 6	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Related Topics

Preassigning IP Addresses

Monitoring DHCP Server Port-Based Address Allocation

Table 84: Commands for Displaying DHCP Port-Based Address Allocation Information

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

Related Documents

Related Topic	Document Title

Standards and RFCs

Standard/RFC	Title

MIBs

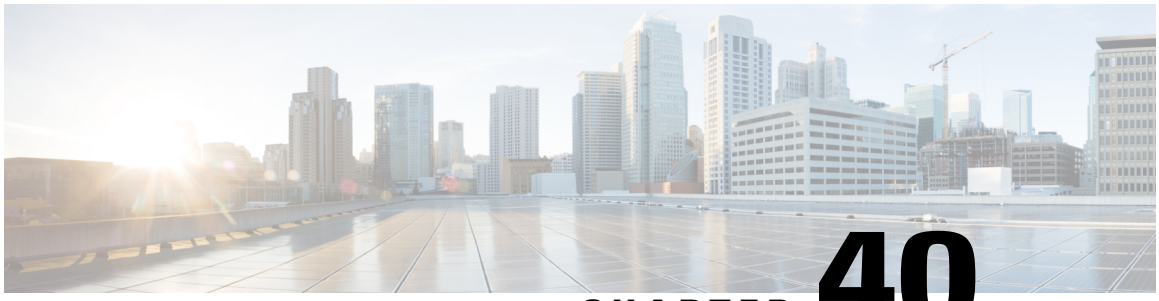
MIB	MIBs Link
	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

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 for DHCP Snooping and Option 82

Feature Name	Releases	Feature Information



CHAPTER 40

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 for the Catalyst 3850 Switch:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 IP Source Guard

This section lists the prerequisites for IP Source Guard.

Restrictions on IP Source Guard

This section lists the restrictions on IP Source Guard.

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 Catalyst 3850 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 master 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.

- When configuring IP source guard on interfaces on which a private VLAN is configured, port security is not supported.
- IP source guard is not supported on EtherChannels.
- You can enable this feature when 802.1x port-based authentication is enabled.
- When you configure IP source guard smart logging, packets with a source address other than the specified address or an address learned by DHCP are denied, and the packet contents are sent to a NetFlow collector. If you configure this feature, make sure that smart logging is globally enabled.
- In a switch stack, if IP source guard is configured on a stack member interface and you remove the the configuration of that switch by entering the **no switch** *stack-member-number* **provision** global configuration command, the interface static bindings are removed from the binding table, but they are not removed from the running configuration. If you again provision the switch by entering the **switch** *stack-member-number* **provision** command, the binding is restored.

  To remove the binding from the running configuration, you must disable IP source guard before entering the **no switch provision** command. The configuration is also removed if the switch reloads while the interface is removed from the binding table.

How to Configure IP Source Guard

Enabling IP Source Guard

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip verify source** [**mac-check**]
4. Use one of the following:
   - **ip verify source[smartlog]**
   - **ip verify source port-security**
5. **exit**
6. **ip source binding** *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id*
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet 1/0/1`	Specifies the interface to be configured, and enters interface configuration mode.
Step 3	**ip verify source** [**mac-check**] **Example:** `Controller(config-if)# ip verify source`	Enables IP source guard with source IP address filtering. (Optional) **mac-check—**Enables IP Source Guard with source IP address and MAC address filtering.
Step 4	Use one of the following: • **ip verify source[smartlog]** • **ip verify source port-security**	Enables IP source guard with source IP address filtering. • (Optional) Enter **smartlog** to configure the switch to send the contents of dropped packets to a NetFlow collector. Enables IP source guard with source IP and MAC address filtering.

	Command or Action	Purpose

	Example:
	`Controller(config-if)# ip verify source`
	or
	`Controller(config-if)# ip verify source port-security`
		When you enable both IP source guard and port security by using the **ip verify source port-security** interface configuration command, there are two caveats:
		• The DHCP server must support option 82, or the client is not assigned an IP address.
		• The MAC address in the DHCP packet is not learned as a secure address. The MAC address of the DHCP client is learned as a secure address only when the switch receives non-DHCP data traffic.

Step 5	**exit** **Example:** `Controller(config-if)# exit`	Returns to global configuration mode.
Step 6	**ip source binding** *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id* **Example:** `Controller(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:** `Controller(config)# end`	Returns to privileged EXEC mode.

Enabling IP source guard with source IP and MAC filtering on VLANs 10 and 11

```
Controller# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Controller(config)# interface gigabitethernet 1/0/1
Controller(config-if)# ip verify source
Controller(config-if)# exit
Controller(config)# ip source binding 0100.0022.0010 vlan 10 10.0.0.2 interface
gigabitethernet 1/0/1
Controller(config)# ip source binding 0100.0230.0002 vlan 11 10.0.0.4 interface
gigabitethernet 1/0/1
Controller(config)# end
```

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. This requirement also applies to IPSG with static hosts on a private VLAN host port.

SUMMARY STEPS

1. **configure terminal**
2. **ip device tracking**
3. **interface** *interface-id*
4. **switchport mode access**
5. **switchport access vlan** *vlan-id*
6. **ip verify source**[**tracking**] [**mac-check**]
7. **ip device tracking maximum** *number*
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ip device tracking** **Example:** `Controller(config)# ip device tracking`	Turns on the IP host table, and globally enables IP device tracking.
Step 3	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode.
Step 4	**switchport mode access** **Example:** `Controller(config-if)# switchport mode access`	Configures a port as access.

	Command or Action	Purpose
Step 5	**switchport access vlan** *vlan-id* **Example:** `Controller(config-if)# switchport access vlan 10`	Configures the VLAN for this port.
Step 6	**ip verify source**[**tracking**] [**mac-check**] **Example:** `Controller(config-if)# ip verify source tracking mac-check`	Enables IP source guard with source IP address filtering. (Optional) **tracking—**Enables IP source guard for static hosts. (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 7	**ip device tracking maximum** *number* **Example:** `Controller(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 8	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Eight Examples

This example shows how to stop IPSG with static hosts on an interface.

```
Controller(config-if)# no ip verify source
Controller(config-if)# no ip device tracking max
```

This example shows how to enable IPSG with static hosts on a port.

```
Controller(config)# ip device tracking
Controller(config-if)# ip device tracking maximum 10
Controller(config-if)# ip verify source tracking
```

This example shows how to enable IPSG for static hosts with IP filters on a Layer 2 access port and to verify the valid IP bindings on the interface Gi1/0/3:

```
Controller# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Controller(config)# ip device tracking
Controller(config)# interface gigabitethernet1/0/3
Controller(config-if)# switchport mode access
Controller(config-if)# switchport access vlan 10
Controller(config-if)# ip device tracking maximum 5
Controller(config-if)# ip verify source tracking
```

```
Controller(config-if)# end

Controller# show ip verify source
Interface  Filter-type  Filter-mode  IP-address       Mac-address        Vlan
---------  -----------  -----------  ---------------  -----------------  ----
Gi1/0/3      ip trk       active       40.1.1.24                         10
Gi1/0/3      ip trk       active       40.1.1.20                         10
Gi1/0/3      ip trk       active       40.1.1.21                         10
```

This example shows how to enable IPSG for static hosts with IP-MAC filters on a Layer 2 access port, to verify the valid IP-MAC bindings on the interface Gi1/0/3, and to verify that the number of bindings on this interface has reached the maximum:

```
Controller# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Controller(config)# ip device tracking
Controller(config)# interface gigabitethernet1/0/3
Controller(config-if)# switchport mode access
Controller(config-if)# switchport access vlan 1
Controller(config-if)# ip device tracking maximum 5

Controller(config-if)# ip verify source tracking
Controller(config-if)# end

Controller# show ip verify source
Interface  Filter-type  Filter-mode  IP-address       Mac-address        Vlan
---------  -----------  -----------  ---------------  -----------------  ----
Gi1/0/3      ip trk       active       deny-all                          1
```

This example displays all IP or MAC binding entries for all interfaces. The CLI displays all active as well as inactive entries. When a host is learned on a interface, the new entry is marked as active. When the same host is disconnected from that interface and connected to a different interface, a new IP or MAC binding entry displays as active as soon as the host is detected. The old entry for this host on the previous interface is marked as INACTIVE.

```
Controller# show ip device tracking all
IP Device Tracking for wireless clients = Enabled
Global IP Device Tracking for wired clients = Enabled
Global IP Device Tracking Probe Count = 3
Global IP Device Tracking Probe Interval = 30
-----------------------------------------------------------------------------------------------

  IP Address     MAC Address   Vlan  Interface           Probe-Timeout      STATE
-----------------------------------------------------------------------------------------------
200.1.1.8       0001.0600.0000  8   GigabitEthernet1/0/1                   INACTIVE
200.1.1.9       0001.0600.0000  8   GigabitEthernet1/0/1                   INACTIVE
200.1.1.10      0001.0600.0000  8   GigabitEthernet1/0/1                   INACTIVE
200.1.1.1       0001.0600.0000  9   GigabitEthernet1/0/2                   ACTIVE
200.1.1.1       0001.0600.0000  8   GigabitEthernet1/0/1                   INACTIVE
200.1.1.2       0001.0600.0000  9   GigabitEthernet1/0/2                   ACTIVE
200.1.1.2       0001.0600.0000  8   GigabitEthernet1/0/1                   INACTIVE
200.1.1.3       0001.0600.0000  9   GigabitEthernet1/0/2                   ACTIVE
200.1.1.3       0001.0600.0000  8   GigabitEthernet1/0/1                   INACTIVE
200.1.1.4       0001.0600.0000  9   GigabitEthernet1/0/2                   ACTIVE
200.1.1.4       0001.0600.0000  8   GigabitEthernet1/0/1                   INACTIVE
200.1.1.5       0001.0600.0000  9   GigabitEthernet1/0/2                   ACTIVE
200.1.1.5       0001.0600.0000  8   GigabitEthernet1/0/1                   INACTIVE
200.1.1.6       0001.0600.0000  8   GigabitEthernet1/0/1                   INACTIVE
200.1.1.7       0001.0600.0000  8   GigabitEthernet1/0/1                   INACTIVE
```

This example displays all active IP or MAC binding entries for all interfaces:

```
Controller# show ip device tracking all active
IP Device Tracking for wireless clients = Enabled
Global IP Device Tracking for wired clients = Enabled
Global IP Device Tracking Probe Count = 3
Global IP Device Tracking Probe Interval = 30
```

```
------------------------------------------------------------------------------------------------

  IP Address     MAC Address   Vlan  Interface           Probe-Timeout       STATE
------------------------------------------------------------------------------------------------
200.1.1.1       0001.0600.0000  9    GigabitEthernet1/0/1                  ACTIVE
200.1.1.2       0001.0600.0000  9    GigabitEthernet1/0/1                  ACTIVE
200.1.1.3       0001.0600.0000  9    GigabitEthernet1/0/1                  ACTIVE
200.1.1.4       0001.0600.0000  9    GigabitEthernet1/0/1                  ACTIVE
200.1.1.5       0001.0600.0000  9    GigabitEthernet1/0/1                  ACTIVE
```

This example displays all inactive IP or MAC binding entries for all interfaces. The host was first learned on GigabitEthernet 1/0/1 and then moved to GigabitEthernet 0/2. the IP or MAC binding entries learned on GigabitEthernet1/ 0/1 are marked as inactive.

```
Controller# show ip device tracking all inactive
IP Device Tracking for wireless clients = Enabled
Global IP Device Tracking for wired clients= Enabled
Global IP Device Tracking Probe Count = 3
Global IP Device Tracking Probe Interval = 30
------------------------------------------------------------------------------------------------

  IP Address     MAC Address   Vlan  Interface           Probe-Timeout       STATE
------------------------------------------------------------------------------------------------
200.1.1.8       0001.0600.0000  8    GigabitEthernet1/0/1                  INACTIVE
200.1.1.9       0001.0600.0000  8    GigabitEthernet1/0/1                  INACTIVE
200.1.1.10      0001.0600.0000  8    GigabitEthernet1/0/1                  INACTIVE
200.1.1.1       0001.0600.0000  8    GigabitEthernet1/0/1                  INACTIVE
200.1.1.2       0001.0600.0000  8    GigabitEthernet1/0/1                  INACTIVE
200.1.1.3       0001.0600.0000  8    GigabitEthernet1/0/1                  INACTIVE
200.1.1.4       0001.0600.0000  8    GigabitEthernet1/0/1                  INACTIVE
200.1.1.5       0001.0600.0000  8    GigabitEthernet1/0/1                  INACTIVE
200.1.1.6       0001.0600.0000  8    GigabitEthernet1/0/1                  INACTIVE
200.1.1.7       0001.0600.0000  8    GigabitEthernet1/0/1                  INACTIVE
```

This example displays the count of all IP device tracking host entries for all interfaces:

```
Controller# show ip device tracking all count
Total IP Device Tracking Host entries: 5
------------------------------------------------------------------------
  Interface           Maximum Limit           Number of Entries
------------------------------------------------------------------------
Gi1/0/3                  5
```

Configuring IP Source Guard for Static Hosts on a Private VLAN Host Port

You must globally configure the **ip device tracking maximum** *limit-number* interface configuration command globally for IPSG for static hosts to work. If you only configure this command on a port without enabling IP device tracking globally or setting an IP device tracking maximum on that interface, IPSG with static hosts will reject all the IP traffic from that interface. This requirement also applies to IPSG with static hosts on a Layer 2 access port.

SUMMARY STEPS

1. **configure terminal**
2. **vlan** *vlan-id1*
3. **private-vlan primary**
4. **exit**
5. **vlan** *vlan-id2*
6. **private-vlan isolated**
7. **exit**
8. **vlan** *vlan-id1*
9. **private-vlan association 201**
10. **exit**
11. **interface** *interface-id*
12. **switchport mode private-vlan host**
13. **switchport private-vlan host-association** *vlan-id1 vlan-id2*
14. **ip device tracking maximum** *number*
15. **ip verify source tracking**
16. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**vlan** *vlan-id1* **Example:** `Controller(config)# vlan 10`	Enters VLAN configuration mode.
Step 3	**private-vlan primary** **Example:** `Controller(config-vlan)# private-vlan primary`	Establishes a primary VLAN on a private VLAN port.
Step 4	**exit** **Example:** `Controller(config-vlan)# exit`	Exits VLAN configuration mode.

	Command or Action	Purpose
Step 5	**vlan** *vlan-id2* **Example:** `Controller(config)# vlan 20`	Enters configuration VLAN mode for another VLAN.
Step 6	**private-vlan isolated** **Example:** `Controller(config-vlan)# private-vlan isolated`	Establishes an isolated VLAN on a private VLAN port.
Step 7	**exit** **Example:** `Controller(config-vlan)# exit`	Exits VLAN configuration mode.
Step 8	**vlan** *vlan-id1* **Example:** `Controller(config)# vlan 10`	Enters VLAN configuration mode.
Step 9	**private-vlan association 201** **Example:** `Controller(config-vlan)# private-vlan association 201`	Associates the VLAN on an isolated private VLAN port.
Step 10	**exit** **Example:** `Controller(config-vlan)# exit`	Exits VLAN configuration mode.
Step 11	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet 1/0/1`	Enters interface configuration mode.
Step 12	**switchport mode private-vlan host** **Example:** `Controller(config-if)# switchport mode private-vlan host`	(Optional) Establishes a port as a private VLAN host.

	Command or Action	Purpose
Step 13	**switchport private-vlan host-association** *vlan-id1 vlan-id2* **Example:** `Controller(config-if)# switchport private-vlan host-association 10 20`	(Optional) Associates this port with the corresponding private VLAN.
Step 14	**ip device tracking maximum** *number* **Example:** `Controller(config-if)# ip device tracking maximum 8`	Establishes a maximum for the number of static IPs that the IP device tracking table allows on the port. The maximum is 10. **Note** You must globally configure the **ip device tracking maximum** *number* interface command for IPSG for static hosts to work.
Step 15	**ip verify source tracking** **Example:** `Controller(config-if)# ip verify source tracking`	Activates IPSG for static hosts on this port.
Step 16	**end** **Example:** `Controller(config-if)# end`	Exits interface configuration mode.

How to Enable IPSG for Static Hosts with IP Filters on a Private VLAN Host Port

This example shows how to enable IPSG for static hosts with IP filters on a private VLAN host port:

```
Controller(config)# vlan 200
Controller(config-vlan)# private-vlan primary
Controller(config-vlan)# exit
Controller(config)# vlan 201
Controller(config-vlan)# private-vlan isolated
Controller(config-vlan)# exit
Controller(config)# vlan 200
Controller(config-vlan)# private-vlan association 201
Controller(config-vlan)# exit
Controller(config)# interface gigabitethernet1/0/3
Controller(config-if)# switchport mode private-vlan host
Controller(config-if)# switchport private-vlan host-association 200 201
Controller(config-if)# ip device tracking maximum 8
Controller(config-if)# ip verify source tracking

Controller# show ip device tracking all
IP Device Tracking for wireless clients = Enabled
Global IP Device Tracking for wired clients= Enabled
Global IP Device Tracking Probe Count = 3
Global IP Device Tracking Probe Interval = 30
```

```
------------------------------------------------------------------------------------------------

  IP Address     MAC Address   Vlan  Interface           Probe-Timeout      STATE
------------------------------------------------------------------------------------------------
40.1.1.24       0000.0000.0304  200  GigabitEthernet1/0/3                   ACTIVE
40.1.1.20       0000.0000.0305  200  GigabitEthernet1/0/3                   ACTIVE
40.1.1.21       0000.0000.0306  200  GigabitEthernet1/0/3                   ACTIVE
40.1.1.22       0000.0000.0307  200  GigabitEthernet1/0/3                   ACTIVE
40.1.1.23       0000.0000.0308  200  GigabitEthernet1/0/3                   ACTIVE
```

The output shows the five valid IP-MAC bindings that have been learned on the interface Fa0/3. For the private VLAN cases, the bindings are associated with primary VLAN ID. So, in this example, the primary VLAN ID, 200, is shown in the table.

```
Controller# show ip verify source
Interface  Filter-type  Filter-mode  IP-address       Mac-address        Vlan
---------  -----------  -----------  ---------------  -----------------  -----
Gi1/0/3      ip trk       active       40.1.1.23                          200
Gi1/0/3      ip trk       active       40.1.1.24                          200
Gi1/0/3      ip trk       active       40.1.1.20                          200
Gi1/0/3      ip trk       active       40.1.1.21                          200
Gi1/0/3      ip trk       active       40.1.1.22                          200
Gi1/0/3      ip trk       active       40.1.1.23                          201
Gi1/0/3      ip trk       active       40.1.1.24                          201
Gi1/0/3      ip trk       active       40.1.1.20                          201
Gi1/0/3      ip trk       active       40.1.1.21                          201
Gi1/0/3      ip trk       active       40.1.1.22                          201
```

The output shows that the five valid IP-MAC bindings are on both the primary and secondary VLAN.

Monitoring IP Source Guard

Table 85: 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 86: Global Configuration Commands

Command	Purpose

Table 87: 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

Related Documents

Related Topic	Document Title
IP Source Guard command reference	<TBD>
Platform-independent configuration information	<TBD>

Standards and RFCs

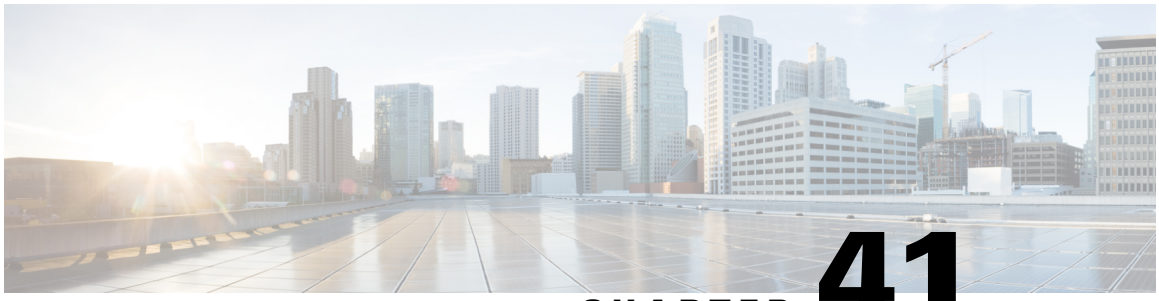
Standard/RFC	Title

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



CHAPTER 41

Configuring Dynamic ARP Inspection

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Dynamic ARP Inspection

Restrictions for Dynamic ARP Inspection

This section lists the restrictions and guidelines for configuring Dynamic ARP Inspection on the Catalyst 3850 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, EtherChannel ports, and private VLAN 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.

  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.

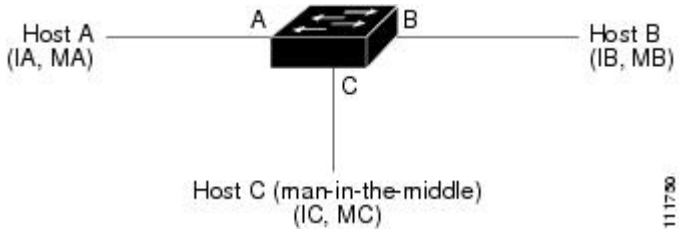
  If the EtherChannel receives more ARP packets than the configured rate, the channel (including all physical ports) is placed in the error-disabled state.
- Make sure to limit the rate of ARP packets on incoming trunk ports. Configure trunk ports with higher rates to reflect their aggregation and to handle 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.
- When you configure dynamic ARP inspection smart logging, the contents of all packets in the log buffer (by default, all dropped packets) are sent to a NetFlow collector. If you configure this feature, make sure that smart logging is globally enabled. For more information about smart logging, see the “Configuring Smart Logging” section on page xxx.

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 26: 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

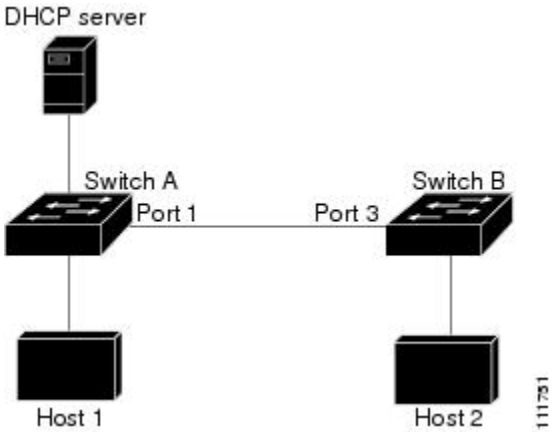
Dynamic ARP inspection associates a trust state with each interface on the switch. Packets arriving on trusted interfaces bypass all dynamic ARP inspection validation checks, and those arriving on untrusted interfaces undergo the dynamic ARP inspection validation process.

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 27: 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 theip arp inspection limitinterface 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.
Feature	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.
Dynamic ARP inspection	No ARP ACLs are defined.
Interface trust state	No checks are performed.
Rate limit of incoming ARP packets	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.
ARP ACLs for non-DHCP environments	All denied or dropped ARP packets are logged.

Restrictions for Dynamic ARP Inspection

This section lists the restrictions and guidelines for configuring Dynamic ARP Inspection on the Catalyst 3850 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, EtherChannel ports, and private VLAN 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.

  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.

- When you configure dynamic ARP inspection smart logging, the contents of all packets in the log buffer (by default, all dropped packets) are sent to a NetFlow collector. If you configure this feature, make sure that smart logging is globally enabled. For more information about smart logging, see the "Configuring Smart Logging" section on page xxx.

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.

Beginning in privileged EXEC mode, follow these steps to configure an ARP ACL on Switch A. This procedure is required in non-DHCP environments.

SUMMARY STEPS

1. **Configureterminal**
2. **arp access-list** *acl-name*
3. **permit ip host** *sender-ip* **mac host** *sender-mac* log
4. exit
5. **ip arp inspection filter** *arp-acl-name* **vlan** *vlan-range* [**static**]
6. **ip arp inspection smartlog**
7. **interface** *interface-id*
8. **no ip arp inspection trust**
9. **end**
10. **show arp access-list** acl-name **show ip arp inspection vlan** *vlan-range* **show ip arp inspection interfaces**
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**Configureterminal**	Enter global configuration mode.
Step 2	**arp access-list** *acl-name*	Define an ARP ACL, and enter 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.
Step 3	**permit ip host** *sender-ip* **mac host** *sender-mac* log	Permit 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. • (Optional) Specify **log** to log a packet in the log buffer when it matches the access control entry (ACE). Matches are logged if you also configure the matchlog keyword in theip arp inspection vlan logging global configuration command. For more information, see the section, “Configuring the Log Buffer."
Step 4	exit	Return to global configuration mode.
Step 5	**ip arp inspection filter** *arp-acl-name* **vlan** *vlan-range* [**static**]	Apply the 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 6	**ip arp inspection smartlog**	Specify that whatever packets are currently being logged are also smart-logged. By default, all dropped packets are logged.
Step 7	**interface** *interface-id*	Specify the Switch A interface that is connected to Switch B, and enter interface configuration mode.
Step 8	**no ip arp inspection trust**	Configure the Switch A interface that is connected to Switch B as untrusted. By default, all interfaces are untrusted.

	Command or Action	Purpose
		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. For more information, see the section, “Configuring the Log Buffer."
Step 9	**end**	Return to privileged EXEC mode.
Step 10	**show arp access-list** acl-name **show ip arp inspection vlan** *vlan-range* **show ip arp inspection interfaces**	Verify your entries.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the ARP ACL, use the **no arp access-list** global configuration command. To remove the ARP ACL attached to a VLAN, use the **no ip arp inspection filter** *arp-acl-name* **vlan** *vlan-range* global configuration command.

This example shows how to configure an ARP ACL called host2 on Switch A, to permit ARP packets from Host 2 (IP address 1.1.1.1 and MAC address 0001.0001.0001), to apply the ACL to VLAN 1, and to configure port 1 on Switch A as untrusted:

```
Controller(config)#arp access-list host2
 Controller(config-arp-acl)#permit ip host 1.1.1.1 mac host 1.1.1
 Controller(config-arp-acl)# exit
Controller(config)# ip arp inspection filter host2 vlan 1

Controller(config)# interface gigabitethernet1/0/1

 Controller(config-if)# no ip arp inspection trust
```

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.

Beginning in privileged EXEC mode, follow these steps to configure dynamic ARP inspection. You must perform this procedure on both switches. This procedure is required.

SUMMARY STEPS

1. **show cdp neighbors**
2. **configure terminal**
3. **ip arp inspection vlan** *vlan-range*
4. **ip arp inspection smartlog**
5. **Interface***interface-id*
6. **ip arp inspection trust**
7. **end**
8. **show ip arp inspection interfacesshow ip arp inspection vlan** *vlan-range*
9. **show ip dhcp snooping binding**
10. **show ip arp inspection statistics vlan** *vlan-range*
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show cdp neighbors** **Example:**	Verify the connection between the switches.
Step 2	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 3	**ip arp inspection vlan** *vlan-range* **Example:**	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 4	**ip arp inspection smartlog** **Example:**	(Optional). Specify that whatever packets are currently being logged are also smart-logged. By default, all dropped packets are logged.
Step 5	**Interface***interface-id* **Example:**	Specify the interface connected to the other switch, and enter interface configuration mode.
Step 6	**ip arp inspection trust** **Example:**	Configure 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.

	Command or Action	Purpose
		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. For more information, see the “Configuring the Log Buffer” section on page xxx.
Step 7	**end** **Example:**	Return to privileged EXEC mode.
Step 8	**show ip arp inspection interfacesshow ip arp inspection vlan** *vlan-range* **Example:**	Verify the dynamic ARP inspection configuration.
Step 9	**show ip dhcp snooping binding** **Example:**	Verify the DHCP bindings.
Step 10	**show ip arp inspection statistics vlan** *vlan-range* **Example:**	Check the dynamic ARP inspection statistics.
Step 11	**copy running-config startup-config** **Example:**	(Optional) Save your entries in the configuration file.

To disable dynamic ARP inspection, use the **no ip arp inspection vlan** *vlan-range* global configuration command. To return the interfaces to an untrusted state, use the **no ip arp inspection trust** interface configuration command.

This example shows how to configure dynamic ARP inspection on Switch A in VLAN 1. You would perform a similar procedure on Switch B:

```
Controller(config)# ip arp inspection vlan 1

Controller(config)# interface gigabitethernet1/0/1
Controller(config-if)#ip arp inspection trust
```

How to Limit 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 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.

For configuration guidelines for rate limiting trunk ports and EtherChannel ports, see the section, “Dynamic ARP Inspection Configuration Guidelines."

To return to the default rate-limit configuration, use the no ip arp inspection limit interface configuration command. To disable error recovery for dynamic ARP inspection, use the **no errdisable recovery cause arp-inspection** global configuration command.

Beginning in privileged EXEC mode, follow these steps to limit the rate of incoming ARP packets. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip arp inspection limit {rate pps [burst interval seconds] | none}**
4. **exit**
5. **errdisable detect cause arp-inspection** and **errdisable recovery causearp-inspection errdisable recovery interval** *interval*
6. **exit**
7. **show ip arp inspection interfaces show errdisable recovery**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface to be rate-limited, and enter interface configuration mode.
Step 3	**ip arp inspection limit {rate pps [burst interval seconds] | none}**	Limit the rate of incoming ARP requests and responses on the interface. Limit the rate of incoming ARP requests and responses on the interface. The default rate is 15 pps on untrusted interfaces and unlimited on trusted interfaces. The burst interval is 1 second. 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.

	Command or Action	Purpose
		• (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 4	**exit**	Return to global configuration mode.
Step 5	**errdisable detect cause arp-inspection** and **errdisable recovery causearp-inspection errdisable recovery interval** *interval*	(Optional) Enable 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 6	**exit**	Return to privileged EXEC mode.
Step 7	**show ip arp inspection interfaces** **show errdisable recovery**	Verify your settings.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

How to Perform 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. Beginning in privileged EXEC mode, follow these steps to perform specific checks on incoming ARP packets.

This procedure is optional.

To disable checking, use the**no ip arp inspection validate [src-mac] [dst-mac] [ip]** global configuration command. To display statistics for forwarded, dropped, and MAC and IP validation failure packets, use the **show ip arp inspection statistics** privileged EXEC command.

SUMMARY STEPS

1. **configure terminal**
2. **ip arp inspection validate {[src-mac] [dst-mac] [ip]}**
3. **exit**
4. **show ip arp inspection vlan** *vlan-range*
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**ip arp inspection validate {[src-mac] [dst-mac] [ip]}**	Perform 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. • 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 3	**exit**	Return to privileged EXEC mode.
Step 4	**show ip arp inspection vlan** *vlan-range*	Verify your settings.
Step 5	**copy running-config startup-config**	(Optional) Save 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.

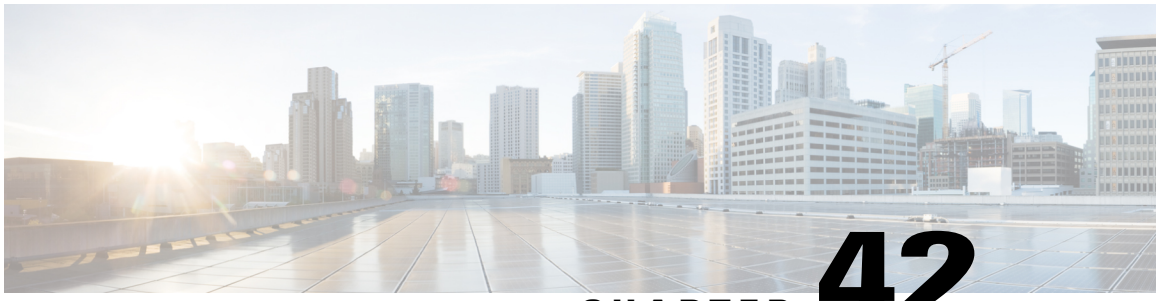
Command	Description
show ip arp inspection statistics [**vlan** *vlan-range*]	Displays statistics for forwarded, dropped, MAC validation failure, IP validation failure, ACL permitted and denied, and DHCP permitted and denied packets for the specified VLAN. If no VLANs are specified or if a range is specified, 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).



CHAPTER 42

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.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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.

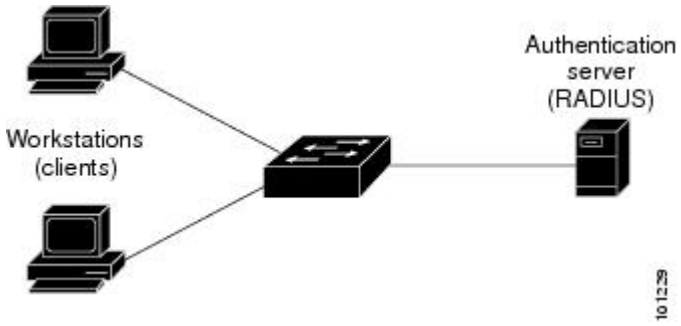
Prerequisites for 802.1x Port-Based Authentication

This section lists the prerequisites for 802.1x Port-Based Authentication.

Device Roles

With 802.1x port-based authentication, the devices in the network have specific roles.

Figure 28: 802.1x Device Roles



- *Client*—the device (workstation) that requests access to the LAN and switch services and responds to requests from the switch. The workstation must be running 802.1x-compliant client software such as that offered in Microsoft Windows operating systems. (The client is the *supplicant* in the 802.1x standard.)
- *Authentication server*—performs the actual authentication of the client. The authentication server validates the identity of the client and notifies the switch whether or not the client is authorized to access the LAN and switch services. Because the switch acts as the proxy, the authentication service is transparent to the client. In this release, the RADIUS security system with Extensible Authentication Protocol (EAP) extensions is the only supported authentication server. It is available in Cisco Secure Access Control Server Version 3.0 or later. RADIUS operates in a client/server model in which secure authentication information is exchanged between the RADIUS server and one or more RADIUS clients.
- *Switch* (edge switch or wireless access point)—controls the physical access to the network based on the authentication status of the client. The switch acts as an intermediary (proxy) between the client and the authentication server, requesting identity information from the client, verifying that information with the authentication server, and relaying a response to the client. The switch includes the RADIUS client, which is responsible for encapsulating and decapsulating the EAP frames and interacting with the authentication server. (The switch is the *authenticator* in the 802.1x standard.)

  When the switch receives EAPOL frames and relays them to the authentication server, the Ethernet header is stripped, and the remaining EAP frame is re-encapsulated in the RADIUS format. The EAP frames are not modified during encapsulation, and the authentication server must support EAP within the native frame format. When the switch receives frames from the authentication server, the server's frame header is removed, leaving the EAP frame, which is then encapsulated for Ethernet and sent to the client.

  The devices that can act as intermediaries include the Catalyst 3850-X, Catalyst 3750-X, Catalyst 3750-E, Catalyst 3750, Catalyst 3650-X, Catalyst 3560-E, Catalyst 3560, Catalyst 3550, Catalyst 2970, Catalyst 2960, Catalyst 2955, Catalyst 2950, Catalyst 2940 switches, or a wireless access point. These devices must be running software that supports the RADIUS client and IEEE 802.1x authentication.

Restrictions for 802.1x Port-Based Authentication

This section lists the restrictions for 802.1x Port-Based Authentication.

Information About 802.1x Port-Based Authentication

Switches also support Cisco TrustSec Security Group Tag (SGT) Exchange Protocol (SxP). The SXP control protocol allows carrying the SGT information between access-layer devices at the Cisco TrustSec domain edge, and distribution layer devices within the Cisco TrustSec domain when the access-layer devices do not have the hardware capability to tag the packets. These switches operate as access layer switches in the Cisco TrustSec network.

For more information about Cisco TrustSec, see the "Cisco TrustSec Switch Configuration Guide" at this URL: http://www.cisco.com/en/US/docs/switches/lan/trustsec/configuration/guide/trustsec.html

The sections on SXP define the capabilities supported on the switch.

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.

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.

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 12.4* and the command reference for this release.

Port-Based Authentication Process

When 802.1x port-based authentication is enabled and the client supports 802.1x-compliant client software, these events occur:

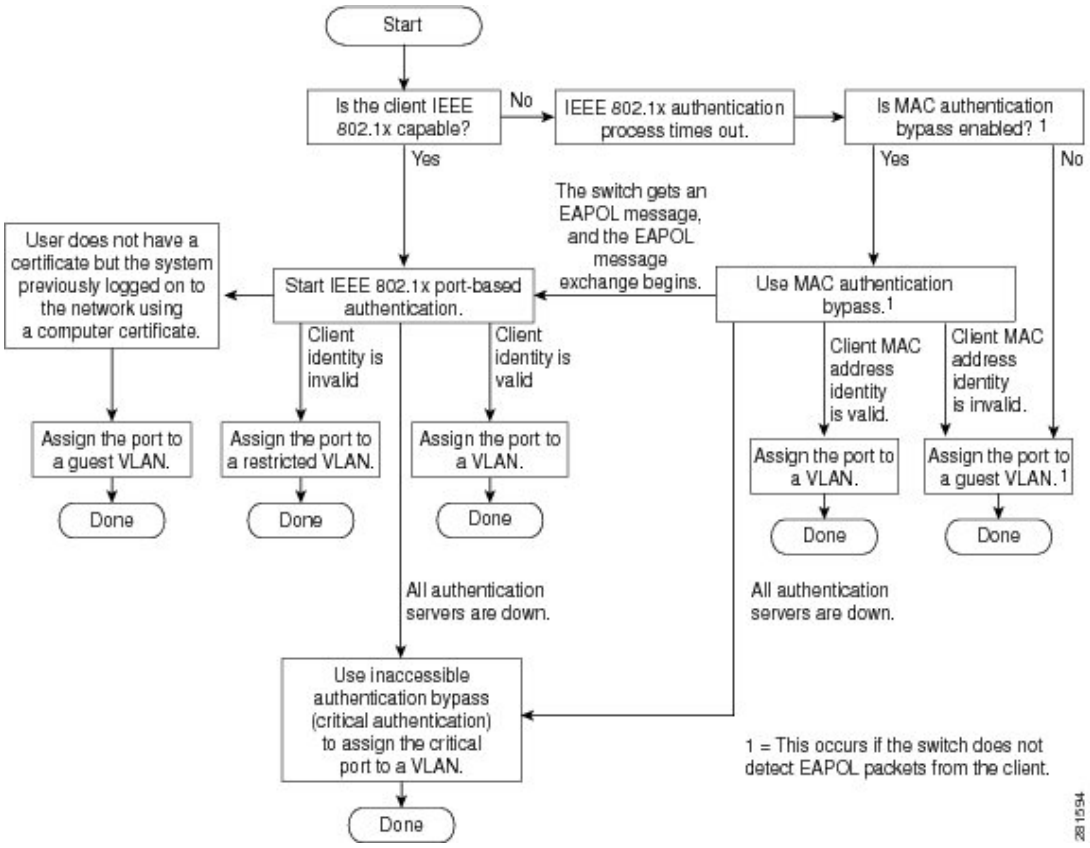
- If the client identity is valid and the 802.1x authentication succeeds, the switch grants the client access to the network.
- If 802.1x authentication times out while waiting for an EAPOL message exchange and MAC authentication bypass is enabled, the switch can use the client MAC address for authorization. If the client MAC address is valid and the authorization succeeds, the switch grants the client access to the network. If the client MAC address is invalid and the authorization fails, the switch assigns the client to a guest VLAN that provides limited services if a guest VLAN is configured.
- If the switch gets an invalid identity from an 802.1x-capable client and a restricted VLAN is specified, the switch can assign the client to a restricted VLAN that provides limited services.
- If the RADIUS authentication server is unavailable (down) and inaccessible authentication bypass is enabled, the switch grants the client access to the network by putting the port in the critical-authentication state in the RADIUS-configured or the user-specified access VLAN.

Note Inaccessible authentication bypass is also referred to as critical authentication or the AAA fail policy.

If Multi Domain Authentication (MDA) is enabled on a port, this flow can be used with some exceptions that are applicable to voice authorization.

This figure shows the authentication process.

Figure 29: Authentication Flowchart



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.

- 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.

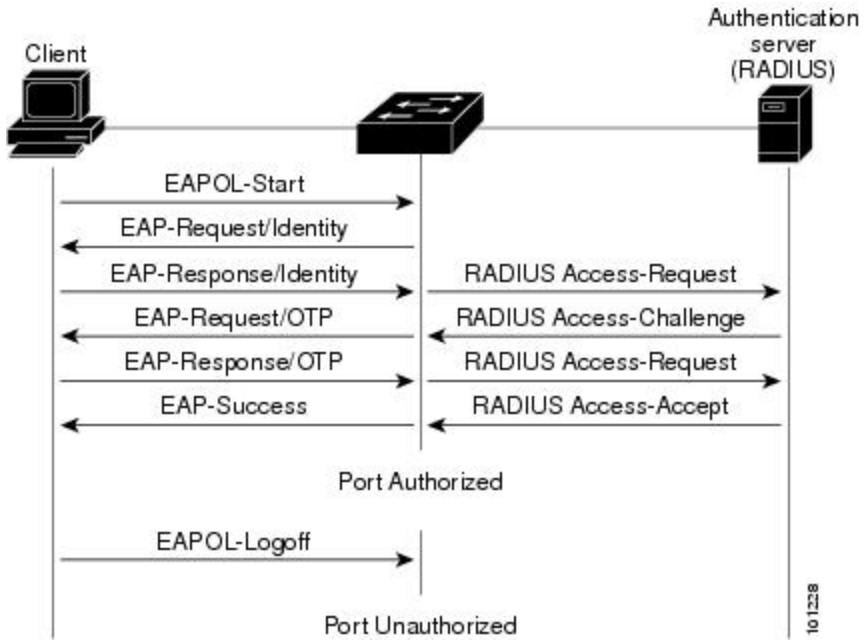
Note If 802.1x authentication is not enabled or supported on the network access device, any EAPOL frames from the client are dropped. If the client does not receive an EAP-request/identity frame after three attempts to start authentication, the client sends frames as if the port is in the authorized 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.

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.

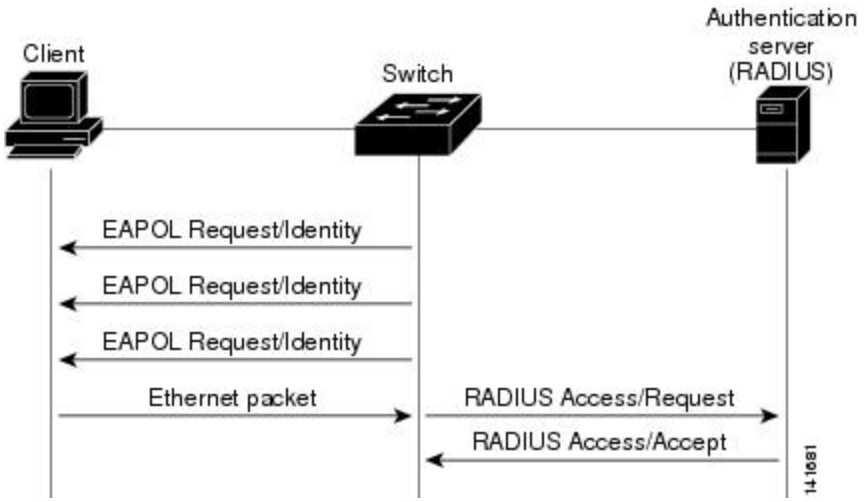
Figure 30: Message Exchange



If 802.1x authentication times out while waiting for an EAPOL message exchange and MAC authentication bypass is enabled, the switch can authorize the client when the switch detects an Ethernet packet from the client. The switch uses the MAC address of the client as its identity and includes this information in the RADIUS-access/request frame that is sent to the RADIUS server. After the server sends the switch the RADIUS-access/accept frame (authorization is successful), the port becomes authorized. If authorization fails and a guest VLAN is specified, the switch assigns the port to the guest VLAN. If the switch detects an EAPOL packet while waiting for an Ethernet packet, the switch stops the MAC authentication bypass process and starts 802.1x authentication.

This figure shows the message exchange during MAC authentication bypass.

Figure 31: Message Exchange During MAC Authentication Bypass



Authentication Manager for Port-Based Authentication

In Cisco IOS Release 12.2(46)SE and earlier, you could not use the same authorization methods, including CLI commands and messages, on this switch and also on other network devices, such as a Catalyst 6000. You had to use separate authentication configurations. Cisco IOS Release 12.2(50)SE and later supports the same authorization methods on all Catalyst switches in a network.

Cisco IOS Release 12.2(55)SE supports filtering verbose system messages from the authentication manager.

Port-Based Authentication Methods

Table 88: 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[12] 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

Authentication method	Mode			
	Single host	Multiple host	MDA	Multiple Authentication
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[13]	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

12 Supported in Cisco IOS Release 12.2(50)SE and later.
13 For clients that do not support 802.1x authentication.

Per-User ACLs and Filter-Ids

ACLs configured on the switch are compatible with other devices running Cisco IOS releases.

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**.)

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.

Beginning with Cisco IOS Release 12.2(55)SE, you can filter out verbose system messages generated by the authentication manager. The filtered content typically relates to authentication success. You can also filter verbose messages for 802.1x authentication and MAB authentication. There is a separate command for each authentication method:

- The **no authentication logging verbose** global configuration command filters verbose messages from the authentication manager.
- The **no dot1x logging verbose** global configuration command filters 802.1x authentication verbose messages.
- The **no mab logging verbose** global configuration command filters MAC authentication bypass (MAB) verbose messages

Table 89: 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.

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 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.
show authentication	**show dot1x**	Display 802.1x statistics, administrative status, and operational status for the switch or for the specified port. authentication manager: compatibility with earlier 802.1x CLI commands

Ports in Authorized and Unauthorized States

During 802.1x authentication, depending on the switch port state, the switch can grant a client access to the network. The port starts in the *unauthorized* state. While in this state, the port that is not configured as a voice VLAN port disallows all ingress and egress traffic except for 802.1x authentication, CDP, and STP packets. When a client is successfully authenticated, the port changes to the *authorized* state, allowing all traffic for the client to flow normally. If the port is configured as a voice VLAN port, the port allows VoIP traffic and 802.1x protocol packets before the client is successfully authenticated.

If a client that does not support 802.1x authentication connects to an unauthorized 802.1x port, the switch requests the client's identity. In this situation, the client does not respond to the request, the port remains in the unauthorized state, and the client is not granted access to the network.

In contrast, when an 802.1x-enabled client connects to a port that is not running the 802.1x standard, the client initiates the authentication process by sending the EAPOL-start frame. When no response is received, the client sends the request for a fixed number of times. Because no response is received, the client begins sending frames as if the port is in the authorized state.

You control the port authorization state by using the **authentication port-control** interface configuration command and these keywords:

- **force-authorized—**disables 802.1x authentication and causes the port to change to the authorized state without any authentication exchange required. The port sends and receives normal traffic without 802.1x-based authentication of the client. This is the default setting.
- **force-unauthorized—**causes the port to remain in the unauthorized state, ignoring all attempts by the client to authenticate. The switch cannot provide authentication services to the client through the port.
- **auto—**enables 802.1x authentication and causes the port to begin in the unauthorized state, allowing only EAPOL frames to be sent and received through the port. The authentication process begins when the link state of the port changes from down to up or when an EAPOL-start frame is received. The switch requests the identity of the client and begins relaying authentication messages between the client and the authentication server. Each client attempting to access the network is uniquely identified by the switch by using the client MAC address.

Note In Session Aware Networking mode, the **authentication port-control** command is **access-session port-control**.

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 master is removed from the switch stack. Note that if the stack master fails, a stack member becomes the new stack master 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 **authentication periodic** 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 master and another to a stack member, and if the stack master fails, the switch stack still has connectivity to the RADIUS server.

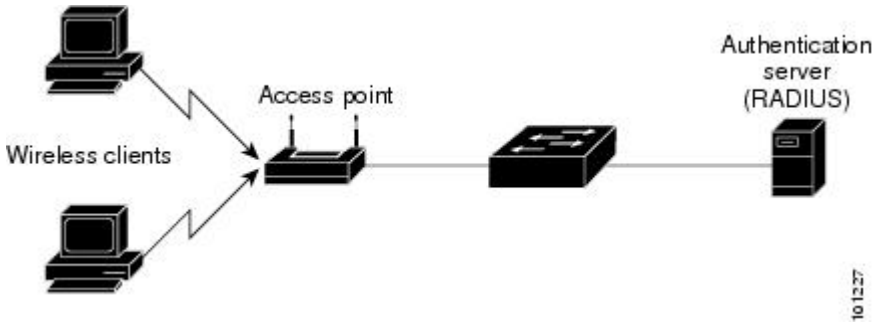
802.1x Host Mode

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.

This figure shows 802.1x port-based authentication in a wireless LAN.

Figure 32: Multiple Host Mode Example



802.1x Multiple Authentication Mode

Multiple-authentication (multiauth) mode allows one client on the voice VLAN and multiple authenticated clients on the data VLAN. When a hub or access point is connected to an 802.1x-enabled port, multiple-authentication mode provides enhanced security over multiple-hosts mode by requiring authentication of each connected client. For non-802.1x devices, you can use MAC authentication bypass or web authentication as the fallback method for individual host authentications to authenticate different hosts through by different methods on a single port.

Multiple-authentication mode also supports MDA functionality on the voice VLAN by assigning authenticated devices to either a data or voice VLAN, depending on the VSAs received from the authentication server.

Note Guest VLAN and authentication-failed VLAN features are supported for ports configured in multiple-authentication mode.

Beginning with Cisco IOS Release 12.2(55)SE, you can assign a RADIUS-server-supplied VLAN in multi-auth mode, under these conditions:

- Only one voice VLAN assignment is supported on a multi-auth port.
- 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.

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

Beginning with Cisco IOS Release 12.2(55)SE, 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 90: 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 Number	AV Pair Name	START	INTERIM	STOP
Attribute[8]	Framed-IP-Address	Never	Sometimes[14]	Sometimes
Attribute[25]	Class	Always	Always	Always
Attribute[30]	Called-Station-ID	Always	Always	Always
Attribute[31]	Calling-Station-ID	Always	Always	Always
Attribute[40]	Acct-Status-Type	Always	Always	Always
Attribute[41]	Acct-Delay-Time	Always	Always	Always
Attribute[42]	Acct-Input-Octets	Never	Always	Always
Attribute[43]	Acct-Output-Octets	Never	Always	Always
Attribute[44]	Acct-Session-ID	Always	Always	Always
Attribute[45]	Acct-Authentic	Always	Always	Always
Attribute[46]	Acct-Session-Time	Never	Always	Always
Attribute[49]	Acct-Terminate-Cause	Never	Never	Always
Attribute[61]	NAS-Port-Type	Always	Always	Always

14 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.

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.

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 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.

- 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.

Related Topics

Configuring 802.1x Readiness Check, on page 754

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.

Related Topics

Configuring the Switch-to-RADIUS-Server Communication, on page 762

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. 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.

When the port is in the force authorized, force unauthorized, unauthorized, or shutdown state, it is put into the configured access 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, 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.

The 802.1x authentication with VLAN assignment feature is not supported on trunk ports, dynamic ports, or with dynamic-access port assignment through a VLAN Membership Policy Server (VMPS).

To configure VLAN assignment you need to perform these tasks:

- Enable AAA authorization by using the **network** keyword to allow interface configuration from the RADIUS server.
- Enable 802.1x authentication. (The VLAN assignment feature is automatically enabled when you configure 802.1x authentication on an access port).
- Assign vendor-specific tunnel attributes in the RADIUS server. The RADIUS server must return these attributes to the switch:
  - [64] Tunnel-Type = VLAN
  - [65] Tunnel-Medium-Type = 802
  - [81] Tunnel-Private-Group-ID = VLAN name or VLAN ID

  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).

Only one 802.1x-authenticated user is supported on a port. If the multiple-hosts mode is enabled on the port, the per-user ACL attribute is disabled for the associated port.

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

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.

Beginning with Cisco IOS Release 12.2(55)SE, 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 by using the **ip access-list extended auth-default-acl** global configuration command.

Note The auth-default-ACL does not support Cisco Discovery Protocol (CDP) bypass in the single host mode. You must configure a static ACL on the interface to support CDP bypass.

The 802.1x and MAB authentication methods support two authentication modes, *open* and *closed*. If there is no static ACL on a port in *closed* authentication mode:

- An auth-default-ACL is created.
- The auth-default-ACL allows only DHCP traffic until policies are enforced.
- When the first host authenticates, the authorization policy is applied without IP address insertion.
- When a second host is detected, the policies for the first host are refreshed, and policies for the first and subsequent sessions are enforced with IP address insertion.

If there is no static ACL on a port in *open* authentication mode:

- An auth-default-ACL-OPEN is created and allows all traffic.
- Policies are enforced with IP address insertion to prevent security breaches.
- Web authentication is subject to the auth-default-ACL-OPEN.

To control access for hosts with no authorization policy, you can configure a directive. The supported values for the directive are *open* and *default*. When you configure the *open* directive, all traffic is allowed. The *default* directive subjects traffic to the access provided by the port. You can configure the directive either in the user profile on the AAA server or on the switch. To configure the directive on the AAA server, use the **authz-directive =<open/default>** global command. To configure the directive on the switch, use the **epm access-control open** global configuration command.

Note The default value of the directive is *default*.

If a host falls back to web authentication on a port without a configured ACL:

- If the port is in open authentication mode, the auth-default-ACL-OPEN is created.

- If the port is in closed authentication mode, the auth-default-ACL is created.

The access control entries (ACEs) in the fallback ACL are converted to per-user entries. If the configured fallback profile does not include a fallback ACL, the host is subject to the auth-default-ACL associated with the port.

Note If you use a custom logo with web authentication and it is stored on an external server, the port ACL must allow access to the external server before authentication. You must either configure a static port ACL or change the auth-default-ACL to provide appropriate access to the external server.

Cisco Secure ACS and Attribute-Value Pairs for the Redirect URL

The switch uses these *cisco-av-pair* VSAs:

- url-redirect is the HTTP 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.

- 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.

Use a restricted VLAN to allow clients that failed authentication access to the network by entering the **dot1x auth-fail vlan** *vlan-id* interface configuration command.

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.

When the switch port is moved to the guest VLAN, the number of allowed 802.1x-incapable hosts is determined by the configured host-mode. 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.

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 master checks the status of the RADIUS servers by sending keepalive packets. When the status of a RADIUS server changes, the stack master 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 stack master 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 master sends the member the server status.

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.

Note If an IP phone and PC are connected to a switchport, and the port is configured in single- or multi-host mode, we do not recommend configuring that port in standalone MAC authentication bypass mode. We recommend only using MAC authentication bypass as a fallback method to 802.1x authentication with the timeout period set to the default of five seconds.

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 Catalyst 3850 series switch port, you can configure an access port VLAN that is also a voice VLAN.

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.

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—MAC authentication bypass and IEEE 802.1x authentication are configured independently 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 802.1x 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 admission control (NAC) Layer 2 IP validation—This feature takes effect after an IEEE 802.1x port is authenticated with MAC authentication bypass, including hosts in the exception list.
- Network Edge Access Topology (NEAT)—MAB and NEAT are mutually exclusive. You cannot enable MAB when NEAT is enabled on an interface, and you cannot enable NEAT when MAB is enabled on an interface.

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.
- 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.

Related Topics

Configuring Flexible Authentication Ordering, on page 800

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.

In Session Aware Networking mode, to enable open authentication, use **no access-session closed**. To disable open authentication, use **access-session closed**.

Related Topics

Configuring Open1x, on page 801

Multidomain Authentication

The switch supports multidomain authentication (MDA), which allows both a data device and voice device, such as an IP phone (Cisco or non-Cisco), to authenticate on the same switch port. The port is divided into a data domain and a voice domain.

MDA does not enforce the order of device authentication. However, for best results, we recommend that a voice device is authenticated before a data device on an MDA-enabled port.

Follow these guidelines for configuring MDA:

- 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.

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.

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.

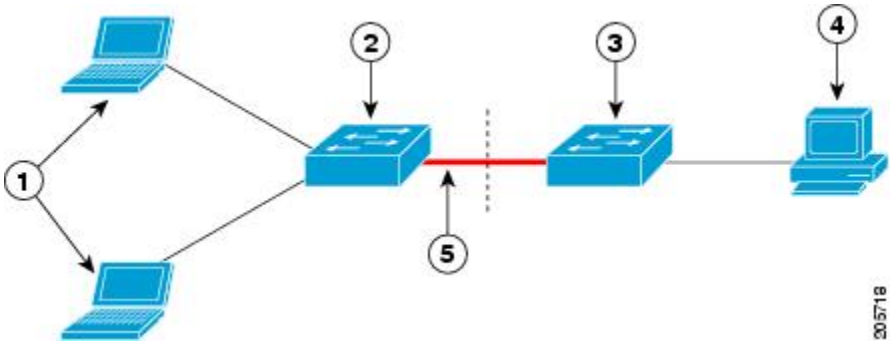
- 802.1x switch supplicant: You can configure a switch to act as a supplicant to another switch by using the 802.1x supplicant feature. This configuration is helpful in a scenario, where, for example, a switch is outside a wiring closet and is connected to an upstream switch through a trunk port. A switch 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.
- If the access VLAN is configured on the authenticator switch, it becomes the native VLAN for the trunk port after successful authentication.

You can enable MDA or multiauth mode on the authenticator switch interface that connects to one more supplicant switches. Multihost mode is not supported on the authenticator switch interface.

Use the **dot1x supplicant force-multicast** global configuration command on the supplicant switch for Network Edge Access Topology (NEAT) to work in all host modes.

- Host Authorization: Ensures that only traffic from authorized hosts (connecting to the switch with supplicant) is allowed on the network. The switches use Client Information Signalling Protocol (CISP) to send the MAC addresses connecting to the supplicant switch to the authenticator switch.
- 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 33: 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		

Voice Aware 802.1x Security

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.

Related Topics

Configuring Voice Aware 802.1x Security, on page 755

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:

```
Controller# 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 91: 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.

Feature	Default Setting
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).
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.
Guest VLAN	None specified.
Inaccessible authentication bypass	Disabled.
Restricted VLAN	None specified.
Authenticator (switch) mode	None specified.
MAC authentication bypass	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:
  - Trunk port—If you try to enable 802.1x authentication on a trunk port, an error message appears, and 802.1x authentication is not enabled. If you try to change the mode of an 802.1x-enabled port to trunk, an error message appears, and the port mode is not changed.
  - Dynamic ports—A port in dynamic mode can negotiate with its neighbor to become a trunk port. If you try to enable 802.1x authentication on a dynamic port, an error message appears, and 802.1x authentication is not enabled. If you try to change the mode of an 802.1x-enabled port to dynamic, an error message appears, and the port mode is not changed.
  - Dynamic-access ports—If you try to enable 802.1x authentication on a dynamic-access (VLAN Query Protocol [VQP]) port, an error message appears, and 802.1x authentication is not enabled. If you try to change an 802.1x-enabled port to dynamic VLAN assignment, an error message appears, and the VLAN configuration is not changed.
  - EtherChannel port—Do not configure a port that is an active or a not-yet-active member of an EtherChannel as an 802.1x port. If you try to enable 802.1x authentication on an EtherChannel port, an error message appears, and 802.1x authentication is not enabled.
  - Switched Port Analyzer (SPAN) and Remote SPAN (RSPAN) destination ports—You can enable 802.1x authentication on a port that is a SPAN or RSPAN destination port. However, 802.1x authentication is disabled until the port is removed as a SPAN or RSPAN destination port. You can enable 802.1x authentication on a SPAN or RSPAN source port.
- Before globally enabling 802.1x authentication on a switch by entering the **dot1x system-auth-control** global configuration command, remove the EtherChannel configuration from the interfaces on which 802.1x authentication and EtherChannel are configured.
- If you are using a device running the Cisco Access Control Server (ACS) application for IEEE 802.1x authentication with EAP-Transparent LAN Services (TLS) and EAP-MD5, make sure that the device is running ACS Version 3.2.1 or later.
- 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 Only Catalyst 3750, 3560, and 2960 switches support CDP bypass. The Catalyst 3750-X, 3560-X, 3750-E, and 3560-E switches do not support CDP bypass.

- 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.
- The 802.1x authentication with VLAN assignment feature is not supported on trunk ports, dynamic ports, or with dynamic-access port assignment through a VMPS.
- You can configure 802.1x authentication on a private-VLAN port, but do not configure IEEE 802.1x authentication with port security, a voice VLAN, a guest VLAN, a restricted VLAN, or a per-user ACL on private-VLAN ports.
- You can configure any VLAN except an RSPAN VLAN, private VLAN, or a voice VLAN as an 802.1x guest VLAN. The guest VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.
- After you configure a guest VLAN for an 802.1x port to which a DHCP client is connected, you might need to get a host IP address from a DHCP server. You can change the settings for restarting the 802.1x authentication process on the switch before the DHCP process on the client times out and tries to get a host IP address from the DHCP server. Decrease the settings for the 802.1x authentication process (**authentication timer inactivity** and **authentication timer reauthentication** interface configuration commands). The amount to decrease the settings depends on the connected 802.1x client type.
- When configuring the inaccessible authentication bypass feature, follow these guidelines:
  - The feature is supported on 802.1x port in single-host mode and multihosts mode.
  - If the client is running Windows XP and the port to which the client is connected is in the critical-authentication state, Windows XP might report that the interface is not authenticated.
  - If the Windows XP client is configured for DHCP and has an IP address from the DHCP server, receiving an EAP-Success message on a critical port might not 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.
- You can configure any VLAN except an RSPAN VLAN or a voice VLAN as an 802.1x restricted VLAN. The restricted VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.

MAC Authentication Bypass

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.

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

Beginning in privileged EXEC mode, follow these steps to enable the 802.1x readiness check on the switch:

SUMMARY STEPS

1. **dot1x test eapol-capable** [**interface** *interface-id*]
2. **configure terminal**
3. **dot1x test timeout** *timeout*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**dot1x test eapol-capable** [**interface** *interface-id*] **Example:** `Controller# dot1x test eapol-capable interface gigabitethernet1/0/13`	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.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Controller# configure terminal`	(Optional) Enters global configuration mode.
Step 3	**dot1x test timeout** *timeout* **Example:** `Controller(config)# dot1x test timeout 300`	(Optional) Configures the timeout used to wait for EAPOL response. The range is from 1 to 65535 seconds. The default is 10 seconds.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Related Topics

802.1x Readiness Check, on page 733

Configuring Voice Aware 802.1x Security

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**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**errdisable detect cause security-violation shutdown vlan** **Example:** `Controller(config)# errdisable detect cause security-violation shutdown vlan`	Shuts down any VLAN on which a security violation error occurs. **Note** If the **shutdown vlan** keywords are not included, the entire port enters the error-disabled state and shuts down.
Step 3	**errdisable recovery cause security-violation** **Example:** `Controller(config)# errdisable recovery cause security-violation`	(Optional) Enables automatic per-VLAN error recovery.
Step 4	**clear errdisable interface** *interface-id* **vlan** [*vlan-list*] **Example:** `Controller(config)# clear errdisable interface GigabitEthernet4/0/2 vlan`	(Optional) Reenables individual VLANs that have been error disabled. • For *interface-id*, specify the port on which to reenable individual VLANs. • (Optional) For *vlan-list*, specify a list of VLANs to be re-enabled. If *vlan-list* is not specified, all VLANs are re-enabled.

	Command or Action	Purpose

Step 5	Enter the following:
	• **shutdown**
	• **no shutdown**
	
	Example:
	`Controller(config-if)# shutdown`
	`Controller(config-if)# no shutdown`
		(Optional) Re-enables an error-disabled VLAN, and clear all error-disable indications.

Step 6	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.

Related Topics

Voice Aware 802.1x Security, on page 749

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:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**aaa new-model** **Example:** `Controller(config)# aaa new-model`	Enables AAA.
Step 3	**aaa authentication dot1x** {**default**} *method1* **Example:** `Controller(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:** `Controller(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:** `Controller(config-if)# switchport mode access`	Sets the port to access mode.
Step 6	**authentication violation** {**shutdown** | **restrict** | **protect** | **replace**} **Example:** `Controller(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:** `Controller(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.	

	Command or Action	Purpose
Step 7	The user disconnects from the port.	
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:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**aaa new-model** **Example:** `Controller(config)# aaa new-model`	Enables AAA.
Step 3	**aaa authentication dot1x** {**default**} *method1*	Creates an 802.1x authentication method list.

	Command or Action	Purpose
	Example: `Controller(config)# aaa authentication dot1x default group radius`	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	**dot1x system-auth-control** **Example:** `Controller(config)# dot1x system-auth-control`	Enables 802.1x authentication globally on the switch.
Step 5	**aaa authorization network** {**default**} **group radius** **Example:** `Controller(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:** `Controller(config)# radius-server host 124.2.2.12`	(Optional) Specifies the IP address of the RADIUS server.
Step 7	**radius-server key** *string* **Example:** `Controller(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:** `Controller(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.

	Command or Action	Purpose
Step 9	**switchport mode access** **Example:** `Controller(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:** `Controller(config-if)# authentication port-control auto`	Enables 802.1x authentication on the port.
Step 11	**dot1x pae authenticator** **Example:** `Controller(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:** `Controller(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.

Beginning in privileged EXEC mode, 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. **configure terminal**
2. **radius-server host** {*hostname* | *ip-address*} **auth-port** *port-number* **key** *string*
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**radius-server host** {*hostname* | *ip-address*} **auth-port** *port-number* **key** *string* **Example:** `Controller(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 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Related Topics

Switch-to-RADIUS-Server Communication, on page 734

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:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(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:** `Controller(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 one client on the voice VLAN and multiple authenticated clients on the 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.

	Command or Action	Purpose
Step 4	**end** **Example:** `Controller(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:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet2/0/1`	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**authentication periodic** **Example:** `Controller(config-if)# authentication`	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.

	Command or Action	Purpose
	`periodic`	
Step 4	**authentication timer** {{[**inactivity** | **reauthenticate** | **restart**]} {*value*}} **Example:** `Controller(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 • **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:** `Controller(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 inactivity** 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 inactivity** *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:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet2/0/1`	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**authentication timer inactivity** *seconds* **Example:** `Controller(config-if)# authentication timer inactivity 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:** `Controller(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show authentication sessions interface** *interface-id* **Example:** `Controller# show authentication sessions interface gigabitethernet2/0/1`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Controller# 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:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet2/0/1`	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**authentication timer reauthenticate** *seconds* **Example:** `Controller(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:** `Controller(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show authentication sessions interface** *interface-id* **Example:** `Controller# show authentication sessions interface`	Verifies your entries.

	Command or Action	Purpose
	`gigabitethernet2/0/1`	
Step 6	**copy running-config startup-config** **Example:** `Controller# 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:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface`	Specifies the port to be configured, and enter interface configuration mode.

	Command or Action	Purpose
	`gigabitethernet2/0/1`	
Step 3	**dot1x max-reauth-req** *count* **Example:** `Controller(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:** `Controller(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:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller# interface gigabitethernet2/0/1`	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**switchport mode access** **Example:** `Controller(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:** `Controller(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:** `Controller(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:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**authentication mac-move permit** **Example:** `Controller(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:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 4	**show running-config** **Example:** `Controller# show running-config`	Verifies your entries.
Step 5	**copy running-config startup-config** **Example:** `Controller# 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:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(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:** `Controller(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:** `Controller(config-if)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	**show running-config** **Example:** `Controller# show running-config`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Controller# 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:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(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:** `Controller(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:** `Controller(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:** `Controller(config-if)# end`	Returns to privileged EXEc mode.

	Command or Action	Purpose
Step 6	**show running-config** **Example:** `Controller# show running-config`	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** `Controller# 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:** `Controller# configure terminal`	Enters the global configuration mode.

	Command or Action	Purpose
Step 2	**interface** *interface-id* **Example:** `Controller(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:** `Controller(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:** `Controller(config-if)# authentication event no-response action authorize vlan 2`	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, a primary private VLAN, or a voice VLAN as an 802.1x guest VLAN.
Step 5	**end** **Example:** `Controller(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:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(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:** `Controller(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:** `Controller(config-if)# authentication port-control auto`	Enables 802.1x authentication on the port.

	Command or Action	Purpose
Step 5	**authentication event fail action authorize vlan** *vlan-id* **Example:** `Controller(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, a primary private VLAN, or a voice VLAN as an 802.1x restricted VLAN.
Step 6	**end** **Example:** `Controller(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:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(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 `Controller(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:** `Controller(config-if)# authentication port-control auto`	Enables 802.1x authentication on the port.
Step 5	**authentication event fail action authorize vlan** *vlan-id* **Example:** `Controller(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, a primary private VLAN, or a voice VLAN as an 802.1x restricted VLAN.
Step 6	**authentication event retry** *retry count* **Example:** `Controller(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.

	Command or Action	Purpose
Step 7	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.

Configuring the Inaccessible Authentication Bypass Feature

You can configure the inaccessible bypass feature, also referred to as critical authentication or the AAA fail policy.

Beginning in privileged EXEC mode, follow these steps to configure the port as a critical port and enable the inaccessible authentication bypass feature. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **radius-server dead-criteria time** *time* **tries** *tries*
3. **radius-server deadtime** *minutes*
4. **radius-server host** *ip-address* [**acct-port** *udp-port*] [**auth-port** *udp-port*][**test username** *name* [**idle-time** *time*] [**ignore-acct-port**] [**ignore-auth-port**]] [**key** *string*]
5. **dot1x critical** {**eapol** | **recovery delay** *milliseconds*}
6. **interface** *interface-id*
7. **authentication event server dead action** {**authorize** | **reinitialize**} **vlan** *vlan-id*]
8. **dot1x critical** [**recovery action reinitialize** | **vlan** *vlan-id*]
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**radius-server dead-criteria time** *time* **tries** *tries* **Example:** `Controller(config)# radius-server`	(Optional) Sets the conditions that are used to decide when a RADIUS server is considered unavailable or *dead*. The range for *time* is from 1 to 120 seconds. The switch dynamically determines the default *seconds* value that is 10 to 60 seconds.

	Command or Action	Purpose
	`dead-criteria time 30 tries 20`	The range for *tries* is from 1 to 100. The switch dynamically determines the default *tries* parameter that is 10 to 100.
Step 3	**radius-server deadtime** *minutes* **Example:** `Controller(config)# radius-server deadtime 60`	(Optional) Sets the number of minutes that a RADIUS server is not sent requests. The range is from 0 to 1440 minutes (24 hours). The default is 0 minutes.
Step 4	**radius-server host** *ip-address* [**acct-port** *udp-port*] [**auth-port** *udp-port*][**test username** *name* [**idle-time** *time*] [**ignore-acct-port**] [**ignore-auth-port**]] [**key** *string*] **Example:** `Controller(config)# radius-server host 1.1.1.2 acct-port 1550 auth-port 1560 test username user1 idle-time 30 key abc1234`	(Optional) Configures the RADIUS server parameters by using these keywords: • **acct-port** *udp-port*—Specifies the UDP port for the RADIUS accounting server. The range for the UDP port number is from 0 to 65536. The default is 1646. • **auth-port** *udp-port*—Specifies the UDP port for the RADIUS authentication server. The range for the UDP port number is from 0 to 65536. The default is 1645. **Note** You should configure the UDP port for the RADIUS accounting server and the UDP port for the RADIUS authentication server to nondefault values. • **test username** *name*—Enables automated testing of the RADIUS server status, and specify the username to be used. • **idle-time** *time*—Sets the interval of time in minutes after which the switch sends test packets to the server. The range is from 1 to 35791 minutes. The default is 60 minutes (1 hour). • **ignore-acct-port**—Disables testing on the RADIUS-server accounting port. • **ignore-auth-port**—Disables testing on the RADIUS-server authentication port. • 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. You can also configure the authentication and encryption key by using the **radius-server key** {**0** *string*| **7** *string*	*string*} global configuration command.

	Command or Action	Purpose
Step 5	**dot1x critical** {**eapol** | **recovery delay** *milliseconds*} **Example:** `Controller(config)# dot1x critical eapol recovery delay 2000`	(Optional) Configures the parameters for inaccessible authentication bypass: **eapol—**Specifies that the switch sends an EAPOL-Success message when the switch successfully authenticates the critical port. **recovery delay** *milliseconds***—**Sets the recovery delay period during which the switch waits to 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 6	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet 1/0/1`	Specify the port to be configured, and enter interface configuration mode.
Step 7	**authentication event server dead action** {**authorize** | **reinitialize**} **vlan** *vlan-id*] **Example:** `Controller(config-if)# authentication event server dead action reinitialize vlan 5`	Moves hosts on the port if the RADIUS server is unreachable: • **authorize–**Moves any new hosts trying to authenticate to the user-specified critical VLAN. • **reinitialize–**Moves all authorized hosts on the port to the user-specified critical VLAN.
Step 8	**dot1x critical** [**recovery action reinitialize** | **vlan** *vlan-id*] **Example:** `Controller(config-if)# dot1x critical recovery action reinitialize`	Enables the inaccessible authentication bypass feature, and use these keywords to configure the feature: • **authorize—**Authorizes the port. • **reinitialize—**Reinitializes all authorized clients.
Step 9	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.

Example of Configuring Inaccessible Authentication Bypass

This example shows how to configure the inaccessible authentication bypass feature:

```
Controller(config)# radius-server dead-criteria time 30 tries 20
Controller(config)# radius-server deadtime 60
Controller(config)# radius-server host 1.1.1.2 acct-port 1550 auth-port 1560 test username
 user1 idle-time 30 key abc1234
Controller(config)# dot1x critical eapol
```

```
Controller(config)# dot1x critical recovery delay 2000
Controller(config)# interface gigabitethernet 1/0/1
Controller(config-if)# dot1x critical
Controller(config-if)# dot1x critical recovery action reinitialize
Controller(config-if)# dot1x critical vlan 20
Controller(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:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet2/0/3`	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**authentication control-direction** {**both**	**in**} **Example:** `Controller(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.

	Command or Action	Purpose
Step 4	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.
Step 5	**show authentication sessions interface** *interface-id* **Example:** `Controller# show authentication sessions interface gigabitethernet2/0/3`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Controller# 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:** `Controller# configure terminal`	Enters the global configuration mode.

	Command or Action	Purpose
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet2/0/1`	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**authentication port-control auto** **Example:** `Controller(config-if)# authentication port-control auto`	Enables 802.1x authentication on the port.
Step 4	**mab** [**eap**] **Example:** `Controller(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:** `Controller(config-if)# end`	Returns to privileged EXEC mode.

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:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**vlan group** *vlan-group-name* **vlan-list** *vlan-list* **Example:** `Controller(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:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 4	**no vlan group** *vlan-group-name* **vlan-list** *vlan-list* **Example:** `Controller(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:

```
Controller(config)# vlan group eng-dept vlan-list 10

Controller(config)# show vlan group group-name eng-dept
Group Name                       Vlans Mapped
-------------                    --------------
eng-dept                         10

Controller(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:

```
Controller(config)# vlan group eng-dept vlan-list 30
Controller(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:

```
Controller# 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:

```
Controller(config)# no vlan group eng-dept vlan-list 30
Vlan 30 is successfully cleared from vlan group eng-dept.

Controller(config)# show vlan group group-name eng-dept
```

This example shows how to clear all the VLAN groups:

```
Controller(config)# no vlan group end-dept vlan-list all
Controller(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:** `Controller# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet2/0/3`	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**switchport mode access** **Example:** `Controller(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:** `Controller(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:** `Controller(config-if)# authentication periodic`	Enables periodic re-authentication of the client, which is disabled by default.
Step 6	**authentication timer reauthenticate** **Example:** `Controller(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.
Step 7	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.
Step 8	**show authentication sessions interface** *interface-id* **Example:** `Controller# show authentication sessions interface gigabitethernet2/0/3`	Verifies your entries.

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

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 *cisco-av-pairs* must be configured as *device-traffic-class=switch* on the ACS, which sets the interface as a trunk after the supplicant is successfully authenticated.

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:** `Controller# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**cisp enable** **Example:** `Controller(config)# cisp enable`	Enables CISP.
Step 3	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet2/0/1`	Specifies the port to be configured, and enter interface configuration mode.
Step 4	**switchport mode access** **Example:** `Controller(config-if)# switchport mode access`	Sets the port mode to **access**.
Step 5	**authentication port-control auto** **Example:** `Controller(config-if)# authentication port-control auto`	Sets the port-authentication mode to auto.
Step 6	**dot1x pae authenticator** **Example:** `Controller(config-if)# dot1x pae authenticator`	Configures the interface as a port access entity (PAE) authenticator.
Step 7	**spanning-tree portfast** **Example:** `Controller(config-if)# spanning-tree portfast trunk`	Enables Port Fast on an access port connected to a single workstation or server..
Step 8	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.
Step 9	**show running-config interface** *interface-id* **Example:** `Controller# show running-config interface gigabitethernet2/0/1`	Verifies your configuration.

	Command or Action	Purpose
Step 10	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

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 credentials** *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:** `Controller# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**cisp enable** **Example:** `Controller(config)# cisp enable`	Enables CISP.
Step 3	**dot1x credentials** *profile* **Example:** `Controller(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:** `Controller(config)# username suppswitch`	Creates a username.
Step 5	**password** *password* **Example:** `Controller(config)# password myswitch`	Creates a password for the new username.
Step 6	**dot1x supplicant force-multicast** **Example:** `Controller(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:** `Controller(config)# interface gigabitethernet1/0/1`	Specifies the port to be configured, and enter interface configuration mode.
Step 8	**switchport trunk encapsulation dot1q** **Example:** `Controller(config-if)# switchport trunk encapsulation dot1q`	Sets the port to trunk mode.
Step 9	**switchport mode trunk** **Example:** `Controller(config-if)# switchport mode trunk`	Configures the interface as a VLAN trunk port.

	Command or Action	Purpose
Step 10	**dot1x pae supplicant** **Example:** `Controller(config-if)# dot1x pae supplicant`	Configures the interface as a port access entity (PAE) supplicant.
Step 11	**dot1x credentials** *profile-name* **Example:** `Controller(config-if)# dot1x credentials test`	Attaches the 802.1x credentials profile to the interface.
Step 12	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.
Step 13	**show running-config interface** *interface-id* **Example:** `Controller# show running-config interface gigabitethernet1/0/1`	Verifies your configuration.
Step 14	**copy running-config startup-config** **Example:** `Controller# 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:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**ip device tracking** **Example:** `Controller(config)# ip device tracking`	Sets the ip device tracking table.
Step 3	**aaa new-model** **Example:** `Controller(config)# aaa new-model`	Enables AAA.
Step 4	**aaa authorization network default local group radius** **Example:** `Controller(config)# aaa authorization network default`	Sets the authorization method to local. To remove the authorization method, use the **no aaa authorization network default local group radius** command.

	Command or Action	Purpose
	`local group radius`	
Step 5	**radius-server vsa send authentication** **Example:** `Controller(config)# radius-server vsa send authentication`	Configures the radius vsa send authentication.
Step 6	**interface** *interface-id* **Example:** `Controller(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:** `Controller(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:** `Controller(config-if)# show running-config interface gigabitethernet2/0/4`	Verifies your configuration.
Step 9	**copy running-config startup-config** **Example:** `Controller# 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:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**access-list** *access-list-number* **{ deny | permit } { hostname | any | host } log** **Example:** `Controller(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: • **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.

	Command or Action	Purpose
Step 3	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet2/0/2`	Enters interface configuration mode.
Step 4	**ip access-group** *acl-id* **in** **Example:** `Controller(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:** `Controller(config-if)# exit`	Returns to global configuration mode.
Step 6	**aaa new-model** **Example:** `Controller(config)# aaa new-model`	Enables AAA.
Step 7	**aaa authorization network default group radius** **Example:** `Controller(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:** `Controller(config)# ip device tracking`	Enables the IP device tracking table. To disable the IP device tracking table, use the **no ip device tracking** global configuration commands.
Step 9	**ip device tracking probe** [**count** | **interval** | **use-svi**] **Example:** `Controller(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.

	Command or Action	Purpose
Step 10	**radius-server vsa send authentication** **Example:** `Controller(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:** `Controller(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:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**mab request format attribute 32 vlan access-vlan** **Example:** `Controller(config)# mab request format attribute 32 vlan access-vlan`	Enables VLAN ID-based MAC authentication.

	Command or Action	Purpose
Step 3	**copy running-config startup-config** **Example:** `Controller# 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.

Note Before changing the default order and priority of these authentication methods, however, you should understand the potential consequences of those changes. See http://www.cisco.com/en/US/prod/collateral/iosswrel/ps6537/ps6586/ps6638/application_note_c27-573287_ps6638_Products_White_Paper.html for details.

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:** `Controller# configure terminal`	Enters the global configuration mode.

	Command or Action	Purpose
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet 1/0/1`	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**switchport mode access** **Example:** `Controller(config-if)# switchport mode access`	Sets the port to access mode only if you previously configured the RADIUS server.
Step 4	**authentication order** [**dot1x** | **mab**] | {**webauth**} **Example:** `Controller(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:** `Controller(config-if)# authentication priority mab dot1x`	(Optional) Adds an authentication method to the port-priority list.
Step 6	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.

Related Topics

Flexible Authentication Ordering, on page 746

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:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet 1/0/1`	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**switchport mode access** **Example:** `Controller(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:** `Controller(config-if)# authentication control-direction both`	(Optional) Configures the port control as unidirectional or bidirectional.
Step 5	**authentication fallback** *name* **Example:** `Controller(config-if)# authentication fallback`	(Optional) Configures a port to use web authentication as a fallback method for clients that do not support 802.1x authentication.

	Command or Action	Purpose
	`profile1`	
Step 6	**authentication host-mode** [**multi-auth** | **multi-domain** | **multi-host** | **single-host**] **Example:** `Controller(config-if)# authentication host-mode multi-auth`	(Optional) Sets the authorization manager mode on a port.
Step 7	**authentication open** **Example:** `Controller(config-if)# authentication open`	(Optional) Enables or disable open access on a port.
Step 8	**authentication order** [**dot1x** | **mab**] | {**webauth**} **Example:** `Controller(config-if)# authentication order dot1x webauth`	(Optional) Sets the order of authentication methods used on a port.
Step 9	**authentication periodic** **Example:** `Controller(config-if)# authentication periodic`	(Optional) Enables or disable reauthentication on a port.
Step 10	**authentication port-control** {**auto** | **force-authorized** | **force-un authorized**} **Example:** `Controller(config-if)# authentication port-control auto`	(Optional) Enables manual control of the port authorization state.
Step 11	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.

Related Topics

Open1x Authentication, on page 746

Configuring a Web Authentication Local Banner

Beginning in privileged EXEC mode, follow these steps to configure a local banner on a switch that has web authentication configured.

SUMMARY STEPS

1. **configure terminal**
2. **ip admission auth-proxy-banner http** [*banner-text* | *file-path*]
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ip admission auth-proxy-banner http** [*banner-text*	*file-path*] **Example:** `Controller(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 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

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** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet2/0/1`	Specifies the port to be configured, and enter interface configuration mode.
Step 3	**switchport mode access** **Example:** `Controller(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:** `Controller(config-if)# no dot1x pae authenticator`	Disables 802.1x authentication on the port.
Step 5	**end** **Example:** `Controller(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:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet1/0/2`	Enters interface configuration mode, and specify the port to be configured.
Step 3	**dot1x default** **Example:** `Controller(config-if)# dot1x default`	Resets the 802.1x parameters to the default values.
Step 4	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.

Monitoring 802.1x Statistics and Status

Table 92: 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

Command	Purpose
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 93: 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

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

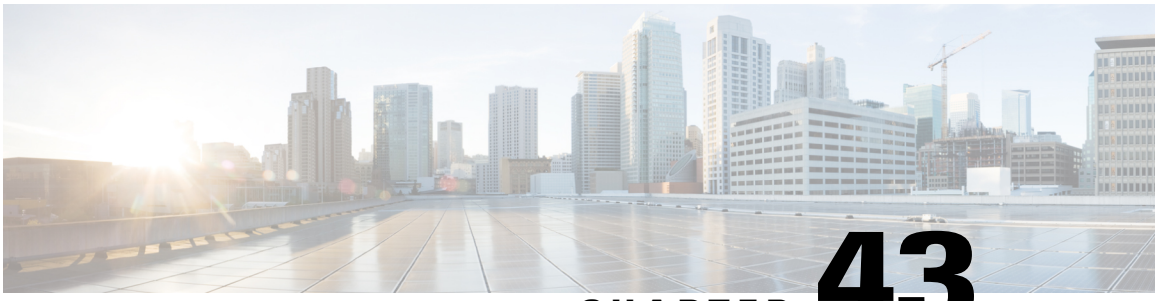
Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, 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 for 802.1x Port-Based Authentication

This table lists the features in this module and provides links to specific configuration information.

Table 94: Feature Information for 802.1x Port-Based Authentication

Feature Name	Releases	Feature Information



CHAPTER 43

Configuring Web-Based Authentication

This chapter describes how to configure web-based authentication on the Catalyst 3850 switch. It contains these sections:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Web-Based Authentication

None.

Restrictions for Web-Based Authentication

Restrictions are found in the Guideline sections.

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.

These sections describe the role of web-based authentication as part of AAA:

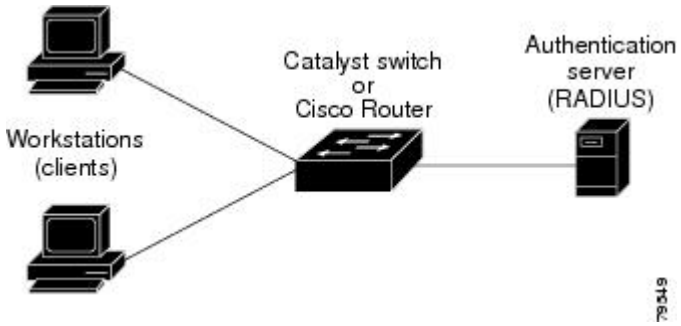
Device Roles

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.

This figure shows the roles of these devices in a network.

Figure 34: Web-Based Authentication Device Roles



Host Detection

The switch maintains an IP device tracking table to store information about detected hosts.

Note By default, the IP device tracking feature is disabled on a switch. You must enable the IP device tracking feature to use web-based authentication.

For Layer 2 interfaces, web-based authentication detects IP hosts by using these mechanisms:

- ARP based trigger—ARP redirect ACL allows web-based authentication to detect hosts with a static IP address or a dynamic IP address.
- Dynamic ARP inspection
- DHCP snooping—Web-based authentication is notified when the switch creates a DHCP-binding entry for the host.

Session Creation

When web-based authentication detects a new host, it creates a session as follows:

- Reviews the exception list.

  If the host IP is included in the exception list, the policy from the exception list entry is applied, and the session is established.

- Reviews for authorization bypass

  If the host IP is not on the exception list, web-based authentication sends a nonresponsive-host (NRH) request to the server.

  If the server response is access accepted, authorization is bypassed for this host. The session is established.

- Sets up the HTTP intercept ACL

  If the server response to the NRH request is access rejected, the HTTP intercept ACL is activated, and the session waits for HTTP traffic from the host.

Authentication Process

When you enable web-based authentication, these events occur:

- The user initiates an HTTP session.
- The HTTP traffic is intercepted, and authorization is initiated. The switch sends the login page to the user. The user enters a username and password, and the switch sends the entries to the authentication server.
- If the authentication succeeds, the switch downloads and activates the user's access policy from the authentication server. The login success page is sent to the user.
- If the authentication fails, the switch sends the login fail page. The user retries the login. If the maximum number of attempts fails, the switch sends the login expired page, and the host is placed in a watch list. After the watch list times out, the user can retry the authentication process.

- If the authentication server does not respond to the switch, and if an AAA fail policy is configured, the switch applies the failure access policy to the host. The login success page is sent to the user.
- 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:

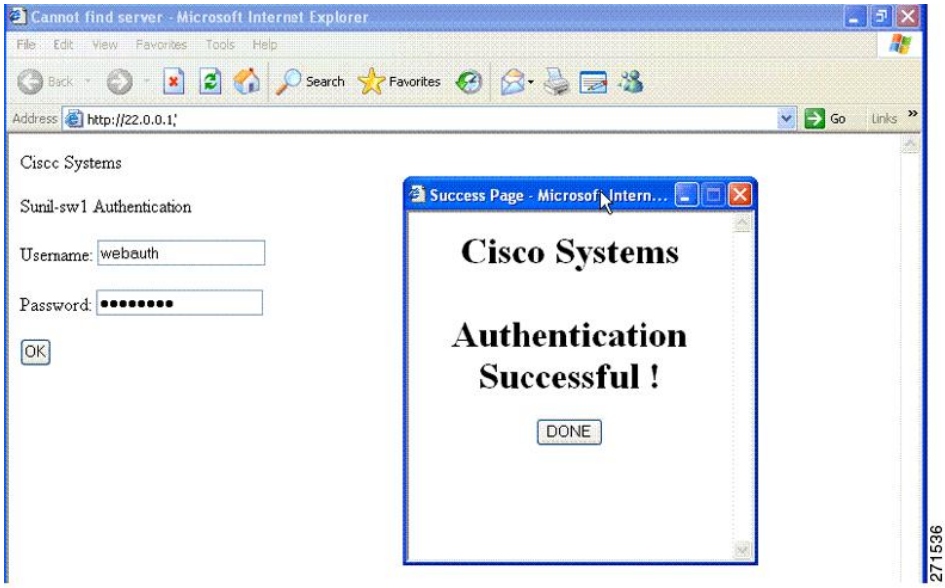
- *Authentication Successful*
- *Authentication Failed*
- *Authentication Expired*

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 35: Authentication Successful Banner

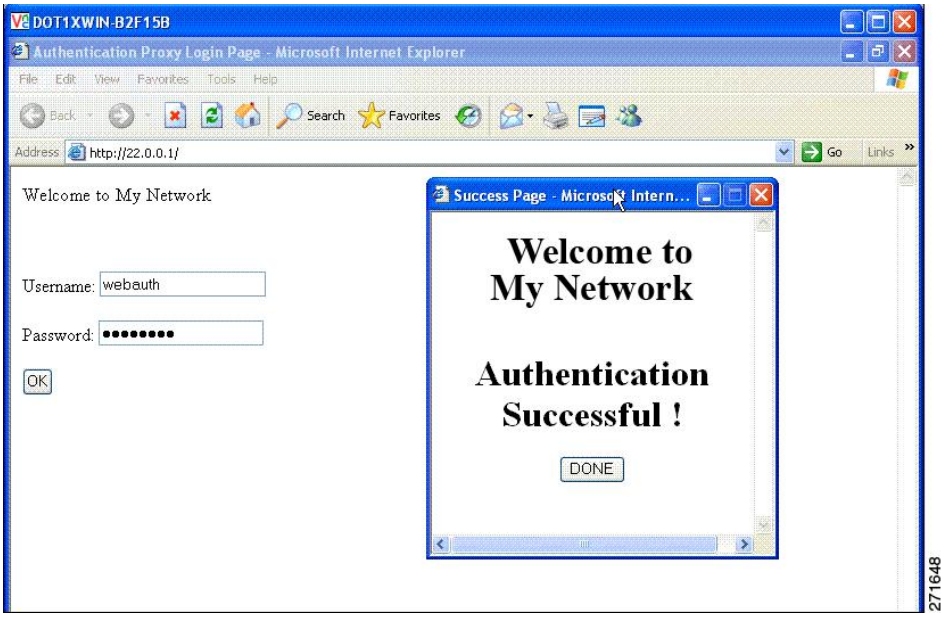


The banner can be customized as follows:

- 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.

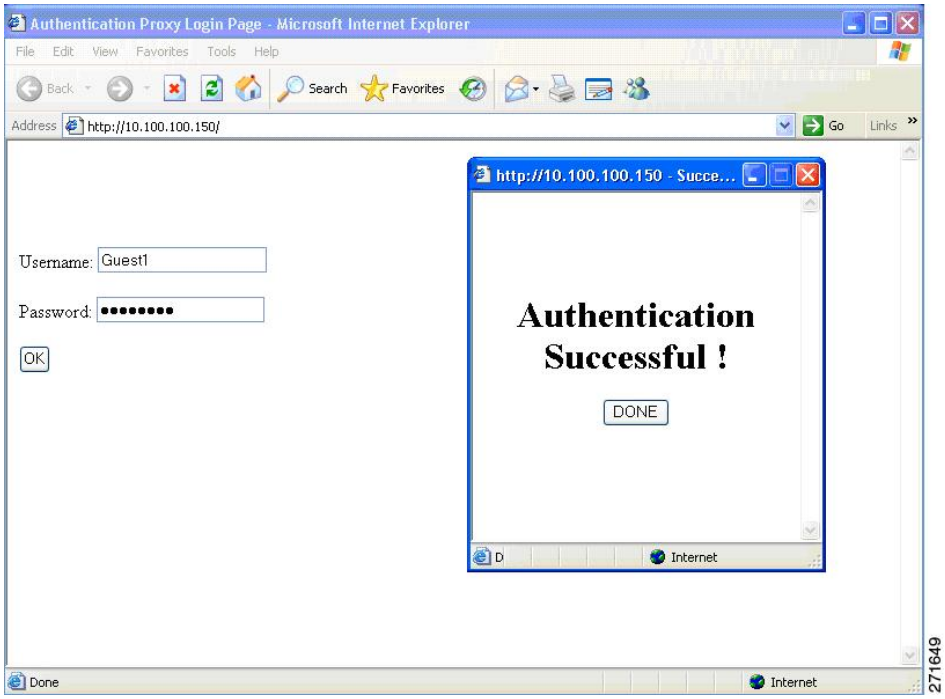
- New-style mode—Use the **parameter-map type webauth global banner** global configuration command

Figure 36: 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 37: 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.

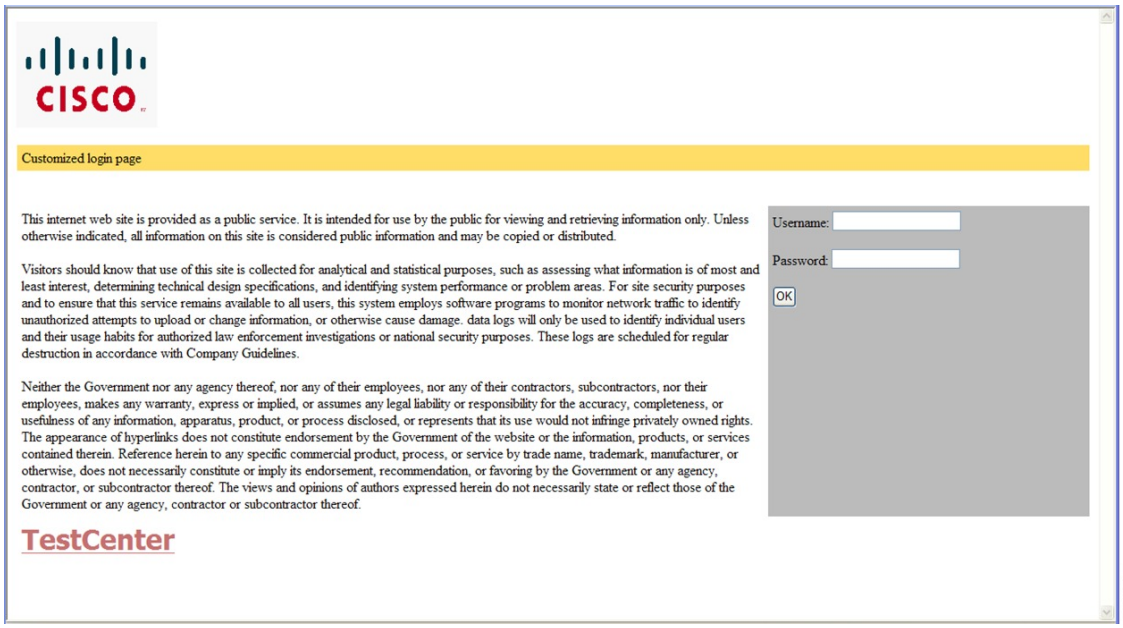
Guidelines

- 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 38: Customizeable 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.
- T o 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 826

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.

Related Topics

Specifying a Redirection URL for Successful Login, on page 828

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.

For more information about enabling port security, see the .

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 95: 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 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)*

This example shows how to verify the configuration:

```
Controller# show ip admission status
IP admission status:
  Enabled interfaces            0
  Total sessions                0
  Init sessions                 0     Max init sessions allowed     100
    Limit reached               0     Hi watermark                  0
  TCP half-open connections     0     Hi watermark                  0
  TCP new connections           0     Hi watermark                  0
  TCP half-open + new           0     Hi watermark                  0
  HTTPD1 Contexts               0     Hi watermark                  0

  Parameter Map: Global
    Custom Pages
      Custom pages not configured
    Banner
      Banner not configured
```

Beginning in privileged EXEC mode, follow these steps to configure the authentication rule and interfaces:

SUMMARY STEPS

1. **configure terminal**
2. **ip admission name** *name* **proxy http**
3. **interface** *type slot/port*
4. **ip access-group** *name*
5. **ip admission** *name*
6. **exit**
7. **ip device tracking**
8. **end**
9. **show ip admission status**
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**ip admission name** *name* **proxy http** **Example:** `Controller(config)# ip admission name webauth1 proxy http`	Configures an authentication rule for web-based authorization.
Step 3	**interface** *type slot/port* **Example:** `Controller(config)# interface gigabitEthernet1/0/1`	Enters interface configuration mode and specifies the ingress Layer 2 or Layer 3 interface to be enabled for web-based authentication. *type* can be fastethernet, gigabit ethernet, or tengigabitethernet.
Step 4	**ip access-group** *name* **Example:** `Controller(config-if)# ip access-group webauthag`	Applies the default ACL.
Step 5	**ip admission** *name* **Example:** `Controller(config-if)# ip admission webauth1`	Configures web-based authentication on the specified interface.

	Command or Action	Purpose
Step 6	**exit** **Example:** `Controller(config-if)# exit`	Returns to configuration mode.
Step 7	**ip device tracking** **Example:** `Controller(config)# ip device tracking`	Enables the IP device tracking table.
Step 8	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 9	**show ip admission status** **Example:** `Controller# show ip admission status`	Displays the configuration.
Step 10	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring AAA Authentication

Beginning in privileged EXEC mode, follow these steps to configure AAA authentication:

SUMMARY STEPS

1. **configure terminal**
2. **aaa new-model**
3. **aaa authentication login default group** {**tacacs+** | **radius**}
4. **aaa authorization auth-proxy default group** {**tacacs+** | **radius**}
5. **tacacs-server host** {*hostname* | *ip_address*}
6. **tacacs-server key** {*key-data*}
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**aaa new-model** **Example:** `Controller(config)# aaa new-model`	Enables AAA functionality.
Step 3	**aaa authentication login default group** {**tacacs+** | **radius**} **Example:** `Controller(config)# aaa authentication login default group tacacs+`	Defines the list of authentication methods at login.
Step 4	**aaa authorization auth-proxy default group** {**tacacs+** | **radius**} **Example:** `Controller(config)# aaa authorization auth-proxy default group tacacs+`	Creates an authorization method list for web-based authorization.
Step 5	**tacacs-server host** {*hostname* | *ip_address*} **Example:** `Controller(config)# tacacs-server host 10.1.1.1`	Specifies an AAA server.
Step 6	**tacacs-server key** {*key-data*} **Example:** `Controller(config)# tacacs-server key`	Configures the authorization and encryption key used between the switch and the TACACS server.
Step 7	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Configuring Switch-to-RADIUS-Server Communication

Beginning in privileged EXEC mode, follow these steps to configure the RADIUS server parameters:

Before You Begin

Identify the following RADIUS security server settings that will be used in theses instructions:

- 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.

SUMMARY STEPS

1. **configure terminal**
2. **ip radius source-interface vlan** *vlan interface number*
3. **radius-server host** {*hostname* | *ip-address*} **test username** *username*
4. **radius-server key** *string*
5. **radius-server dead-criteria tries** *num-tries*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ip radius source-interface vlan** *vlan interface number* **Example:** `Controller(config)# ip radius source-interface vlan 80`	Specifies that the RADIUS packets have the IP address of the indicated interface.
Step 3	**radius-server host** {*hostname* | *ip-address*} **test username** *username*	Specifies the host name or IP address of the remote RADIUS server.

	Command or Action	Purpose
	Example: `Controller(config)# radius-server host 172.120.39.46 test username user1`	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 4	**radius-server key** *string* **Example:** `Controller(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 5	**radius-server dead-criteria tries** *num-tries* **Example:** `Controller(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. 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.
Step 6	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

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.

Beginning in privileged EXEC mode, follow these steps to enable the server for either HTTP or HTTPS:

SUMMARY STEPS

1. **configure terminal**
2. **ip http server**
3. **ip http secure-server**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ip http server** **Example:** `Controller(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 3	**ip http secure-server** **Example:** `Controller(config)# ip http secure-server`	Enables HTTPS. You can configure custom authentication proxy web pages or specify a redirection URL for successful login. **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 4	**end** **Example:** `Controller(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)*."

Beginning in privileged EXEC mode, 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. **configure terminal**
2. **ip admission proxy http login page file** *device:login-filename*
3. **ip admission proxy http success page file** *device:success-filename*
4. **ip admission proxy http failure page file** *device:fail-filename*
5. **ip admission proxy http login expired page file** *device:expired-filename*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ip admission proxy http login page file** *device:login-filename* **Example:** `Controller(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 3	**ip admission proxy http success page file** *device:success-filename* **Example:** `Controller(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 4	**ip admission proxy http failure page file** *device:fail-filename* **Example:** `Controller(config)# ip admission proxy http fail`	Specifies the location of the custom HTML file to use in place of the default login failure page.

	Command or Action	Purpose
	`page file disk1:fail.htm`	
Step 5	**ip admission proxy http login expired page file** *device:expired-filename* **Example:** `Controller(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 6	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Verifying Custom Authentication Proxy Web Pages

This example shows how to verify the configuration of a custom authentication proxy web page:

```
Controller# show ip admission status
IP admission status:
  Enabled interfaces            0
  Total sessions                0
  Init sessions                 0     Max init sessions allowed     100
    Limit reached               0     Hi watermark                  0
  TCP half-open connections     0     Hi watermark                  0
  TCP new connections           0     Hi watermark                  0
  TCP half-open + new           0     Hi watermark                  0
  HTTPD1 Contexts               0     Hi watermark                  0

  Parameter Map: Global
    Custom Pages
      Custom pages not configured
    Banner
      Banner not configured
```

Related Topics

Authentication Proxy Web Page Guidelines, on page 817

Specifying a Redirection URL for Successful Login

Beginning in privileged EXEC mode, 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. **configure terminal**
2. **ip admission proxy http success redirect** *url-string*
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ip admission proxy http success redirect** *url-string* **Example:** `Controller(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 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Verifying Redirection URL for Successful Login

```
Controller# show ip admission status
 Enabled interfaces            0
 Total sessions                0
 Init sessions                 0     Max init sessions allowed     100
   Limit reached               0     Hi watermark                  0
 TCP half-open connections     0     Hi watermark                  0
 TCP new connections           0     Hi watermark                  0
 TCP half-open + new           0     Hi watermark                  0
 HTTPD1 Contexts               0     Hi watermark                  0

 Parameter Map: Global
   Custom Pages
     Custom pages not configured
   Banner
     Banner not configured
```

Related Topics

Redirection URL for Successful Login Guidelines, on page 818

Configuring the Web-Based Authentication Parameters

Beginning in privileged EXEC mode, 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. **configure terminal**
2. **ip admission max-login-attempts** *number*
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ip admission max-login-attempts** *number* **Example:** `Controller(config)# ip admission max-login-attempts 10`	Set sthe maximum number of failed login attempts. The range is 1 to 2147483647 attempts. The default is 5.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Configuring a Web Authentication Local Banner

Beginning in privileged EXEC mode, follow these steps to configure a local banner on a switch that has web authentication configured.

SUMMARY STEPS

1. **configure terminal**
2. **ip admission auth-proxy-banner http** [*banner-text* | *file-path*]
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ip admission auth-proxy-banner http** [*banner-text* | *file-path*] **Example:** `Controller(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 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Removing Web-Based Authentication Cache Entries

Beginning in privileged EXEC mode, follow these steps to remove web-based authentication cache entries:

SUMMARY STEPS

1. **clear ip auth-proxy cache** {***** | *host ip address*}
2. **clear ip admission cache** {***** | *host ip address*}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**clear ip auth-proxy cache** {***** | *host ip address*} **Example:** `Controller# 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.

	Command or Action	Purpose
Step 2	**clear ip admission cache** {* | *host ip address*} **Example:** `Controller# 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 96: Privileged EXEC show Commands

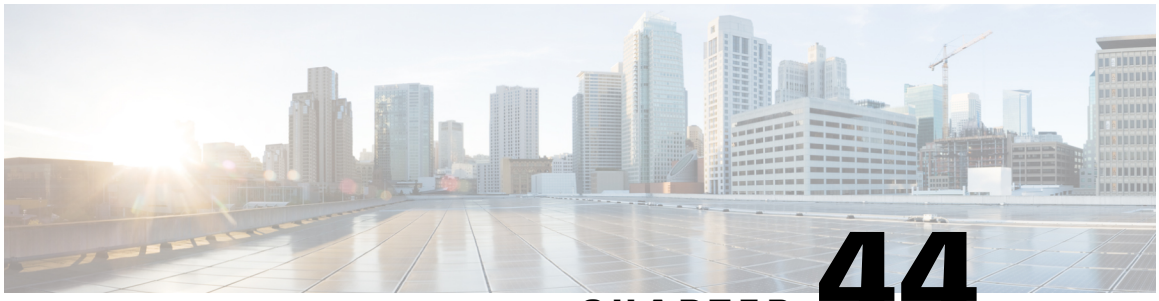
Command	Purpose
show authentication sessions method webauth	Displays the web-based authentication settings for all interfaces for fastethernet, gigabitethernet, or tengigabitethernet
show authentication sessions interface *type slot/port*[**details**]	Displays the web-based authentication settings for 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

This table lists the features in this module and provides links to specific configuration information.

Table 97: Feature Information for Web-Based Authentication

Feature Name	Releases	Feature Information



CHAPTER 44

Configuring Port-Based Traffic Control

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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.

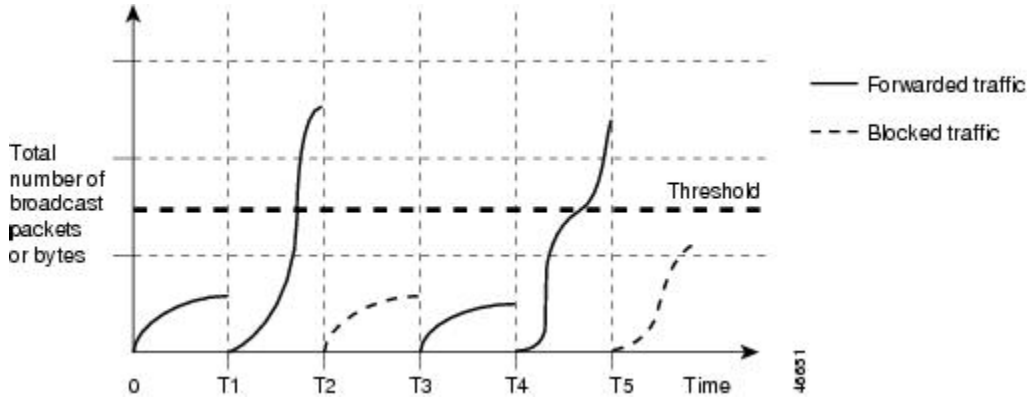
With each method, the port blocks traffic when the rising threshold is reached. The port remains blocked until the traffic rate drops below the falling threshold (if one is specified) and then resumes normal forwarding. If the falling suppression level is not specified, the switch blocks all traffic 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

This example shows broadcast traffic patterns on an interface over a given period of time.

Figure 39: Broadcast Storm Control Example



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.

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. **configure terminal**
2. **interface** *interface-id*
3. **storm-control** {**broadcast** | **multicast** | **unicast**} **level** {*level* [*level-low*] | **bps** *bps* [*bps-low*] | **pps** *pps* [*pps-low*]}
4. **storm-control action** {**shutdown** | **trap**}
5. **end**
6. **show storm-control** [*interface-id*] [**broadcast** | **multicast** | **unicast**]
7. **copy running-config startup-config**

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet1/0/1`	Specifies the interface to be configured, and enter interface configuration mode.
Step 3	**storm-control** {**broadcast** | **multicast** | **unicast**} **level** {*level* [*level-low*] | **bps** *bps* [*bps-low*] | **pps** *pps* [*pps-low*]} **Example:** `Controller(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. • 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 4	**storm-control action** {**shutdown** | **trap**}	Specifies the action to be taken when a storm is detected. The default is to filter out the traffic and not to send traps.

	Command or Action	Purpose
	Example: `Controller(config-if)# storm-control action trap`	• 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 5	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.
Step 6	**show storm-control** [*interface-id*] [**broadcast** | **multicast** | **unicast**] **Example:** `Controller# 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 7	**copy running-config startup-config** **Example:** `Controller# 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. In Cisco IOS Release 12.2(44)SE and later, you can configure a port to be error disabled if small frames arrive at a specified rate (threshold).

You globally enable the small-frame arrival feature on the switch and then configure the small-frame threshold for packets on each interface. Packets smaller than the minimum size and arriving at a specified rate (the threshold) are dropped since the port is error disabled.

SUMMARY STEPS

1. **configure terminal**
2. **errdisable detect cause small-frame**
3. **errdisable recovery interval** *interval*
4. **errdisable recovery cause small-frame**
5. **interface** *interface-id*
6. **small-frame violation-rate** *pps*
7. **end**
8. **show interfaces** *interface-id*
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**errdisable detect cause small-frame** **Example:** `Controller(config)# errdisable detect cause small-frame`	Enables the small-frame rate-arrival feature on the switch.
Step 3	**errdisable recovery interval** *interval* **Example:** `Controller(config)# errdisable recovery interval 60`	(Optional) Specifies the time to recover from the specified error-disabled state.
Step 4	**errdisable recovery cause small-frame** **Example:** `Controller(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 control settings propagate to the EtherChannel physical interfaces.
Step 5	**interface** *interface-id* **Example:** `Controller(config)# interface`	Enters interface configuration mode, and specify the interface to be configured.

	Command or Action	Purpose
	`gigabitethernet1/0/2`	
Step 6	**small-frame violation-rate** *pps* **Example:** `Controller(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 7	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.
Step 8	**show interfaces** *interface-id* **Example:** `Controller# show interfaces gigabitethernet1/0/2`	Verifies the configuration.
Step 9	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Monitoring Storm Control

Table 98: Commands for Displaying Storm Control Status and Configuration

Command	Purpose
show interfaces [*interface-id*] **switchport**	Displays the administrative and operational status of all switching (nonrouting) ports or the specified port, including port blocking and port protection settings.
show storm-control [*interface-id*] [**broadcast** | **multicast** | **unicast**]	Displays storm control suppression levels set on all interfaces or the specified interface for the specified traffic type or for broadcast traffic if no traffic type is entered.

Where to Go Next

-

Additional References

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 Information

Property Type	Property Value	Property Description

-

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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.

Do not configure a private-VLAN port as a protected port. Do not configure a protected port as a private-VLAN port. A private-VLAN isolated port does not forward traffic to other isolated ports or community ports.

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. **configure terminal**
2. **interface** *interface-id*
3. **switchport protected**
4. **end**
5. **show interfaces** *interface-id* **switchport**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet1/0/1`	Specifies the interface to be configured, and enter interface configuration mode.
Step 3	**switchport protected** **Example:** `Controller(config-if)# switchport protected`	Configures the interface to be a protected port.
Step 4	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	**show interfaces** *interface-id* **switchport** **Example:** `Controller# show interfaces gigabitethernet1/0/1 switchport`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Monitoring Protected Ports

Table 99: 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

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 Information

Property Type	Property Value	Property Description

-

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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. **configure terminal**
2. **interface** *interface-id*
3. **switchport block multicast**
4. **switchport block unicast**
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:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet1/0/1`	Specifies the interface to be configured, and enter interface configuration mode.
Step 3	**switchport block multicast** **Example:** `Controller(config-if)# switchport block multicast`	Blocks unknown multicast forwarding out of the port. **Note** Only pure Layer 2 multicast traffic is blocked. Multicast packets that contain IPv4 or IPv6 information in the header are not blocked.
Step 4	**switchport block unicast** **Example:** `Controller(config-if)# switchport block unicast`	Blocks unknown unicast forwarding out of the port.

	Command or Action	Purpose
Step 5	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.
Step 6	**show interfaces** *interface-id* **switchport** **Example:** `Controller# show interfaces gigabitethernet1/0/1 switchport`	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Monitoring Port Blocking

Table 100: 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

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 Information

Property Type	Property Value	Property Description

-

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 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

Enabling and Configuring Port Security, on page 854
Configuration Examples for Port Security, on page 861

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.

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 101: Security Violation Mode Actions

Violation Mode	Traffic is forwarded [15]	Sends SNMP trap	Sends syslog message	Displays error message [16]	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 [17]

15 Packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses.

16 The switch returns an error message if you manually configure an address that would cause a security violation.

17 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

Enabling and Configuring Port Security Aging, on page 858

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 102: Default Port Security Configuration

Feature	Default Setting
Port security	Disabled on a port.
Sticky address learning	Disabled.
Maximum number of secure MAC addresses per port	1.
Violation mode	Shutdown. The port shuts down when the maximum number of secure MAC addresses is exceeded.
Port security aging	Disabled. Aging time is 0. Static aging is disabled. Type is absolute.

Port Security Configuration Guidelines

- Port security can only be configured on static access ports or trunk ports. A secure port cannot be a dynamic access port.
- A secure port cannot be a destination port for Switched Port Analyzer (SPAN).
- A secure port cannot belong to a Gigabit EtherChannel port group.

Note Voice VLAN is only supported on access ports and not on trunk ports, even though the configuration is allowed.

- A secure port cannot be a private-VLAN port.
- 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 103: Port Security Compatibility with Other Switch Features

Type of Port or Feature on Port	Compatible with Port Security
DTP [18] port [19]	No
Trunk port	Yes
Dynamic-access port [20]	No
Routed port	No
SPAN source port	Yes
SPAN destination port	No
EtherChannel	No
Tunneling port	Yes
Protected port	Yes
IEEE 802.1x port	Yes
Voice VLAN port [21]	Yes
Private VLAN port	Yes
IP source guard	Yes
Dynamic Address Resolution Protocol (ARP) inspection	Yes
Flex Links	Yes

18 DTP=Dynamic Trunking Protocol

19 A port configured with the **switchport mode dynamic** interface configuration command.

20 A VLAN Query Protocol (VQP) port configured with the **switchport access vlan dynamic** interface configuration command.

21 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.

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. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode** {**access** | **trunk**}
4. **switchport voice vlan** *vlan-id*
5. **switchport port-security**
6. **switchport port-security [maximum** *value* [**vlan** {*vlan-list* | {**access** | **voice**}}]]
7. **switchport port-security violation** {**protect** | **restrict** | **shutdown** | **shutdown vlan**}
8. **switchport port-security [mac-address** *mac-address* [**vlan** {*vlan-id* | {**access** | **voice**}}]
9. **switchport port-security mac-address sticky**
10. **switchport port-security mac-address sticky** [*mac-address* | **vlan** {*vlan-id* | {**access** | **voice**}}]
11. **end**
12. **show port-security**
13. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet1/0/1`	Specifies the interface to be configured, and enter interface configuration mode.

	Command or Action	Purpose
Step 3	**switchport mode** {**access** | **trunk**} **Example:** `Controller(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 4	**switchport voice vlan** *vlan-id* **Example:** `Controller(config-if)# switchport voice vlan 22`	Enables voice VLAN on a port. vlan-id—Specifies the VLAN to be used for voice traffic.
Step 5	**switchport port-security** **Example:** `Controller(config-if)# switchport port-security`	Enable port security on the interface.
Step 6	**switchport port-security [maximum** *value* [**vlan** {*vlan-list* | {**access** | **voice**}}]] **Example:** `Controller(config-if)# switchport port-security maximum 20`	(Optional) Sets the maximum number of secure MAC addresses for the interface. 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 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 7	**switchport port-security violation** {**protect** | **restrict** | **shutdown** | **shutdown vlan**} **Example:** `Controller(config-if)# switchport`	(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

	Command or Action	Purpose
	`port-security violation restrict`	allowable addresses. You are not notified that a security violation has occurred. **Note** We do not recommend configuring the protect mode on a trunk port. The protect mode disables learning when any VLAN reaches its maximum limit, even if the port has not reached its maximum limit. • **restrict—**When the number of secure MAC addresses reaches the limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses or increase the number of maximum allowable addresses. An SNMP trap is sent, a syslog message is logged, and the violation counter increments. • **shutdown—**The interface is error-disabled when a violation occurs, and the port LED turns off. An SNMP trap is sent, a syslog message is logged, and the violation counter increments. • **shutdown vlan—**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 8	**switchport port-security [mac-address** *mac-address* [**vlan** {*vlan-id* | {**access** | **voice**}}]	(Optional) Enters a secure MAC address for the interface. You can use this command to enter the maximum number of secure MAC addresses. If you configure fewer secure MAC addresses than the maximum, the remaining MAC addresses are dynamically learned. **Note** If you enable sticky learning after you enter this command, the secure addresses that were dynamically learned are converted to sticky secure MAC addresses and are added to the running configuration. (Optional) **vlan—**sets a per-VLAN maximum value. Enter one of these options after you enter the **vlan** keyword: • *vlan-id***—**On a trunk port, 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. If an interface is configured for voice VLAN, configure a maximum of two secure MAC addresses.
	Example: `Controller(config-if)# switchport port-security mac-address 00:A0:C7:12:C9:25 vlan 3 voice`	
Step 9	**switchport port-security mac-address sticky**	(Optional) Enables sticky learning on the interface.

	Command or Action	Purpose
	Example: `Controller(config-if)# switchport port-security mac-address sticky`	
Step 10	**switchport port-security mac-address sticky** [*mac-address* | **vlan** {*vlan-id* | {**access** | **voice**}}] **Example:** `Controller(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 11	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.
Step 12	**show port-security** **Example:** `Controller# show port-security`	Verifies your entries.
Step 13	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Port Security, on page 849
Configuration Examples for Port Security, on page 861

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. **configure terminal**
2. **interface** *interface-id*
3. **switchport port-security aging** {**static** | **time** *time* | **type** {**absolute** | **inactivity**}}
4. **end**
5. **show port-security** [**interface** *interface-id*] [**address**]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet1/0/1`	Specifies the interface to be configured, and enter interface configuration mode.
Step 3	**switchport port-security aging** {**static** | **time** *time* | **type** {**absolute** | **inactivity**}} **Example:** `Controller(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.

	Command or Action	Purpose
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 5	**show port-security** [**interface** *interface-id*] [**address**] **Example:** `Controller# show port-security interface gigabitethernet1/0/1`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Port Security Aging, on page 851

Configuring Port Security and Private VLANs

Port security allows an administrator to limit the number of MAC addresses learned on a port or to define which MAC addresses can be learned on a port.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode private-vlan** {**host** | **promiscuous**}
4. **switchport port-security**
5. **end**
6. **show port-security** [**interface** *interface-id*] [**address**]
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet 1/0/8`	Specifies the interface to be configured, and enter interface configuration mode.
Step 3	**switchport mode private-vlan** {**host** | **promiscuous**} **Example:** `Controller(config-if)# switchport mode private-vlan promiscuous`	Enables a private vlan on the interface.
Step 4	**switchport port-security** **Example:** `Controller(config-if)# switchport port-security`	Enables port security on the interface.
Step 5	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.
Step 6	**show port-security** [**interface** *interface-id*] [**address**] **Example:** `Controller# show port-security interface gigabitethernet1/0/8`	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Monitoring Port Security

This table displays port security information.

Table 104: Commands for Displaying Port Security Status and Configuration

Command	Purpose
show port-security [**interface** *interface-id*]	Displays port security settings for the switch or for the specified interface, including the maximum allowed number of secure MAC addresses for each interface, the number of secure MAC addresses on the interface, the number of security violations that have occurred, and the violation mode.
show port-security [**interface** *interface-id*] **address**	Displays all secure MAC addresses configured on all switch interfaces or on a specified interface with aging information for each address.
show port-security interface *interface-id* **vlan**	Displays the number of secure MAC addresses configured per VLAN on the specified interface.

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.

```
Controller(config)# interface gigabitethernet1/0/1
Controller(config-if)# switchport mode access
Controller(config-if)# switchport port-security
Controller(config-if)# switchport port-security maximum 50
Controller(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:

```
Controller(config)# interface gigabitethernet1/0/2
Controller(config-if)# switchport mode trunk
Controller(config-if)# switchport port-security
Controller(config-if)# switchport port-security mac-address 0000.02000.0004 vlan 3
```

This example shows how to enable sticky port security on a port, to manually configure MAC addresses for data VLAN and voice VLAN, and to set the total maximum number of secure addresses to 20 (10 for data VLAN and 10 for voice VLAN).

```
Controller(config)# interface tengigabitethernet1/0/1
Controller(config-if)# switchport access vlan 21
Controller(config-if)# switchport mode access
Controller(config-if)# switchport voice vlan 22
Controller(config-if)# switchport port-security
Controller(config-if)# switchport port-security maximum 20
Controller(config-if)# switchport port-security violation restrict
```

```
Controller(config-if)# switchport port-security mac-address sticky
Controller(config-if)# switchport port-security mac-address sticky 0000.0000.0002
Controller(config-if)# switchport port-security mac-address 0000.0000.0003
Controller(config-if)# switchport port-security mac-address sticky 0000.0000.0001 vlan voice
Controller(config-if)# switchport port-security mac-address 0000.0000.0004 vlan voice
Controller(config-if)# switchport port-security maximum 10 vlan access
Controller(config-if)# switchport port-security maximum 10 vlan voice
```

Related Topics

Port Security, on page 849
Enabling and Configuring Port Security, on page 854

Where to Go Next

-

Additional References

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 Information

Property Type	Property Value	Property Description

•

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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. **configure termina**l
2. **psp** {**arp** | **dhcp** | **igmp**} pps *value*
3. **errdisable detect cause psp**
4. **errdisable recovery interval** *time*
5. **end**
6. **show psp config** {**arp** | **dhcp** | **igmp**}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure termina**l **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**psp** {**arp**	**dhcp**	**igmp**} pps *value* **Example:** `Controller(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 3	**errdisable detect cause psp** **Example:** `Controller(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 4	**errdisable recovery interval** *time* **Example:** `Controller`	(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.

	Command or Action	Purpose
Step 5	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 6	**show psp config** {**arp** | **dhcp** | **igmp**} **Example:** `Controller# show psp config dhcp`	Verifies your entries.

Monitoring Protocol Storm Protection

Command	Purpose
show psp config {**arp** | **dhcp** | **igmp**}	Verify your entries.

Where to Go Next

-

Additional References

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

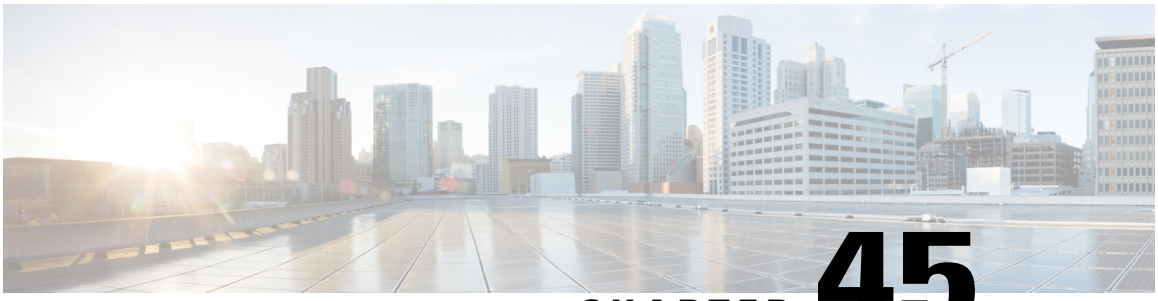
Technical Assistance

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 Information

Property Type	Property Value	Property Description

-



CHAPTER 45

Configuring IPv6 First Hop Security

Feature Information for First Hop Security in IPv6

Feature Name	Releases	Feature Information

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. For information, see the "Implementing IPv6 Addressing and Basic Connectivity" chapter of the Cisco IOS IPv6 Configuration Library on Cisco.com.

Restrictions for First Hop Security in IPv6

Although visible in the command-line help strings, the IPv6 first hop security (FHS) is not supported on the Catalyst 3750-G and 3750v2 switches. The command-line help strings are visible on these switches to support the FHS feature in a mixed switch stack scenario where one of these switches could become an active switch.

Information about First Hop Security in IPv6

First Hop Security in IPv6 (FHS IPv6) is a set of IPv6 policy features that can be applied to an 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:

First Hop Security in IPv6 Features	Description
IPv6 Snooping Policy	IPv6 Snooping Policy acts as a container policy that enables most of the features available with FHS in IPv6.
IPv6 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 L2 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 SA ND message is considered trustworthy if its IPv6-to-Media Access Control (MAC) mapping is verifiable.

First Hop Security in IPv6 Features	Description
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 L2 device with the information found in the received RA frame. Once the L2 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	You can use the DHCP guard to prevent forged messages from being entered in the binding table. The DHCP guard blocks DHCP server messages when they are received on ports that are not explicitly configured as facing a DHCP server or DHCP relay. To use this feature, configure a policy and attach it to a DHCP guard. To debug DHCP guard packets, use the debug ipv6 snooping dhcp-guard privileged EXEC command.

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:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ipv6 snooping policy***policy-name* **Example:** `Controller(config)# ipv6 snooping policy example_policy`	Creates a snooping policy and enters IPv6 Snooping Policy Configuration mode.
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:** Controller(config-ipv6-snooping)# security-level inspect **Example:** Controller(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.

	Command or Action	Purpose
Step 4	**end** **Example:** `Controller(config-ipv6-snooping)# exit`	Exits configuration modes to Privileged EXEC mode.
Step 5	**show ipv6 snooping policy** *policy-name* **Example:** `Controller#show ipv6 snooping policy example_policy`	Displays the snooping policy configuration.

What to Do Next

Attach an IPv6 Snooping policy to interfaces or VLANs.

How to Attach an IPv6 Snooping Policy to an Interface or a VLAN on an Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Snooping Policy ot and 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**}]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** Interface_type *stack/module/port* **Example:** `Controller(config)# interface gigabitethernet 1/1/4`	Specifies an interface type and identifier; enters the interface configuration mode.
Step 3	**switchport** **Example:** `Controller(config-if)# switchport`	Enters the Switchport mode.

	Command or Action	Purpose
		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.

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:
	`Controller(config-if)# ipv6 snooping`
	
	`or`
	
	`Controller(config-if)# ipv6 snooping attach-policy example_policy`
	
	`or`
	`Controller(config-if)# ipv6 snooping vlan 111,112`
	
	`or`
	
	`Controller(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.**

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*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**vlan configuration** *vlan_list* **Example:** `Controller(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:** `Controller(config-vlan-config)#ipv6 snooping attach-policy example_policy`	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 policy is, security-level **guard**, device-role **node**, protocol **ndp** and **dhcp.**

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:** `Controller# 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:** `Controller(config)# ipv6 neighbor binding`	
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:** `Controller(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:** `Controller(config)# ipv6 neighbor binding logging`	Enables the logging of binding table main events.
Step 5	**exit** **Example:** `Controller(config)# exit`	Exits global configuration mode, and places the router in privileged EXEC mode.
Step 6	**show ipv6 neighbor binding** **Example:** `Controller# 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. exit

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	[**no**]**ipv6 nd inspection policy** *policy-name* **Example:** `Controller(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:** `Controller(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:** `Controller(config-nd-inspection)# drop-unsecure`	Drops messages with no or invalid options or an invalid signature.
Step 5	**limit address-count** *value* **Example:** `Controller(config-nd-inspection)# limit address-count 1000`	Enter 1–10,000.

	Command or Action	Purpose
Step 6	**sec-level minimum** *value* **Example:** `Controller(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:** `Controller(config-nd-inspection)# tracking disable stale-lifetime infinite`	Overrides the default tracking policy on a port.
Step 8	**trusted-port** **Example:** `Controller(config-nd-inspection)# trusted-port`	Configures a port to become a trusted port.
Step 9	**validate source-mac** **Example:** `Controller(config-nd-inspection)# validate source-mac`	
Step 10	**no** {**device-role** | **drop-unsecure** | **limit address-count** | **sec-level minimum** | **tracking** | **trusted-port** | **validate source-mac**} **Example:** `Controller(config-nd-inspection)# no validate source-mac`	Remove the current configuration of a parameter with the **no** form of the command.
Step 11	**default** {**device-role** | **drop-unsecure** | **limit address-count** | **sec-level minimum** | **tracking** | **trusted-port** | **validate source-mac**} **Example:** `Controller(config-nd-inspection)# default limit address-count`	Restores configuration to the default values.
Step 12	exit **Example:** `Controller(config-nd-inspection)# default limit address-count`	Exits ND Inspection Configuration mode to Global 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. show command here

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** Interface_type *stack/module/port* **Example:** `Controller(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**}]
	
	Example:
	`Controller(config-if)# ipv6 nd inspection attach-policy example_policy`
	
	`or`
	
	`Controller(config-if)# ipv6 nd inspection attach-policy example_policy vlan 222,223,224`
	
	`or`
	
	`Controller(config-if)# ipv6 nd inspection 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	show command here **Example:** `Controller# show`	

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*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**vlan configuration** *vlan_list* **Example:** `Controller(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:** `Controller(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 xxWHAT??xx, sec-level minimum xxWHAT?xx, tracking xxWHAT?xx, no trusted-port, no validate source-mac.

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. show policy here

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**vlan configuration** *vlan_list* **Example:** `Controller(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:** `Controller(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	show policy here **Example:**	

How to Attach an IPv6 DHCP Guard Policy to 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**}]

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	**interface** Interface_type *stack/module/port* **Example:** `Controller(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_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
	
	Example:
	`Controller(config-if)# ipv6 dhcp guard attach-policy example_policy`
	
	`or`
	
	`Controller(config-if)# ipv6 dhcp guard attach-policy example_policy vlan 222,223,224`
	
	`or`
	
	`Controller(config-if)# ipv6 dhcp guard 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.

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. show policy here

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	**vlan configuration** *vlan_list* **Example:** `Controller(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:** `Controller(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	show policy here **Example:**	

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. **device-role** {**host** | **monitor** | **router** | **switch**}
4. **hop-limit** {**maximum** | **minimum**} *value*
5. **managed-config-flag** {**off** | **on**}
6. **match** {**ipv6 access-list** *list* | **ra prefix-list** *list*}
7. **other-config-flag** {**on** | **off**}
8. **router-preference maximum** {**high** | **medium** | **low**}
9. **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. **no** {**device-role** | **hop-limit** {**maximum** | **minimum**} | **managed-config-flag** | **match** {**ipv6 access-list** | **ra prefix-list** } | **other-config-flag** | **router-preference maximum**| **trusted-port**}
12. exit
13. Show command here

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	[**no**]**ipv6 nd raguard policy** *policy-name* **Example:** `Controller(config)# ipv6 nd raguard policy example_policy`	Specifies the RA Guard policy name and enters RA Guard Policy configuration mode.
Step 3	**device-role** {**host** | **monitor** | **router** | **switch**} **Example:** `Controller(config-nd-raguard)# device-role switch`	Specifies the role of the device attached to the port. The default is **host**.
Step 4	**hop-limit** {**maximum** | **minimum**} *value* **Example:** `Controller(config-nd-raguard)# hop-limit maximum 33`	Enables verification of the advertised Hop count limit. (1–255) Maximum hop count value allowed. (1–255) Minimum hop count value allowed.
Step 5	**managed-config-flag** {**off** | **on**} **Example:** `Controller(config-nd-raguard)# managed-config-flag on`	Enables verification of the advertised M flag
Step 6	**match** {**ipv6 access-list** *list* | **ra prefix-list** *list*} **Example:** `Controller(config-nd-raguard)# match ipv6 access-list example_list`	Matches a specified prefix list or access list.
Step 7	**other-config-flag** {**on** | **off**} **Example:** `Controller(config-nd-raguard)# other-config-flag on`	Enables verification of the advertised O flag.
Step 8	**router-preference maximum** {**high** | **medium** | **low**} **Example:** `Controller(config-nd-raguard)# router-preference maximum high`	Enables verification of the advertised Router Preference flag. • **high—**Discards RAs with router preference greater than high. • **low—**Discards RAs with router preference greater than low. • **medium—**Discards RAs with router preference greater than medium.

	Command or Action	Purpose
Step 9	**trusted-port** **Example:** `Controller(config-nd-raguard)# trusted-port`	Configures a port to become a trusted port.
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:** `Controller(config-nd-raguard)# default hop-limit`	Restores a command to its default value.
Step 11	**no** {**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:** `Controller(config-nd-raguard)# no match ipv6 access-list`	Remove the current configuration of a parameter with the **no** form of the command.
Step 12	exit **Example:** `Controller(config-nd-raguard)# default limit address-count`	Exits ND RA Guard configuration mode to Global Configuration mode.
Step 13	Show command here **Example:** `Controller(config-nd-raguard)# show`	(Optional)—Exits the ND Guard Policy configuration mode to Global configuration mode.

How to Attach an IPv6 RA 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. show command here

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** Interface_type *stack/module/port* **Example:** `Controller(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_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
	
	Example:
	`Controller(config-if)# ipv6 nd raguard attach-policy example_policy`
	
	`or`
	
	`Controller(config-if)# ipv6 nd raguard attach-policy example_policy vlan 222,223,224`
	
	or
	
	`Controller(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	show command here **Example:** `Controller# show`	

How to Attach an IPv6 RA Guard Policy to VLANs Globally

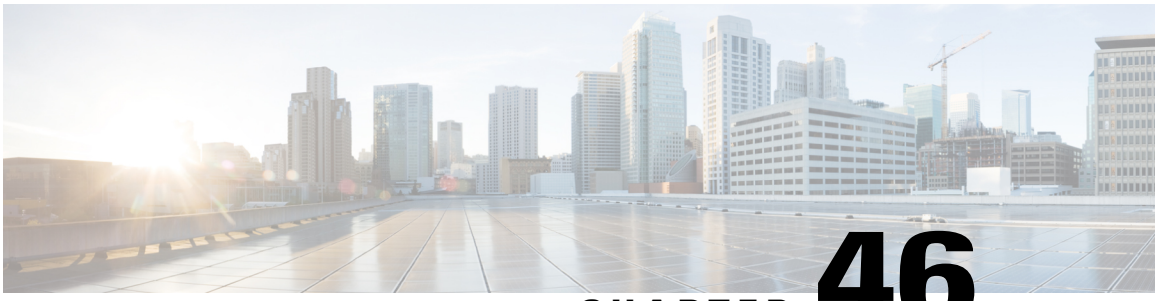
Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Router Advertisement policy to VLANs regardless of interface:

SUMMARY STEPS

1. **configure terminal**
2. **vlan configuration** *vlan_list*
3. **ipv6 dhcp guard** [**attach-policy** *policy_name*]
4. show policy here

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**vlan configuration** *vlan_list* **Example:** `Controller(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:** `Controller(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	show policy here **Example:**	



CHAPTER 46

Configuring Wireless Guest Access

Configuring Guest Access

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Guest Access

- All mobility peers should be configured for hierarchical mobility architecture.
- For Guest Controller Mobility Anchor configuration on WLAN is must on Mobility Agent and Guest Controller.
- Guest Access can be a 3 box solution or 2 box solution. The mobility tunnel link status should be up between:
    - Mobility Agent, Mobility Controller and Guest Controller.

  or
    - Mobility Agent/Mobility Controller and Guest Controller

Restrictions for Guess Access

Guest Controller functionality is supported only on Catalyst 5760.

Information about Wireless Guest Access

Ideally, the implementation of a wireless guest network uses as much of an enterprise's existing wireless and wired infrastructure as possible to avoid the cost and complexity of building a physical overlay network. Assuming this is the case, the following additional elements and functions are needed:

- A dedicated guest WLAN/SSID—Implemented throughout the campus wireless network wherever guest access is required. A guest WLAN is identified by a WLAN with mobility anchor (Guest Controller) configured.
- Guest traffic segregation—Requires implementing Layer 2 or Layer 3 techniques across the campus network to restrict where guests are allowed to go.
- Access control—Involves using imbedded access control functionality within the campus network or implementing an external platform to control guest access to the Internet from the enterprise network.
- Guest user credential management—A process by which a sponsor or lobby administrator can create temporary credentials in behalf of a guest. This function might be resident within an access control platform or it might be a component of AAA or some other management system.

Fast Secure Roaming

Fast secure roaming can be achieved by caching the Pairwise Master Key (PMK) information for Cisco Centralized Key Management (CCKM), 802.11r and 802.11i clients. Cisco Centralized Key Management (CCKM) helps to improve roaming. Only the client can initiate the roaming process, which depends on factors such as:

- Overlap between APs
- Distance between APs
- Channel, signal strength, and load on the AP
- Data rates and output power

Whenever a fast-roaming client 802.11i, [CCKM]) roams to a new device, after fast-roaming the clients go through mobility "handoff" procedure. And new AAA attributes learned through mobility "handoff" procedure get re-applied.

Full L2 authentication must be avoided during roaming if the client uses the 802.11i WPA2, CCKM, 802.11r to achieve the full requirements of fast secure roaming. The PMK cache (802.11i, CCKM, and 802.11r) is used to authenticate and derive the keys for roaming clients to avoid full L2 authentication. This requires all Mobility Anchors (MA) and Mobility Controllers (MC) in the mobility group to have the same PMK cache values.

The session timeout defines when a PMK cache will expire. A PMK cache can also be deleted when a client fails to re-authenticate or when it is manually deleted them from the CLI. The deletion on the original controller or switch shall be propagated to other controllers or switches in the same mobility group.

How to Configure Guest Access

Creating a Lobby Administrator Account

SUMMARY STEPS

1. **configure terminal**
2. **user-name** *user-name*
3. **type lobby-admin**
4. **password 0** *password*
5. **end**
6. **show running-config | section** *user-name* (or) **show running-config | section** *configured lobby admin username*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller # configure terminal`	Enters global configuration mode.
Step 2	**user-name** *user-name* **Example:** `Controller (config)# user-name lobby`	Creates a user account.
Step 3	**type lobby-admin** **Example:** `Controller (config-user-name)# type lobby-admin`	Specifies the account type as lobby admin.
Step 4	**password 0** *password* **Example:** `Controller(config-user-name)# password 0 lobby`	Creates a password for the lobby administrator account.
Step 5	**end** **Example:** `Controller (config-user-name)# end`	Returns to privileged EXEC mode.
Step 6	**show running-config | section** *user-name* (or) **show running-config | section** *configured lobby admin username* **Example:** `Controller # show running-config | section lobby`	Displays the configuration details.

Configuring Guest User Accounts

SUMMARY STEPS

1. **configure terminal**
2. **user-name** *user-name*
3. **password** *unencrypted/hidden-password password*
4. **type network-user description** *description* **guest-user lifetime year** *0-1* **month** *0-11* **day** *0-30* **hour** *0-23* **minute** *0-59* **second** *0-59*
5. **end**
6. **show aaa local netuser all**
7. **show running-config | section***user-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller # configure terminal`	Enters global configuration mode.
Step 2	**user-name** *user-name* **Example:** `Controller (config)# user-name guest`	Creates a username for the lobby ambassador account.
Step 3	**password** *unencrypted/hidden-password password* **Example:** `Controller (config-user-name)# password 0 guest`	Specifies the password for the user.
Step 4	**type network-user description** *description* **guest-user lifetime year** *0-1* **month** *0-11* **day** *0-30* **hour** *0-23* **minute** *0-59* **second** *0-59* **Example:** `Controller (config-user-name)# type network-user description guest guest-user lifetime year 1 month 10 day 3 hour 1 minute 5 second 30`	Specifies the type of user.
Step 5	**end** **Example:** `Controller (config-user-name)# end`	Returns to privileged EXEC mode.
Step 6	**show aaa local netuser all** **Example:** `Controller # show aaa local netuser all`	Displays the configuration details. After the lifetime, the user-name with guest type will be deleted and the client associated with the guest user-name will be de-authenticated.
Step 7	**show running-config | section***user-name*	Displays the configuration details.

	Command or Action	Purpose
	Example: `Controller # show running-config | section guest`	

Configuring Mobility Agent (MA)

SUMMARY STEPS

1. **configure terminal**
2. **wireless mobility controller ip***mc-ipaddress* **public-ip** *mc-publicipaddress*
3. **wlan** *wlan-name wlan-id ssid*
4. **client vlan id***vlan-group name/vlan-id*
5. **no security wpa**
6. **mobility anchor** *ipaddress*
7. **aaa-override**
8. **no shutdown**
9. **end**
10. **show wireless mobility summary**
11. **show wlan name** *wlan-name/id*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller # configure terminal`	Enters global configuration mode.
Step 2	**wireless mobility controller ip***mc-ipaddress* **public-ip** *mc-publicipaddress* **Example:** `Controller (config) # wireless mobility controller ip27.0.0.1 public-ip 27.0.0.1`	Configures the Mobility Controller to which the MA will be associated.
Step 3	**wlan** *wlan-name wlan-id ssid* **Example:** `Controller (config) # wlan mywlan 34 mywlan-ssid`	• For *wlan-name* enter, enter the profile name. The range is 1- 32 characters. • For *wlan-id*, enter the WLAN ID. The range is 1-512. • For *ssid*, enter the Service Set IDentifier (SSID) for this WLAN. If the SSID is not specified, the WLAN profile name is set as the SSID.

	Command or Action	Purpose
Step 4	**client vlan id***vlan-group name/vlan-id* **Example:** `Controller (config-wlan) # client vlan VLAN0136`	Configures the VLAN id or group of the WLAN.
Step 5	**no security wpa** **Example:** `Controller (config-wlan) # no security wpa`	The security configuration must be the same for the WLAN created on the GC. This example is for open authentication. For other security types such as open and webauth, appropriate command should be provided.
Step 6	**mobility anchor** *ipaddress* **Example:** `Controller (config-wlan) # mobility anchor 9.3.32.2`	Configures the Guest Controller as mobility anchor.
Step 7	**aaa-override** **Example:** `Controller (config-wlan) # aaa-override`	(Optional) Enables AAA override. AAA override is required for non open authentication in case AAA attributes are to be prioritized. It is required only in case guest user need to be deauthenticated after lifetime or have to give aaa-override attribute to the user.
Step 8	**no shutdown** **Example:** `Controller(config-wlan) # no shutdown`	Enables the WLAN.
Step 9	**end** **Example:** `Controller (config) # end`	Returns to privileged EXEC mode.
Step 10	**show wireless mobility summary** **Example:** `Controller  # show wireless mobility summary`	Verifies the mobility controller IP address and mobility tunnel status.
Step 11	**show wlan name** *wlan-name/id* **Example:** `Controller # show wlan name mywlan`	Displays the configuration of mobility anchor.

Configuring Mobility Controller

Mobility Controller mode should be enabled using the command **wireless mobility controller**.

SUMMARY STEPS

1. **configure terminal**
2. **wireless mobility group member ip** *ip-address* **public-ip** *ip-address* **group** *group-name*
3. **wireless mobility controller peer-group** *peer-group-name*
4. **wireless mobility controller peer-group** *peer-group-name* **member ip** *ipaddress* **public-ip** *ipaddress*
5. **end**
6. **show wireless mobility summary**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller # configure terminal`	Enters global configuration mode.
Step 2	**wireless mobility group member ip** *ip-address* **public-ip** *ip-address* **group** *group-name* **Example:** `Controller (config) # wireless mobility group member ip 27.0.0.1 public-ip 23.0.0.1 group test`	Adds all peers within the MC group. The *ip-address* should be the guest controller's IP address.
Step 3	**wireless mobility controller peer-group** *peer-group-name* **Example:** `Controller (config) # wireless mobility controller peer-group pg`	Creates the switch peer group.
Step 4	**wireless mobility controller peer-group** *peer-group-name* **member ip** *ipaddress* **public-ip** *ipaddress* **Example:** `Controller (config) # wireless mobility controller peer-group pg member ip 9.7.136.10 public-ip 9.7.136.10`	Adds the MA to the switch peer group.
Step 5	**end** **Example:** `Controller (config) # end`	Returns to privileged EXEC mode.
Step 6	**show wireless mobility summary** **Example:** `Controller # show wireless mobility summary`	Displays the configuration details.

Configuring Guest Controller

SUMMARY STEPS

1. **configure terminal**
2. **wireless mobility group member ip** *ip-address* **public-ip** *ip-address* **group** *group-name*
3. **wlan** *wlan-name wlan-id ssid*
4. **client vlan id***vlan-group name/vlan-id*
5. **no security wpa**
6. **mobility anchor** *self-ipaddress*
7. **aaa-override**
8. **security web-auth authentication-list***authentication list name*
9. **security web-auth parameter-map** *parameter-map name*
10. **no shutdown**
11. **end**
12. **show wireless mobility summary**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller # configure terminal`	Enters global configuration mode.
Step 2	**wireless mobility group member ip** *ip-address* **public-ip** *ip-address* **group** *group-name* **Example:** `Controller (config) # wireless mobility group member ip 27.0.0.1 public-ip 23.0.0.1 group test`	Configures the mobility group member with mobility controller's IP address.
Step 3	**wlan** *wlan-name wlan-id ssid* **Example:** `Controller (config) # wlan mywlan 34 mywlan-ssid`	• For *wlan-name* enter, enter the profile name. The range is 1- 32 characters. • For *wlan-id*, enter the WLAN ID. The range is 1-64. • For *ssid*, enter the Service Set IDentifier (SSID) for this WLAN. If the SSID is not specified, the WLAN profile name is set as the SSID.
Step 4	**client vlan id***vlan-group name/vlan-id* **Example:** `Controller (config-wlan) # client vlan VLAN0136`	Configures the VLAN id or group of the WLAN.

	Command or Action	Purpose
Step 5	**no security wpa** **Example:** `Controller (config-wlan) # no security wpa`	The security configuration must be the same for the WLAN created on the MA. For security types such as open and webauth, appropriate command should be provided.
Step 6	**mobility anchor** *self-ipaddress* **Example:** `Controller (config-wlan) # mobility anchor 9.3.32.2`	Configures WLAN with the mobility anchor IP as a self IP address.
Step 7	**aaa-override** **Example:** `Controller (config-wlan) # aaa-override`	(Optional) Enables AAA override.
Step 8	**security web-auth authentication-list***authentication list name* **Example:** `Controller (config-wlan) # security web-auth authentication-list test`	Allows you to map the authentication list name with the web-auth WLAN.
Step 9	**security web-auth parameter-map** *parameter-map name* **Example:** `Controller (config-wlan) # security web-auth parameter-map webparalocal`	Allows you to map the parameter-map name with the web-auth WLAN.
Step 10	**no shutdown** **Example:** `Controller (config-wlan) # no shutdown`	Enables the WLAN.
Step 11	**end** **Example:** `Controller (config-wlan) # end`	Returns to privileged EXEC mode.
Step 12	**show wireless mobility summary** **Example:** `Controller # show wireless mobility summary`	Displays the configuration details. The WLAN configuration in MA and MC should be same including the WLAN id.

Obtaining a Web Authentication Certificate

SUMMARY STEPS

1. **configure terminal**
2. **crypto pki import** *trustpoint name* **pkcs12 tftp:** *passphrase*
3. **end**
4. **show crypto pki** *trustpoints cert*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller # configure terminal`	Enters global configuration mode.
Step 2	**crypto pki import** *trustpoint name* **pkcs12 tftp:** *passphrase* **Example:** `Controller (config)# crypto pki import cert pkcs12 tftp://9.1.0.100/ldapserver-cert.p12 cisco`	Imports certificate.
Step 3	**end** **Example:** `Controller (config)# end`	Returns to privileged EXEC mode.
Step 4	**show crypto pki** *trustpoints cert* **Example:** `Controller # show crypto pki trustpoints cert`	Displays the configuration details.

Displaying a Web Authentication Certificate

SUMMARY STEPS

1. **show crypto ca certificate verb**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show crypto ca certificate verb** **Example:** `Controller # show crypto ca certificate verb`	Displays the current web authentication certificate details.

Choosing the Default Web Authentication Login Page

AAA override flag should be enabled on the WLAN for Web-Authentication using local or remote AAA server.

SUMMARY STEPS

1. **configure terminal**
2. **parameter-map type webauth** *parameter-map name*
3. **wlan** *wlan-name*
4. **shutdown**
5. **security web-auth authentication-list** *authentication list name*
6. **security web-auth parameter-map** *parameter-map name*
7. **no shutdown**
8. **end**
9. **show running-config | section** *wlan-name*
10. **show running-config | section parameter-map type webauth** *parameter-map*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller # configure terminal`	Enters global configuration mode.
Step 2	**parameter-map type webauth** *parameter-map name* **Example:** `Controller (config) # parameter-map type webauth test`	Configures the web-auth parameter-map.
Step 3	**wlan** *wlan-name* **Example:** `Controller (config) # wlan wlan10`	For the wlan-name, enter the profile name. The range is 1- 32 characters.
Step 4	**shutdown**	Disables WLAN.

	Command or Action	Purpose
	Example: `Controller (config) # shutdown`	
Step 5	**security web-auth authentication-list** *authentication list name* **Example:** `Controller (config-wlan) # security web-auth authentication-list test`	Allows you to map the authentication list name with the web-auth WLAN.
Step 6	**security web-auth parameter-map** *parameter-map name* **Example:** `Controller (config) # security web-auth parameter-map test`	Allows you to map the parameter-map name with the web-auth WLAN.
Step 7	**no shutdown** **Example:** `Controller (config) # no shutdown`	Enables the WLAN.
Step 8	**end** **Example:** `Controller (config) # end`	Returns to privileged EXEC mode.
Step 9	**show running-config | section** *wlan-name* **Example:** `Controller# show running-config | section mywlan`	Displays the configuration details.
Step 10	**show running-config | section parameter-map type webauth** *parameter-map* **Example:** `Controller# show running-config | section parameter-map type webauth test`	Displays the configuration details.

Choosing a Customized Web Authentication Login Page from an External Web Server

AAA override flag should be enabled on the WLAN for Web Authentication using local or remote AAA server.

SUMMARY STEPS

1. **configure terminal**
2. **parameter-map type webauth global**
3. **virtual-ip {ipv4 | ipv6}** *ip-address*
4. **parameter-map type webauth** *parameter-map name*
5. **type {authbypass | consent | webauth | webconsent}**
6. **redirect [for-login|on-success|on-failure]** *URL*
7. **redirect portal {ipv4 | ipv6}** *ip-address*
8. **end**
9. show running-config | section parameter-map

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller # configure terminal`	Enters global configuration mode.
Step 2	**parameter-map type webauth global** **Example:** `Controller (config) # parameter-map type webauth global`	Configures a global webauth type parameter.
Step 3	**virtual-ip {ipv4 | ipv6}** *ip-address* **Example:** `Controller (config-params-parameter-map) # virtual-ip ipv4 1.1.1.1`	Configures the virtual IP address.
Step 4	**parameter-map type webauth** *parameter-map name* **Example:** `Controller (config-params-parameter-map) # parameter-map type webauth test`	Configures the webauth type parameter.
Step 5	**type {authbypass | consent | webauth | webconsent}** **Example:** `Controller (config-params-parameter-map) # type webauth`	Configures webauth subtypes such as consent, passthru, webauth, or webconsent.

Step 6	**redirect [for-login|on-success|on-failure]** *URL*
	
	Example:
	`Controller (config-params-parameter-map)`
	`# redirect for-login http://9.1.0.100/login.html`
		Configures the redirect URL for the log in page, success page, and failure page.

Step 7	**redirect portal {ipv4 | ipv6}** *ip-address*	Configures the external portal IPv4 address.

	Command or Action	Purpose
	Example: `Controller (config-params-parameter-map) # redirect portal ipv4 23.0.0.1`	
Step 8	**end** **Example:** `Controller (config-params-parameter-map) # end`	Returns to privileged EXEC mode.
Step 9	show running-config | section parameter-map **Example:** `Controller # show running-config | section parameter-map`	Displays the configuration details.

Assigning Login, Login Failure, and Logout Pages per WLAN

SUMMARY STEPS

1. **configure terminal**
2. **parameter-map type webauth** *parameter-map-name*
3. **custom-page login device** *html-filename*
4. **custom-page login expired** *html-filename*
5. **custom-page failure device** *html-filename*
6. **custom-page success device** *html-filename*
7. **end**
8. **show running-config | section parameter-map type webauth** *parameter-map*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller # configure terminal`	Enters global configuration mode.
Step 2	**parameter-map type webauth** *parameter-map-name* **Example:** `Controller (config) # parameter-map type webauth test`	Configures the webauth type parameter.

	Command or Action	Purpose
Step 3	**custom-page login device** *html-filename* **Example:** `Controller (config-params-parameter-map)# custom-page login device device flash:login.html`	Allows you to specify the filename for web authentication customized login page.
Step 4	**custom-page login expired** *html-filename* **Example:** `Controller (config-params-parameter-map)# custom-page login expired device flash:loginexpired.html`	Allows you to specify the filename for web authentication customized login expiry page.
Step 5	**custom-page failure device** *html-filename* **Example:** `Controller (config-params-parameter-map)# custom-page failure device device flash:loginfail.html`	Allows you to specify the filename for web authentication customized login failure page.
Step 6	**custom-page success device** *html-filename* **Example:** `Controller (config-params-parameter-map)# custom-page success device device flash:loginsuccess.html`	Allows you to specify the filename for web authentication customized login success page.
Step 7	**end** **Example:** `Controller (config-params-parameter-map)# end`	Returns to privileged EXEC mode.
Step 8	**show running-config | section parameter-map type webauth** *parameter-map* **Example:** `Controller (config) # show  running-config | section parameter-map type webauth test`	Displays the configuration details.

Configuring AAA-Override

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan-name*
3. **aaa-override**
4. **end**
5. **show running-config | section** *wlan-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller # configure terminal`	Enters global configuration mode.
Step 2	**wlan** *wlan-name* **Example:** `Controller (config) # wlan ramban`	For *wlan-name*, enter the profile name. The range is 1- 32 characters.
Step 3	**aaa-override** **Example:** `Controller (config-wlan) # aaa-override`	Enables AAA override on the WLAN.
Step 4	**end** **Example:** `Controller (config-wlan) # end`	Returns to privileged EXEC mode.
Step 5	**show running-config | section** *wlan-name* **Example:** `Controller # show running-config | section ramban`	Displays the configuration details.

Configuring Client Load Balancing

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan-name*
3. **shutdown**
4. **mobility anchor** *ip-address1*
5. **mobility anchor** *ip-address2*
6. **no shutdown wlan**
7. **end**
8. **show running-config | section** *wlan-name*

DETAILED STEPS

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

	Command or Action	Purpose
	Example: `Controller # configure terminal`	
Step 2	**wlan** *wlan-name* **Example:** `Controller (config)# wlan ramban`	For *wlan-name*, enter the profile name.
Step 3	**shutdown** **Example:** `Controller (config-wlan)# shutdown`	Disables WLAN.
Step 4	**mobility anchor** *ip-address1* **Example:** `Controller (config-wlan) # mobility anchor 9.7.136.15`	Configures a guest controller as mobility anchor.
Step 5	**mobility anchor** *ip-address2* **Example:** `Controller (config-wlan) # mobility anchor 9.7.136.16`	Configures a guest controller as mobility anchor.
Step 6	**no shutdown wlan** **Example:** `Controller (config-wlan) # no shutdown wlan`	Enables the WLAN.
Step 7	**end** **Example:** `Controller (config-wlan) # end`	Returns to privileged EXEC mode.
Step 8	**show running-config | section** *wlan-name* **Example:** `Controller # show running-config | section ramban`	Displays the configuration details.

Configuring Preauthentication ACL

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan-name*
3. **shutdown**
4. **ip access-group web** *preauthrule*
5. **no shutdown**
6. **end**
7. **show wlan name** *wlan-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *wlan-name* **Example:** `Controller (config)# wlan ramban`	For *wlan-name*, enter the profile name.
Step 3	**shutdown** **Example:** `Controller (config-wlan)# shutdown`	Disables the WLAN.
Step 4	**ip access-group web** *preauthrule* **Example:** `Controller (config-wlan)# ip access-group web preauthrule`	Configures ACL that has to be applied before authentication.
Step 5	**no shutdown** **Example:** `Controller (config)# no shutdown`	Enables the WLAN.
Step 6	**end** **Example:** `Controller (config-wlan)# end`	Returns to privileged EXEC mode.
Step 7	**show wlan name** *wlan-name* **Example:** `Controller# show wlan name ramban`	Displays the configuration details.

Configuring IOS ACL Definition

SUMMARY STEPS

1. **configure terminal**
2. **ip access-list extended** *access-list number*
3. **permit udp any eq** *port number* **any**
4. **end**
5. **show access-lists** *ACL number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller # configure terminal`	Enters global configuration mode.
Step 2	**ip access-list extended** *access-list number* **Example:** `Controller (config) # ip access-list extended 102`	Configures extended IP access-list.
Step 3	**permit udp any eq** *port number* **any** **Example:** `Controller (config-ext-nacl) # permit udp any eq 8080 any`	Configures destination host.
Step 4	**end** **Example:** `Controller (config-wlan) # end`	Returns to privileged EXEC mode.
Step 5	**show access-lists** *ACL number* **Example:** `Controller # show access-lists 102`	Displays the configuration details.

Configuring Webpassthrough

SUMMARY STEPS

1. **configure terminal**
2. **parameter-map type webauth** *parameter-map name*
3. **type consent**
4. **end**
5. **show running-config | section parameter-map type webauth** *parameter-map*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller # configure terminal`	Enters global configuration mode.
Step 2	**parameter-map type webauth** *parameter-map name* **Example:** `Controller (config)  # parameter-map type webauth webparalocal`	Configures the webauth type parameter.
Step 3	**type consent** **Example:** `Controller (config-params-parameter-map) # type consent`	Configures webauth type as consent.
Step 4	**end** **Example:** `Controller (config-params-parameter-map) # end`	Returns to privileged EXEC mode.
Step 5	**show running-config | section parameter-map type webauth** *parameter-map* **Example:** `Controller (config) # show  running-config | section parameter-map type webauth test`	Displays the configuration details.

Configuration Examples for Guest Access

Example: Creating a Lobby Ambassador Account

This example shows how to configure a lobby ambassador account.

```
Controller# configure terminal
```

```
Controller(config)# user-name lobby
Controller(config)# type lobby-admin
Controller(config)# password 0 lobby
Controller(config)# end
Controller#  show running-config | section lobby
    user-name lobby
    creation-time 1351118727
    password 0 lobby
    type lobby-admin
```

Example: Obtaining Web Authentication Certificate

This example shows how to obtain web authentication certificate.

```
Controller# configure terminal
Controller(config)# crypto pki import cert pkcs12 tftp://9.1.0.100/ldapserver-cert.p12 cisco
Controller(config)# end
Controller# show crypto pki trustpoints cert
 Trustpoint cert:
    Subject Name:
    e=rkannajr@cisco.com
    cn=sthaliya-lnx
    ou=WNBU
    o=Cisco
    l=SanJose
    st=California
    c=US
          Serial Number (hex): 00
    Certificate configured.
Controller# show  crypto pki certificates cert
Certificate
  Status: Available
  Certificate Serial Number (hex): 04
  Certificate Usage: General Purpose
  Issuer:
    e=rkannajr@cisco.com
    cn=sthaliya-lnx
    ou=WNBU
    o=Cisco
    l=SanJose
    st=California
    c=US
  Subject:
    Name: ldapserver
    e=rkannajr@cisco.com
    cn=ldapserver
    ou=WNBU
    o=Cisco
    st=California
    c=US
  Validity Date:
    start date: 07:35:23 UTC Jan 31 2012
    end   date: 07:35:23 UTC Jan 28 2022
  Associated Trustpoints: cert ldap12
  Storage: nvram:rkannajrcisc#4.cer

CA Certificate
  Status: Available
  Certificate Serial Number (hex): 00
  Certificate Usage: General Purpose
  Issuer:
    e=rkannajr@cisco.com
    cn=sthaliya-lnx
    ou=WNBU
    o=Cisco
    l=SanJose
    st=California
    c=US
  Subject:
    e=rkannajr@cisco.com
```

```
    cn=sthaliya-lnx
    ou=WNBU
    o=Cisco
    l=SanJose
    st=California
    c=US
  Validity Date:
    start date: 07:27:56 UTC Jan 31 2012
    end   date: 07:27:56 UTC Jan 28 2022
  Associated Trustpoints: cert ldap12 ldap
  Storage: nvram:rkannajrcisc#0CA.cer
```

Example: Displaying a Web Authentication Certificate

This example shows how to display a web authentication certificate.

```
Controller# show crypto ca certificate verb
     Certificate
     Status: Available
     Version: 3
     Certificate Serial Number (hex): 2A9636AC00000000858B
     Certificate Usage: General Purpose
     Issuer:
    cn=Cisco Manufacturing CA
    o=Cisco Systems
    Subject:
    Name: WS-C3780-6DS-S-2037064C0E80
    Serial Number: PID:WS-C3780-6DS-S SN:FOC1534X12Q
    cn=WS-C3780-6DS-S-2037064C0E80
    serialNumber=PID:WS-C3780-6DS-S SN:FOC1534X12Q
    CRL Distribution Points:
    http://www.cisco.com/security/pki/crl/cmca.crl
    Validity Date:
    start date: 15:43:22 UTC Aug 21 2011
    end   date: 15:53:22 UTC Aug 21 2021
    Subject Key Info:
    Public Key Algorithm: rsaEncryption
    RSA Public Key: (1024 bit)
    Signature Algorithm: SHA1 with RSA Encryption
    Fingerprint MD5: A310B856 A41565F1 1D9410B5 7284CB21
    Fingerprint SHA1: 04F180F6 CA1A67AF 9D7F561A 2BB397A1 0F5EB3C9
    X509v3 extensions:
    X509v3 Key Usage: F0000000
      Digital Signature
      Non Repudiation
      Key Encipherment
      Data Encipherment
    X509v3 Subject Key ID: B9EEB123 5A3764B4 5E9C54A7 46E6EECA 02D283F7
    X509v3 Authority Key ID: D0C52226 AB4F4660 ECAE0591 C7DC5AD1 B047F76C
    Authority Info Access:
    Associated Trustpoints: CISCO_IDEVID_SUDI
    Key Label: CISCO_IDEVID_SUDI
```

Example: Configuring Guest User Accounts

This example shows how to configure a guest user account.

```
Controller# configure terminal
Controller(config)# user-name guest
Controller(config-user-name)# password 0 guest
Controller(config-user-name)# type network-user description guest guest-user lifetime year
 1 month 10 day 3 hour 1 minute 5 second 30
Controller(config-user-name)# end
Controller# show aaa local netuser all
User-Name           : guest
Type                : guest
```

```
Password            : guest
Is_passwd_encrypted : No
Descriptio          : guest
Attribute-List      : Not-Configured
First-Login-Time    : Not-Logged-In
Num-Login           : 0
Lifetime            : 1 years 10 months 3 days 1 hours 5 mins 30 secs
Start-Time          : 20:47:37 chennai Dec 21 2012
```

Example: Configuring Mobility Controller

This example shows how to configure a mobility controller.

```
Controller# configure terminal
Controller(config)# wireless mobility group member ip 27.0.0.1 public-ip 23.0.0.1 group
test
Controller(config)# wireless mobility controller peer-group pg
Controller(config)# wireless mobility controller peer-group pg member ip 9.7.136.10 public-ip
 9.7.136.10
Controller(config)# end
Controller# show wireless mobility summary

Mobility Controller Summary:

Mobility Role                                 : Mobility Controller
Mobility Protocol Port                        : 16666
Mobility Group Name                           : default
Mobility Oracle                               : Enabled
DTLS Mode                                     : Enabled
Mobility Domain ID for 802.11r                : 0xac34
Mobility Keepalive Interval                   : 10
Mobility Keepalive Count                      : 3
Mobility Control Message DSCP Value           : 7
Mobility Domain Member Count                  : 3

Link Status is Control Link Status : Data Link Status

Controllers configured in the Mobility Domain:

IP              Public IP        Group Name       Multicast IP     Link Status
------------------------------------------------------------------------------
9.9.9.2         -                default          0.0.0.0          UP   : UP
12.12.11.11     12.13.12.12      rasagna-grp                       DOWN : DOWN
27.0.0.1        23.0.0.1         test                              DOWN : DOWN

Switch Peer Group Name            : spg1
Switch Peer Group Member Count    : 0
Bridge Domain ID                  : 0
Multicast IP Address              : 0.0.0.0

Switch Peer Group Name            : pg
Switch Peer Group Member Count    : 1
Bridge Domain ID                  : 0
Multicast IP Address              : 0.0.0.0

IP              Public IP             Link Status
------------------------------------------------
9.7.136.10      9.7.136.10            DOWN : DOWN
```

Example: Choosing the Default Web Authentication Login Page

This example shows how to choose a default web authentication login page.

```
Controller# configure terminal
Controller(config)# parameter-map type webauth test
This operation will permanently convert all relevant authentication commands to their CPL
control-policy equivalents. As this conversion is irreversible and will
disable the conversion CLI 'authentication display [legacy|new-style]', you are strongly
```

```
advised to back up your current configuration before proceeding.
Do you wish to continue? [yes]: yes
Controller(config)# wlan wlan50
Controller(config-wlan)# shutdown
Controller(config-wlan)# security web-auth authentication-list test
Controller(config-wlan)# security web-auth parameter-map test
Controller(config-wlan)# no shutdown
Controller(config-wlan)# end
Controller# show running-config | section wlan50
wlan wlan50 50 wlan50
 security wpa akm cckm
 security wpa wpa1
 security wpa wpa1 ciphers aes
 security wpa wpa1 ciphers tkip
 security web-auth authentication-list test
 security web-auth parameter-map test
 session-timeout 1800
 no shutdown

Controller# show running-config | section parameter-map type webauth test
parameter-map type webauth test
 type webauth
```

Example: Choosing a Customized Web Authentication Login Page from an External Web Server

This example shows how to choose a customized web authentication login page from an external web server.

```
Controller# configure terminal
Controller(config)# parameter-map type webauth global
Controller(config-params-parameter-map)# virtual-ip ipv4 1.1.1.1
Controller(config-params-parameter-map)# parameter-map type webauth test
Controller(config-params-parameter-map)# type webauth
Controller(config-params-parameter-map)# redirect for-login http://9.1.0.100/login.html
Controller(config-params-parameter-map)# redirect portal ipv4 23.0.0.1
Controller(config-params-parameter-map)# end
Controller# show running-config | section parameter-map
parameter-map type webauth global
virtual-ip ipv4 1.1.1.1
parameter-map type webauth test
type webauth
redirect for-login http://9.1.0.100/login.html
redirect portal ipv4 23.0.0.1
security web-auth parameter-map rasagna-auth-map
security web-auth parameter-map test
```

Example: Assigning Login, Login Failure, and Logout Pages per WLAN

This example shows how to assign login, login failure and logout pages per WLAN.

```
Controller# configure terminal
Controller(config)# parameter-map type webauth test
Controller(config-params-parameter-map)# custom-page login device flash:loginsantosh.html
Controller(config-params-parameter-map)# custom-page login expired device
flash:loginexpire.html
Controller(config-params-parameter-map)# custom-page failure device flash:loginfail.html
Controller(config-params-parameter-map)# custom-page success device flash:loginsucess.html
Controller(config-params-parameter-map)# end
Controller# show running-config | section parameter-map type webauth test
 parameter-map type webauth test
 type webauth
 redirect for-login http://9.1.0.100/login.html
 redirect portal ipv4 23.0.0.1
 custom-page login device flash:loginsantosh.html
 custom-page success device flash:loginsucess.html
 custom-page failure device flash:loginfail.html
 custom-page login expired device flash:loginexpire.html
```

Example: Configuring AAA-Override

This example shows how to configure aaa-override.

```
Controller# configure terminal
Controller(config)# wlan fff
Controller(config-wlan)# aaa-override
Controller(config-wlan)# end
Controller# show running-config | section fff
 wlan fff 44 fff
 aaa-override
 shutdown
```

Example: Configuring Client Load Balancing

This example shows how to configure client load balancing.

```
Controller# configure terminal
Controller(config)# wlan fff
Controller(config-wlan)# shutdown
Controller(config-wlan)# mobility anchor 9.7.136.15
Controller(config-wlan)# mobility anchor 9.7.136.16
Controller(config-wlan)# no shutdown wlan
Controller(config-wlan)# end
Controller# show running-config | section fff
wlan fff 44 fff
 aaa-override
 shutdown
```

Example: Configuring Preauthentication ACL

This example shows how to configure preauthentication ACL.

```
Controller# configure terminal
Controller(config)# wlan fff
Controller(config-wlan)# shutdown
Controller(config-wlan)# ip access-group web preauthrule
Controller(config-wlan)# no shutdown
Controller(config-wlan)# end
Controller# show wlan name fff
```

Example: Configuring IOS ACL Definition

This example shows how to configure IOS ACL definition.

```
Controller# configure terminal
Controller(config)# ip access-list extended 102
Controller(config-ext-nacl)# permit udp any eq 8080 any
Controller(config-ext-nacl)# end
Controller# show access-lists 102
 Extended IP access list 102
    10 permit udp any eq 8080 any
```

Example: Configuring Webpassthrough

This example shows how to configure webpassthrough.

```
Controller# configure terminal
Controller(config)# parameter-map type webauth webparalocal
Controller(config-params-parameter-map)# type consent
```

```
Controller(config-params-parameter-map)# end
Controller# show running-config | section parameter-map type webauth test
 parameter-map type webauth test
 type webauth
 redirect for-login http://9.1.0.100/login.html
 redirect portal ipv4 23.0.0.1
```

Where to Go Next

Additional References for Guest Access

Related Documents

Related Topic	Document Title
Mobility CLI commands	*Mobility Command Reference, Cisco IOS XE 3SE (Cisco WLC 5700 Series)*
Mobility configuration	*Mobility Configuration Guide, Cisco IOS XE 3SE (Cisco WLC 5700 Series)*
Security CLI commands	*Security Command Reference, Cisco IOS Release 3SE (Cisco WLC 5700 Series)*
Configuring web-based authentication on the Catalyst 5700 Series Wireless Controller	*Security Configuration Guide, Cisco IOS Release 3SE (Cisco WLC 5700 Series)*

Standards and RFCs

Standard/RFC	Title
None	-

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

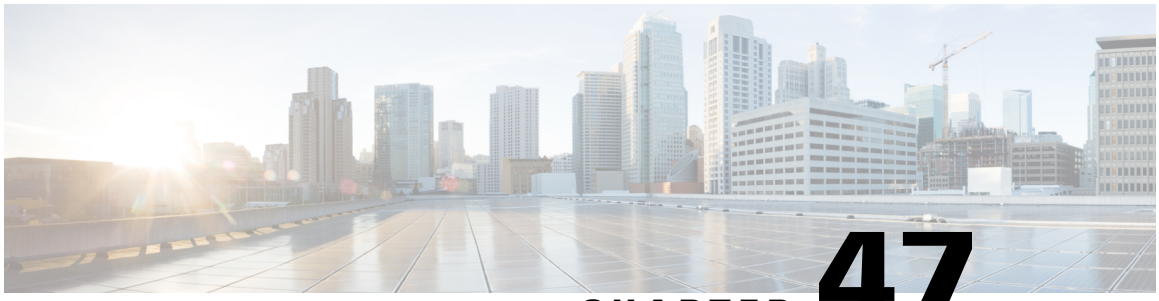
Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, 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 Guest Access

Releases	Feature Information
Cisco IOS XE Release 3.2SE	This feature was introduced.

PART VIII

Layer 2



CHAPTER 47

Configuring EtherChannels

This chapter contains the following topics:

Configuring EtherChannels

You can configure EtherChannels on Layer 2 and Layer 3 ports on the switch. EtherChannel provides fault-tolerant high-speed links between switches, routers, and servers. You can use Etherchannels 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.

Feature Information for EtherChannels

Feature Information for EtherChannels

Releases	Feature Information
	This feature was introduced.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 EtherChannels

The following are restrictions for EtherChannels:

- Layer 3 EtherChannels are not supported if running the LAN Base license feature set.

Information About EtherChannels

Port-Channel Interfaces

When you create an EtherChannel, a port-channel logical interface is involved:

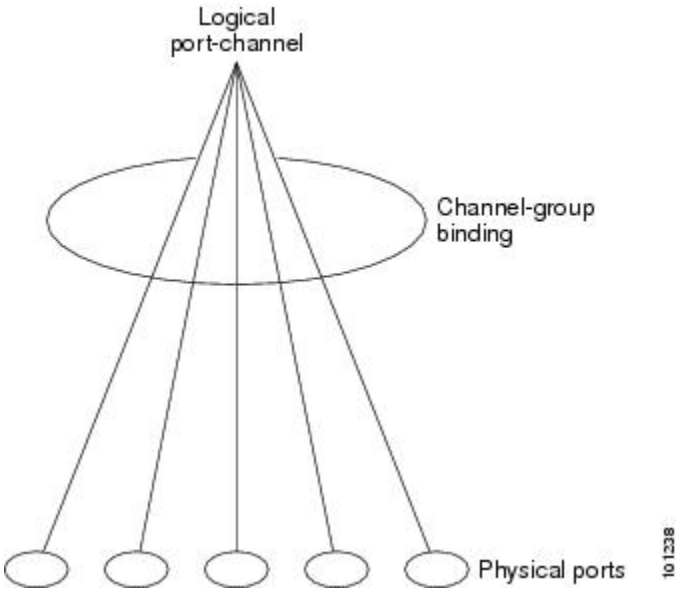
- With Layer 2 ports, use the **channel-group** interface configuration command to dynamically create the port-channel logical interface.

  You also can use the **interface port-channel** *port-channel-number* global configuration command to manually create the port-channel logical interface, but then you must use the **channel-group** *channel-group-number* command to bind the logical interface to a physical port. The *channel-group-number* can be the same as the *port-channel-number,* or you can use a new number. If you use a new number, the **channel-group** command dynamically creates a new port channel.

- With Layer 3 ports, you should manually create the logical interface by using the **interface port-channel** global configuration command followed by the **no switchport** interface configuration command. You then manually assign an interface to the EtherChannel by using the **channel-group** interface configuration command.

The **channel-group** command binds the physical port and the logical 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.

Figure 40: Relationship of Physical Ports, Logical Port Channels, and Channel Groups



After you configure an EtherChannel, configuration changes applied to the port-channel interface apply to all the physical ports assigned to the port-channel interface. Configuration changes applied to the physical port affect only the port where you apply the configuration. To change the parameters of all ports in an EtherChannel, apply configuration commands to the port-channel interface, for example, spanning-tree commands or commands to configure a Layer 2 EtherChannel as a trunk.

Related Topics

EtherChannel Configuration Guidelines, on page 927
Layer 2 EtherChannel Configuration Guidelines, on page 928
Layer 3 EtherChannel Configuration Guidelines, on page 929
Default EtherChannel Configuration, on page 925
EtherChannel Configuration Guidelines, on page 927
Layer 2 EtherChannel Configuration Guidelines, on page 928
Layer 3 EtherChannel Configuration Guidelines, on page 929
Default EtherChannel Configuration, on page 925

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. PAgP can be enabled on cross-stack EtherChannels.

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 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 controller 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 controller 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 controller 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.

Related Topics

Configuring the PAgP Learn Method and Priority, on page 931
EtherChannel Configuration Guidelines, on page 927
Layer 2 EtherChannel Configuration Guidelines, on page 928
Layer 3 EtherChannel Configuration Guidelines, on page 929
Default EtherChannel Configuration, on page 925
Monitoring EtherChannel, PAgP, and LACP Status, on page 933

PAgP Interaction with Virtual Switches and Dual-Active Detection

A virtual switch can be two or more core switches connected by virtual switch links (VSLs) that carry control and data traffic between them. One of the switches is in active mode. The others are in standby mode. For redundancy, remote switches are connected to the virtual switch by remote satellite links (RSLs).

If the VSL between two switches fails, one switch does not know the status of the other. Both switches could change to the active mode, causing a *dual-active situation* in the network with duplicate configurations (including duplicate IP addresses and bridge identifiers). The network might go down.

To prevent a dual-active situation, the core switches send PAgP protocol data units (PDUs) through the RSLs to the remote switches. The PAgP PDUs identify the active switch, and the remote switches forward the PDUs to core switches so that the core switches are in sync. If the active switch fails or resets, the standby switch takes over as the active switch. If the VSL goes down, one core switch knows the status of the other and does not change state.

PAgP Interaction with Other Features

The Dynamic Trunking Protocol (DTP) and the Cisco Discovery Protocol (CDP) send and receive packets over the physical ports in the EtherChannel. Trunk ports send and receive PAgP protocol data units (PDUs) on the lowest numbered VLAN.

In Layer 2 EtherChannels, the first port in the channel that comes up provides its MAC address to the EtherChannel. If this port is removed from the bundle, one of the remaining ports in the bundle provides its MAC address to the EtherChannel. For Layer 3 EtherChannels, the MAC address is allocated by the active controller as soon as the interface is created (through the **interface port-channel** global configuration command).

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 controllers to manage Ethernet channels between controllers 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 controller 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 105: 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.

Related Topics

Configuring Layer 2 EtherChannels, on page 929
EtherChannel Configuration Guidelines, on page 927
Layer 2 EtherChannel Configuration Guidelines, on page 928
Layer 3 EtherChannel Configuration Guidelines, on page 929
Default EtherChannel Configuration, on page 925

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. For Layer 3 EtherChannels, the MAC address is allocated by the active switch as soon as the interface is created through the **interface port-channel** global configuration command.

LACP sends and receives LACP PDUs only from ports that are up and have LACP enabled for the active or passive mode.

EtherChannel On Mode

EtherChannel **on** mode can be used to manually configure an EtherChannel. The **on** mode forces a port to join an EtherChannel without negotiations. The **on** mode can be useful if the remote device does not support PAgP or LACP. In the **on** mode, a usable EtherChannel exists only when the switches at both ends of the link are configured in the **on** mode.

Ports that are configured in the **on** mode in the same channel group must have compatible port characteristics, such as speed and duplex. Ports that are not compatible are suspended, even though they are configured in the **on** mode.

Caution You should use care when using the **on** mode. This is a manual configuration, and ports on both ends of the EtherChannel must have the same configuration. If the group is misconfigured, packet loss or spanning-tree loops can occur.

Load-Balancing and Forwarding Methods

EtherChannel balances the traffic load across the links in a channel by reducing part of the binary pattern formed from the addresses in the frame to a numerical value that selects one of the links in the channel. EtherChannel load-balancing can use several methods of load balancing that include MAC addresses or IP addresses, source or destination addresses, or both source and destination addresses. The selected mode applies to all EtherChannels configured on the switch. You configure the load-balancing and forwarding method by using the **port-channel load-balance** global configuration command.

Related Topics

Configuring EtherChannel Load-Balancing
EtherChannel Configuration Guidelines, on page 927
Layer 2 EtherChannel Configuration Guidelines, on page 928
Layer 3 EtherChannel Configuration Guidelines, on page 929
Default EtherChannel Configuration, on page 925

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.

Related Topics

Configuring EtherChannel Load-Balancing
EtherChannel Configuration Guidelines, on page 927
Layer 2 EtherChannel Configuration Guidelines, on page 928

Layer 3 EtherChannel Configuration Guidelines, on page 929
Default EtherChannel Configuration, on page 925

IP Address Forwarding

With source-IP address-based forwarding, when packets are forwarded to an EtherChannel, they are distributed across the ports in the EtherChannel based on the source-IP address of the incoming packet. Therefore, to provide load-balancing, packets from different IP addresses use different ports in the channel, but packets from the same IP address use the same port in the channel.

With destination-IP address-based forwarding, when packets are forwarded to an EtherChannel, they are distributed across the ports in the EtherChannel based on the destination-IP address of the incoming packet. Therefore, to provide load-balancing, packets from the same IP source address sent to different IP destination addresses could be sent on different ports in the channel. But packets sent from different source IP addresses to the same destination IP address are always sent on the same port in the channel.

With source-and-destination IP address-based forwarding, when packets are forwarded to an EtherChannel, they 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.

Related Topics

Configuring EtherChannel Load-Balancing
EtherChannel Configuration Guidelines, on page 927
Layer 2 EtherChannel Configuration Guidelines, on page 928
Layer 3 EtherChannel Configuration Guidelines, on page 929
Default EtherChannel Configuration, on page 925

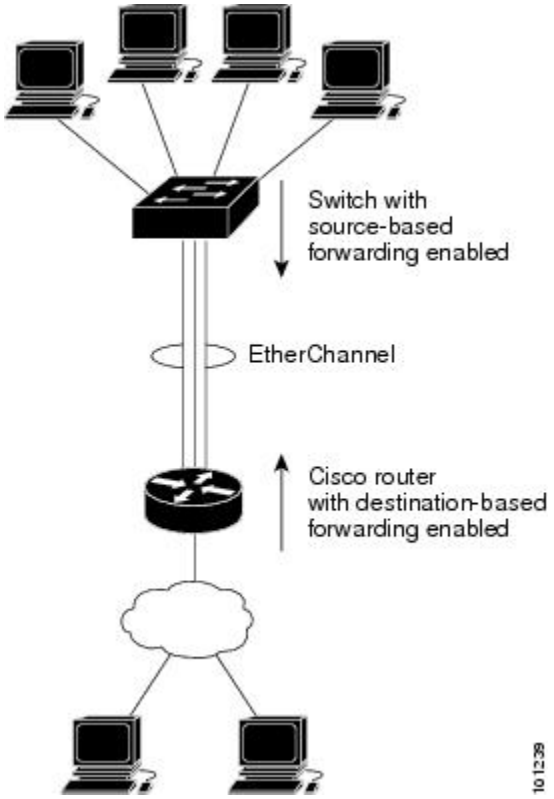
Load Balancing Advantages

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.

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.

Figure 41: Load Distribution and Forwarding Methods



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.

Related Topics

Configuring EtherChannel Load-Balancing
EtherChannel Configuration Guidelines, on page 927
Layer 2 EtherChannel Configuration Guidelines, on page 928
Layer 3 EtherChannel Configuration Guidelines, on page 929
Default EtherChannel Configuration, on page 925

Default EtherChannel Configuration

The default EtherChannel configuration is described in this table.

Table 106: 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.

Related Topics

Configuring Layer 2 EtherChannels, on page 929
EtherChannel Overview
EtherChannel Modes
EtherChannel Link Failover
LACP Modes, on page 921
Port-Channel Interfaces, on page 918
Port-Channel Interfaces, on page 918
Configuring EtherChannel Load-Balancing
Load-Balancing and Forwarding Methods, on page 923
MAC Address Forwarding, on page 923
IP Address Forwarding, on page 924
Load Balancing Advantages, on page 924
Configuring the PAgP Learn Method and Priority, on page 931
PAgP Learn Method and Priority, on page 920

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 48 EtherChannels on the switch or 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.
- If cross-stack EtherChannel is configured and the switch stack partitions, loops and forwarding issues can occur.

Related Topics

Configuring Layer 2 EtherChannels, on page 929
EtherChannel Overview

EtherChannel Modes
EtherChannel Link Failover
LACP Modes, on page 921
Port-Channel Interfaces, on page 918
Port-Channel Interfaces, on page 918
Configuring EtherChannel Load-Balancing
Load-Balancing and Forwarding Methods, on page 923
MAC Address Forwarding, on page 923
IP Address Forwarding, on page 924
Load Balancing Advantages, on page 924
Configuring the PAgP Learn Method and Priority, on page 931
PAgP Learn Method and Priority, on page 920

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.
- If you configure an EtherChannel from trunk ports, verify that the trunking mode (ISL or IEEE 802.1Q) is the same on all the trunks. Inconsistent trunk modes on EtherChannel ports can have unexpected results.
- An EtherChannel supports the same allowed range of VLANs on all the ports in a trunking Layer 2 EtherChannel. If the allowed range of VLANs is not the same, the ports do not form an EtherChannel even when PAgP is set to the **auto** or **desirable** mode.
- Ports with different spanning-tree path costs can form an EtherChannel if they are otherwise compatibly configured. Setting different spanning-tree path costs does not, by itself, make ports incompatible for the formation of an EtherChannel.

Related Topics

Configuring Layer 2 EtherChannels, on page 929
EtherChannel Overview
EtherChannel Modes
EtherChannel Link Failover
LACP Modes, on page 921
Port-Channel Interfaces, on page 918
Port-Channel Interfaces, on page 918
Configuring EtherChannel Load-Balancing
Load-Balancing and Forwarding Methods, on page 923
MAC Address Forwarding, on page 923
IP Address Forwarding, on page 924
Load Balancing Advantages, on page 924
Configuring the PAgP Learn Method and Priority, on page 931

PAgP Learn Method and Priority, on page 920

Layer 3 EtherChannel Configuration Guidelines

When configuring Layer 3 EtherChannels, follow this guideline:

- For Layer 3 EtherChannels, assign the Layer 3 address to the port-channel logical interface, not to the physical ports in the channel.

Related Topics

Configuring Layer 2 EtherChannels, on page 929
EtherChannel Overview
EtherChannel Modes
EtherChannel Link Failover
LACP Modes, on page 921
Port-Channel Interfaces, on page 918
Port-Channel Interfaces, on page 918
Configuring EtherChannel Load-Balancing
Load-Balancing and Forwarding Methods, on page 923
MAC Address Forwarding, on page 923
IP Address Forwarding, on page 924
Load Balancing Advantages, on page 924
Configuring the PAgP Learn Method and Priority, on page 931
PAgP Learn Method and Priority, on page 920

How to Configure EtherChannels

This section describes how to configure EtherChannels, PAgP, and LACP.

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:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(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:** `Controller(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:** `Controller(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:** `Controller(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 128. 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.

	Command or Action	Purpose
		• **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 6	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.

Related Topics

EtherChannel Overview
EtherChannel Modes
EtherChannel Link Failover
LACP Modes, on page 921
EtherChannel Configuration Guidelines, on page 927
Layer 2 EtherChannel Configuration Guidelines, on page 928
Layer 3 EtherChannel Configuration Guidelines, on page 929
Default EtherChannel Configuration, on page 925

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:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet 1/0/2`	Specifies the port for transmission, and enters interface configuration mode.
Step 3	**pagp learn-method physical-port** **Example:** `Controller(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:** `Controller(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:** `Controller(config-if)# end`	Returns to privileged EXEC mode.

Related Topics

PAgP Learn Method and Priority, on page 920
EtherChannel Configuration Guidelines, on page 927
Layer 2 EtherChannel Configuration Guidelines, on page 928
Layer 3 EtherChannel Configuration Guidelines, on page 929
Default EtherChannel Configuration, on page 925
Monitoring EtherChannel, PAgP, and LACP Status, on page 933

Monitoring EtherChannel, PAgP, and LACP Status

Display EtherChannel, PAgP, and LACP status using the commands described in this table.

Table 107: 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.

Related Topics

Configuring the PAgP Learn Method and Priority, on page 931
PAgP Learn Method and Priority, on page 920

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**:

```
Controller# configure terminal
Controller(config)# interface range gigabitethernet2/0/1 -2
Controller(config-if-range)# switchport mode access
Controller(config-if-range)# switchport access vlan 10
Controller(config-if-range)# channel-group 5 mode desirable non-silent
Controller(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**:

```
Controller# configure terminal
Controller(config)# interface range gigabitethernet2/0/1 -2
Controller(config-if-range)# switchport mode access
Controller(config-if-range)# switchport access vlan 10
Controller(config-if-range)# channel-group 5 mode active
Controller(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 2 and one port on stack member 3 as static-access ports in VLAN 10 to channel 5:

```
Controller# configure terminal
Controller(config)# interface range gigabitethernet2/0/4 -5
Controller(config-if-range)# switchport mode access
Controller(config-if-range)# switchport access vlan 10
Controller(config-if-range)# channel-group 5 mode active
Controller(config-if-range)# exit
Controller(config)# interface gigabitethernet3/0/3
Controller(config-if)# switchport mode access
Controller(config-if)# switchport access vlan 10
Controller(config-if)# channel-group 5 mode active
Controller(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:

```
Controller# configure terminal
Controller(config)# interface port-channel 5
Controller(config-if)# no switchport
Controller(config-if)# ip address 172.10.20.10 255.255.255.0
Controller(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**:

```
Controller# configure terminal
Controller(config)# interface range gigabitethernet2/0/1 -2
Controller(config-if-range)# no ip address
Controller(config-if-range)# no switchport
Controller(config-if-range)# channel-group 5 mode active
Controller(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:

```
Controller# configure terminal
Controller(config)# interface range gigabitethernet2/0/4 -5
Controller(config-if-range)# no ip address
Controller(config-if-range)# no switchport
Controller(config-if-range)# channel-group 7 mode active
Controller(config-if-range)# exit
Controller(config)# interface gigabitethernet3/0/3
Controller(config-if)# no ip address
Controller(config-if)# no switchport
Controller(config-if)# channel-group 7 mode active
Controller(config-if)# exit
```

Additional References for EtherChannels

Related Documents

Related Topic	Document Title
Layer 2 command reference	*Layer 2 Command Reference (Cisco WLC 5700 Series)*

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

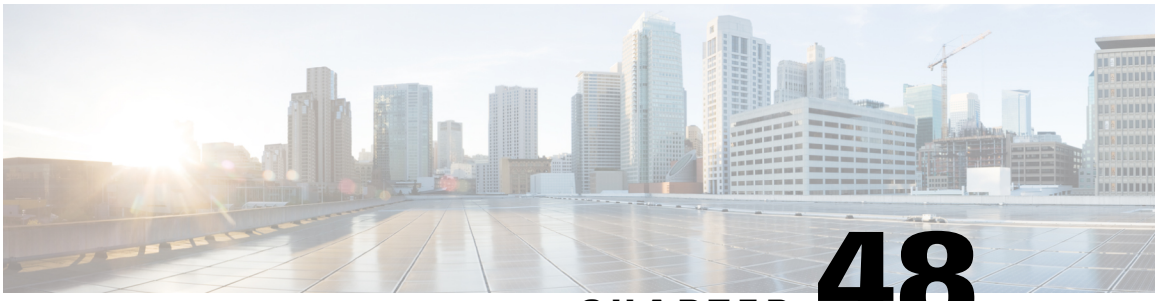
Technical Assistance

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 Information for EtherChannels

Feature Information for EtherChannels

Releases	Feature Information
	This feature was introduced.



CHAPTER 48

Configuring FlexLinks and the MAC Address-Table Move Update Feature

This chapter contains the following topics:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 FlexLinks and MAC Address-Table Move Update

The following are restrictions for configuring FlexLinks and the MAC Address-Table Move Update feature:

- FlexLinks are supported only on Layer 2 ports and port channels, not on VLANs or on Layer 3 ports.
- You can configure up to 16 backup links.
- You can configure only one FlexLinks backup link for any active link, and it must be a different interface from the active interface.

- An interface can belong to only one FlexLinks pair. An interface can be a backup link for only one active link. An active link cannot belong to another FlexLinks pair.
- Neither of the links can be a port that belongs to an EtherChannel. However, you can configure two port channels (EtherChannel logical interfaces) as FlexLinks, and you can configure a port channel and a physical interface as FlexLinks, with either the port channel or the physical interface as the active link.
- A backup link does not have to be the same type (Gigabit Ethernet or port channel) as the active link. However, you should configure both FlexLinks with similar characteristics so that there are no loops or changes in behavior if the standby link begins to forward traffic.
- STP is disabled on FlexLinks ports. A FlexLinks port does not participate in STP, even if the VLANs present on the port are configured for STP. When STP is not enabled, be sure that there are no loops in the configured topology.

Related Topics

Configuring a Preemption Scheme for a Pair of FlexLinks, on page 944
Configuring FlexLinks, on page 943
Configuring FlexLinks: Examples, on page 950
Configuring VLAN Load Balancing on FlexLinks, on page 946
Configuring VLAN Load Balancing on FlexLinks: Examples, on page 950
Configuring a Switch to Obtain and Process MAC Address-Table Move Update Messages, on page 948
Configuring MAC Address-Table Move Update , on page 947
Configuring the MAC Address-Table Move Update: Examples, on page 952

Information About FlexLinks and the MAC Address-Table Move Update

This chapter describes how to configure FlexLinks, a pair of interfaces on the controller that provide a mutual backup. It also describes how to configure the MAC address-table move update feature, also referred to as the FlexLinks bidirectional fast convergence feature.

FlexLinks

FlexLinks are a pair of a Layer 2 interfaces (controller ports or port channels) where one interface is configured to act as a backup to the other. The feature provides an alternative solution to the Spanning Tree Protocol (STP). Users can disable STP and still retain basic link redundancy. FlexLinks are typically configured in service provider or enterprise networks where customers do not want to run STP on the controller. If the controller is running STP, FlexLinks are not necessary because STP already provides link-level redundancy or backup.

You configure FlexLinks on one Layer 2 interface (the active link) by assigning another Layer 2 interface as the FlexLinks or backup link. On controllers, the FlexLinks can be on the same controller or on another controller 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 FlexLinks interfaces.

Related Topics

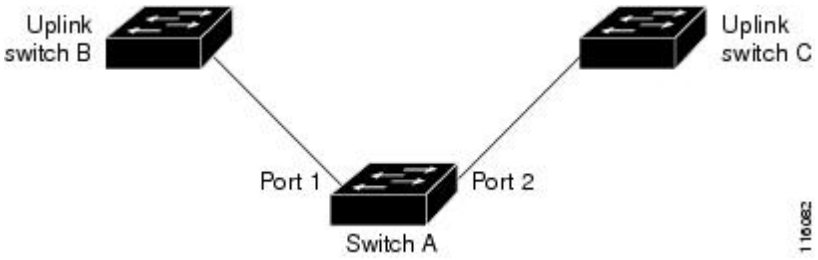
Configuring a Preemption Scheme for a Pair of FlexLinks, on page 944
Configuring FlexLinks, on page 943
Configuring FlexLinks: Examples, on page 950

FlexLinks 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 FlexLinks, only one of the interfaces is forwarding traffic; the other is in standby mode. If port 1 is the active link, it begins forwarding traffic between port 1 and switch B; the link between port 2 (the backup link) and switch C is not forwarding traffic. If port 1 goes down, port 2 comes up and starts forwarding traffic to switch C. When port 1 comes back up, it goes into standby mode and does not forward traffic; port 2 continues forwarding traffic.

You can also configure a preemption function, specifying the preferred port for forwarding traffic. For example, you can configure the FlexLinks pair with preemption mode. In the scenario shown, when port 1 comes back up and has more bandwidth than port 2, port 1 begins forwarding traffic after 60 seconds. Port 2 becomes the standby port. You do this by entering the **switchport backup interface preemption mode bandwidth** and **switchport backup interface preemption delay** interface configuration commands.

Figure 42: FlexLinks Configuration Example



If a primary (forwarding) link goes down, a trap notifies the network management stations. If the standby link goes down, a trap notifies the users.

FlexLinks are supported only on Layer 2 ports and port channels, not on VLANs or on Layer 3 ports.

Related Topics

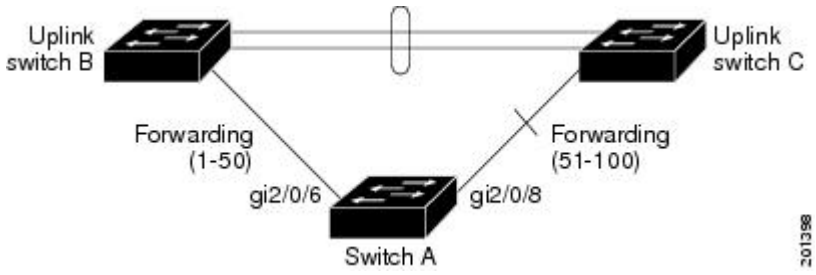
Configuring a Preemption Scheme for a Pair of FlexLinks, on page 944
Configuring FlexLinks, on page 943

VLAN FlexLinks Load Balancing and Support

VLAN FlexLinks load balancing allows users to configure a FlexLinks pair so that both ports simultaneously forward the traffic for some mutually exclusive VLANs. For example, if FlexLinks ports are configured for 1 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 FlexLinks pair can be used for load balancing. FlexLinks VLAN load balancing does not impose any restrictions on uplink controllers.

The following figure displays a VLAN FlexLinks load-balancing configuration.

Figure 43: VLAN Flex Links Load-Balancing Configuration Example



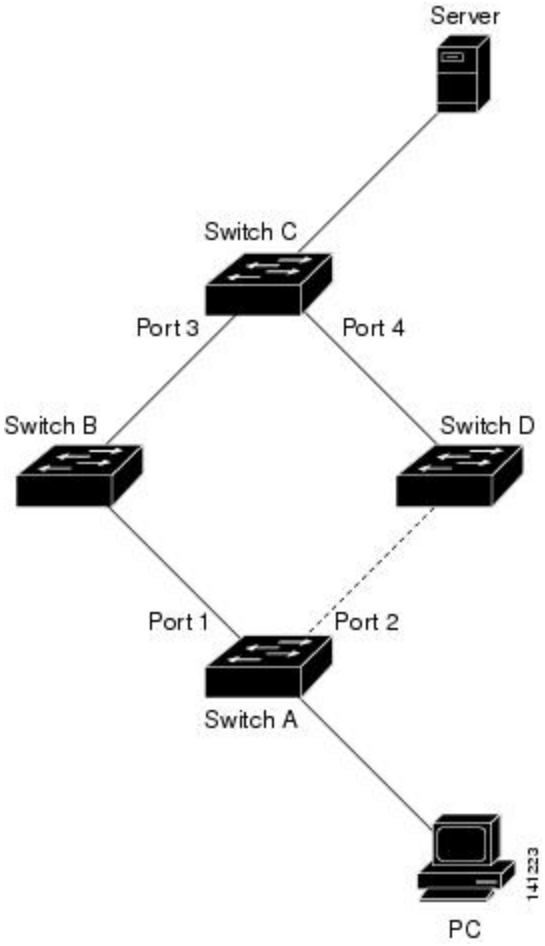
MAC Address-Table Move Update

The MAC address-table move update feature allows the controller to provide rapid bidirectional convergence when a primary (forwarding) link goes down and the standby link begins forwarding traffic.

In the following figure, controller A is an access controller, and ports 1 and 2 on controller A are connected to uplink controllers B and D through a FlexLinks pair. Port 1 is forwarding traffic, and port 2 is in the backup

state. Traffic from the PC to the server is forwarded from port 1 to port 3. The MAC address of the PC has been learned on port 3 of controller C. Traffic from the server to the PC is forwarded from port 3 to port 1.

Figure 44: MAC Address-Table Move Update Example



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 controller 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 controllers, and port 1 goes down, port 2 starts forwarding traffic from the PC to the server. The controller sends a MAC address-table move update packet from port 2. Controller 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 controller, controller A, to *send* MAC address-table move update messages. You can also configure the uplink controllers B, C, and D to *get* and process the MAC address-table move update messages. When controller C gets a MAC address-table move update message from controller A, switch C learns the MAC address of the PC on port 4. Controller C updates the MAC address table, including the forwarding table entry for the PC.

Controller A does not need to wait for the MAC address-table update. The controller detects a failure on port 1 and immediately starts forwarding server traffic from port 2, the new forwarding port. This change occurs in

100 milliseconds (ms). The PC is directly connected to controller A, and the connection status does not change. Controller A does not need to update the PC entry in the MAC address table.

Related Topics

Configuring a Switch to Obtain and Process MAC Address-Table Move Update Messages, on page 948
Configuring MAC Address-Table Move Update , on page 947
Configuring the MAC Address-Table Move Update: Examples, on page 952

VLAN Load Balancing Configuration Guidelines

Follow these guidelines to configure VLAN load balancing on the FlexLinks feature:

- For FlexLinks, VLAN load balancing, you must choose the preferred VLANs on the backup interface.
- You cannot configure a preemption mechanism and VLAN load balancing for the same FlexLinks pair.

Follow these guidelines to configure MAC address-table move update feature:

- You can enable and configure this feature on the access switch to *send* the MAC address-table move updates.
- You can enable and configure this feature on the uplink switches to *get* the MAC address-table move updates.

Related Topics

Configuring VLAN Load Balancing on FlexLinks, on page 946
Configuring VLAN Load Balancing on FlexLinks: Examples, on page 950

Default FlexLinks and MAC Address-Table Move Update Configuration

- FlexLinks 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 controller

Related Topics

Configuring a Preemption Scheme for a Pair of FlexLinks, on page 944
Configuring FlexLinks, on page 943
Configuring FlexLinks: Examples, on page 950

How to Configure FlexLinks and the MAC Address-Table Move Update Feature

Configuring FlexLinks

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:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(conf)# interface gigabitethernet1/0/1`	Specifies the interface, and enter interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface). The port-channel range is 1 to 48.
Step 3	**switchport backup interface** *interface-id* **Example:** `Controller(conf-if)# switchport backup interface gigabitethernet1/0/2`	Configures a physical Layer 2 interface (or port channel) as part of a FlexLinks pair with the interface. When one link is forwarding traffic, the other interface is in standby mode.
Step 4	**end** **Example:** `Controller(conf-if)# end`	Returns to privileged EXEC mode.

Related Topics

FlexLinks, on page 938
Default FlexLinks and MAC Address-Table Move Update Configuration, on page 942

Restrictions for Configuring FlexLinks and MAC Address-Table Move Update, on page 937
Configuring FlexLinks: Examples, on page 950
FlexLinks Configuration, on page 939
Monitoring FlexLinks and the MAC Address-Table Move Update, on page 949
Configuring FlexLinks: Examples, on page 950

Configuring a Preemption Scheme for a Pair of FlexLinks

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:** `Controller# configure terminal`	Enters global configuration mode
Step 2	**interface** *interface-id* **Example:** `Controller(conf)# interface gigabitethernet1/0/1`	Specifies the interface, and enter interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface). The port-channel range is 1 to 48.
Step 3	**switchport backup interface** *interface-id* **Example:** `Controller(conf-if)# switchport backup interface gigabitethernet1/0/2`	Configures a physical Layer 2 interface (or port channel) as part of a FlexLinks pair with the interface. When one link is forwarding traffic, the other interface is in standby mode.
Step 4	**switchport backup interface** *interface-id* **preemption mode** [**forced**	**bandwidth**	**off**]	Configures a preemption mechanism and delay for a FlexLinks interface pair. You can configure the preemption as:

	Command or Action	Purpose
	Example: `Controller(conf-if)# switchport backup interface gigabitethernet1/0/2 preemption mode forced`	• **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:** `Controller(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:** `Controller(conf-if)# end`	Returns to privileged EXEC mode.
Step 7	**show interface** [*interface-id*] **switchport backup** **Example:** `Controller# show interface gigabitethernet1/0/2 switchport backup`	Verifies the configuration.
Step 8	**copy running-config startup config** **Example:** `Controller# copy running-config startup config`	(Optional) Saves your entries in the switch startup configuration file.

Related Topics

FlexLinks, on page 938
Default FlexLinks and MAC Address-Table Move Update Configuration, on page 942
Restrictions for Configuring FlexLinks and MAC Address-Table Move Update, on page 937
Configuring FlexLinks: Examples, on page 950
FlexLinks Configuration, on page 939
Monitoring FlexLinks and the MAC Address-Table Move Update, on page 949
Configuring FlexLinks: Examples, on page 950

Configuring VLAN Load Balancing on FlexLinks

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:** `Controller# 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 FlexLinks 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.

Related Topics

VLAN Load Balancing Configuration Guidelines, on page 942
Restrictions for Configuring FlexLinks and MAC Address-Table Move Update, on page 937
Configuring VLAN Load Balancing on FlexLinks: Examples, on page 950
Configuring VLAN Load Balancing on FlexLinks: Examples, on page 950

Monitoring FlexLinks and the MAC Address-Table Move Update, on page 949

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:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Switch (config)# interface gigabitethernet1/0/1`	Specifies the interface, and enter interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface). The port-channel range is 1 to 48.
Step 3	Use one of the following: • **switchport backup interface** *interface-id* • **switchport backup interface** *interface-id* **mmu primary vlan** *vlan-id* **Example:** `Controller(config-if)# switchport backup interface gigabitethernet0/2 mmu primary vlan 2`	Configures a physical Layer 2 interface (or port channel), as part of a FlexLinks pair with the interface. The MAC address-table move update VLAN is the lowest VLAN ID on the interface. 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.

	Command or Action	Purpose
Step 4	**end** **Example:** `Controller(config-if)# end`	Returns to global configuration mode.
Step 5	**mac address-table move update transmit** **Example:** `Controller(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.

Related Topics

Configuring the MAC Address-Table Move Update: Examples, on page 952
Monitoring FlexLinks and the MAC Address-Table Move Update, on page 949
MAC Address-Table Move Update, on page 940
Restrictions for Configuring FlexLinks and MAC Address-Table Move Update, on page 937
Configuring the MAC Address-Table Move Update: Examples, on page 952

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**	Enters global configuration mode

	Command or Action	Purpose
	Example: `Controller# configure terminal`	
Step 2	**mac address-table move update receive** **Example:** `Switch (config)# mac address-table move update receive`	Enables the controller to obtain and process the MAC address-table move updates.
Step 3	**end** **Example:** `Switch (config)# end`	Returns to privileged EXEC mode.

Related Topics

Monitoring FlexLinks and the MAC Address-Table Move Update, on page 949
Configuring the MAC Address-Table Move Update: Examples, on page 952
MAC Address-Table Move Update, on page 940
Restrictions for Configuring FlexLinks and MAC Address-Table Move Update, on page 937
Configuring the MAC Address-Table Move Update: Examples, on page 952

Monitoring FlexLinks and the MAC Address-Table Move Update

Command	Purpose
show interface [*interface-id*] **switchport backup**	Displays the FlexLinks backup interface configured for an interface or all the configured FlexLinks and the state of each active and backup interface (up or standby mode).
show mac address-table move update	Displays the MAC address-table move update information on the controller

Related Topics

Configuring a Preemption Scheme for a Pair of FlexLinks, on page 944
Configuring FlexLinks, on page 943

Configuration Examples for FlexLinks

Configuring FlexLinks: Examples

This example shows how to verify the configuration after you configure an interface with a backup interface:

```
Controller# 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:

```
Controller# show interface switchport backup detail

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
```

Related Topics

Configuring a Preemption Scheme for a Pair of FlexLinks, on page 944

Configuring FlexLinks, on page 943

FlexLinks, on page 938

Default FlexLinks and MAC Address-Table Move Update Configuration, on page 942

Restrictions for Configuring FlexLinks and MAC Address-Table Move Update, on page 937

Configuring a Preemption Scheme for a Pair of FlexLinks, on page 944

Configuring FlexLinks, on page 943

Configuring VLAN Load Balancing on FlexLinks: Examples

In the following example, VLANs 1 to 50, 60, and 100 to 120 are configured on the controller:

```
Controller(config)# interface gigabitethernet 2/0/6
Controller(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.

```
Controller# 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 FlexLinks interface goes down (LINK_DOWN), VLANs preferred on this interface are moved to the peer interface of the FlexLinks pair. In this example, if interface Gi2/0/6 goes down, Gi2/0/8 carries all VLANs of the FlexLinks pair.

```
Controller# 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 FlexLinks interface comes up, VLANs preferred on this interface are blocked on the peer interface and moved to the forwarding state on the interface that has just come up. In this example, if interface 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.

```
Controller# 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

Controller# 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
```

Related Topics

Configuring VLAN Load Balancing on FlexLinks, on page 946
VLAN Load Balancing Configuration Guidelines, on page 942
Restrictions for Configuring FlexLinks and MAC Address-Table Move Update, on page 937
Configuring VLAN Load Balancing on FlexLinks, on page 946

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:

```
Controller# 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
```

Related Topics

Configuring MAC Address-Table Move Update , on page 947
Configuring a Switch to Obtain and Process MAC Address-Table Move Update Messages, on page 948
Configuring a Switch to Obtain and Process MAC Address-Table Move Update Messages, on page 948
Configuring MAC Address-Table Move Update , on page 947
MAC Address-Table Move Update, on page 940
Restrictions for Configuring FlexLinks and MAC Address-Table Move Update, on page 937

Additional References for FlexLinks and MAC Address-Table Move Update

Related Documents

Related Topic	Document Title
Layer 2 command reference	*Layer 2 Command Reference (Cisco WLC 5700 Series)*

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

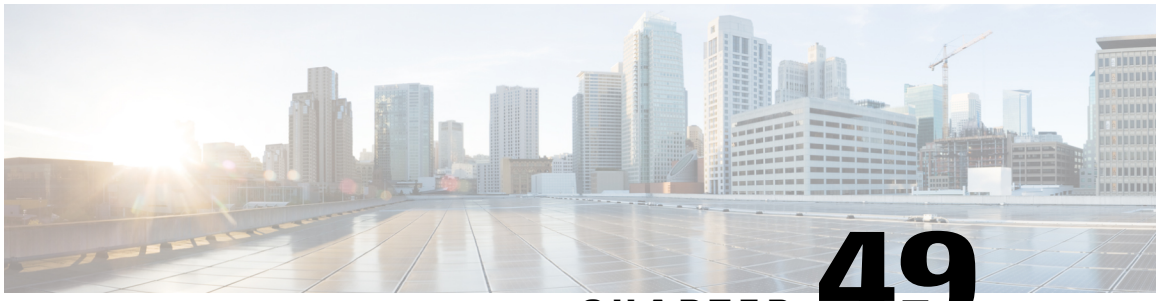
Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, 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 Information for Flex Links and MAC Address-Table Move Update

Releases	Feature Information
	This feature was introduced.

PART IX

WLAN



CHAPTER 49

Configuring WLANs

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see 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 document.

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 WLANs

- You can associate up to 16 WLANs with each access point group and assign specific access points to each group. Each access point advertises only the enabled WLANs that belong to its access point group. The access point (AP) does not advertise disabled WLANs in its access point group or WLANs that belong to another group.
- We recommend that you assign one set of VLANs for WLANs and a different set of VLANs for management interfaces to ensure that controllers properly route VLAN traffic.

- The controller uses different attributes to differentiate between WLANs with the same Service Set Identifier (SSID).
  - WLANs with the same SSID and same Layer 2 policy cannot be created if the WLAN ID is lower than 17.
  - Two WLANs with IDs that are greater than 17 and that have the same SSID and same Layer 2 policy is allowed if WLANs are added in different AP groups.

Note This requirement ensures that clients never detect the SSID present on the same access point radio.

Restrictions for WLANs

The following restrictions apply when configuring WLANs:

- Peer-to-peer blocking does not apply to multicast traffic.
- You can configure a maximum of 12000 clients.
- The WLAN name and SSID can have up to 32 characters. Spaces are not allowed in the WLAN profile name and SSID.
- You cannot map a WLAN to VLAN0, and you cannot map VLANs 1002 to 1006.
- Dual stack clients with Static-IPv4 address is not supported.
- When creating a WLAN with the same SSID, you must create a unique profile name for each WLAN.

Information About WLANs

This feature enables you to control up to 512 WLANs for lightweight access points. Each WLAN has a separate WLAN ID, a separate profile name, and a WLAN SSID. All controllers publish up to 16 WLANs to each connected access point, but you can create up to the maximum number of WLANs supported and then selectively publish these WLANs (using access point groups) to different access points to better manage your wireless network.

You can configure WLANs with different SSIDs or with the same SSID. An SSID identifies the specific wireless network that you want the controller to access.

Band Selection

Band selection enables client radios that are capable of dual-band (2.4- and 5-GHz) operation to move to a less congested 5-GHz access point. The 2.4-GHz band is often congested. Clients on this band typically experience interference from Bluetooth devices, microwave ovens, and cordless phones as well as co-channel interference from other access points because of the 802.11b/g limit of three nonoverlapping channels. To prevent these sources of interference and improve overall network performance, you can configure band selection on the controller.

Band selection works by regulating probe responses to clients. It makes 5-GHz channels more attractive to clients by delaying probe responses to clients on 2.4-GHz channels.

Related Topics

Configuring Advanced WLAN Properties (CLI), on page 966

Off-Channel Scanning Defer

In deployments with certain power-save clients, you sometimes need to defer the Radio Resource Management's (RRM) normal off-channel scanning to avoid missing critical information from low-volume clients (for example, medical devices that use power-save mode and periodically send telemetry information). This feature improves the way that Quality of Service (QoS) interacts with the RRM scan defer feature.

You can use a client's Wi-Fi Multimedia (WMM) UP marking to configure the access point to defer off-channel scanning for a configurable period of time if it receives a packet marked UP.

Off-Channel Scanning Defer is essential to the operation of RRM, which gathers information about alternate channel choices such as noise and interference. Additionally, Off-Channel Scanning Defer is responsible for rogue detection. Devices that need to defer Off-Channel Scanning Defer should use the same WLAN as often as possible. If there are many of these devices (and the possibility exists that Off-Channel Defer scanning could be completely disabled by the use of this feature), you should implement an alternative to local AP Off-Channel Scanning Defer, such as monitoring access points, or other access points in the same location that do not have this WLAN assigned.

You can assign a QoS policy (bronze, silver, gold, and platinum) to a WLAN to affect how packets are marked on the downlink connection from the access point regardless of how they were received on the uplink from the client. UP=1,2 is the lowest priority, and UP=0,3 is the next higher priority. The marking results of each QoS policy are as follows:

- Bronze marks all downlink traffic to UP= 1.
- Silver marks all downlink traffic to UP= 0.
- Gold marks all downlink traffic to UP=4.
- Platinum marks all downlink traffic to UP=6.

Related Topics

Configuring Advanced WLAN Properties (CLI), on page 966

DTIM Period

In 802.11a/n and 802.11b/g/n networks, lightweight access points broadcast a beacon at regular intervals, which coincides with the Delivery Traffic Indication Map (DTIM). After the access point broadcasts the beacon, it transmits any buffered broadcast and multicast frames based on the value set for the DTIM period. This feature allows power-saving clients to wake up at the appropriate time if they are expecting broadcast or multicast data.

Typically, the DTIM value is set to 1 (to transmit broadcast and multicast frames after every beacon) or 2 (to transmit after every other beacon). For instance, if the beacon period of the 802.11a/n or 802.11b/g/n network is 100 ms and the DTIM value is set to 1, the access point transmits buffered broadcast and multicast frames 10 times per second. If the beacon period is 100 ms and the DTIM value is set to 2, the access point transmits

buffered broadcast and multicast frames 5 times per second. Either of these settings are suitable for applications, including Voice Over IP (VoIP), that expect frequent broadcast and multicast frames.

However, the DTIM value can be set as high as 255 (to transmit broadcast and multicast frames after every 255th beacon) if all 802.11a/n or 802.11b/g/n clients have power save enabled. Because the clients have to listen only when the DTIM period is reached, they can be set to listen for broadcasts and multicasts less frequently which results in a longer battery life. For example, if the beacon period is 100 ms and you set the DTIM value to 100, the access point transmits buffered broadcast and multicast frames once every 10 seconds. This rate allows the power-saving clients to sleep longer before they have to wake up and listen for broadcasts and multicasts, which results in a longer battery life.

Note A beacon period, which is specified in milliseconds on the controller, is converted internally by the software to 802.11 Time Units (TUs), where 1 TU = 1.024 milliseconds. On Cisco's 802.11n access points, this value is rounded to the nearest multiple of 17 TUs. For example, a configured beacon period of 100 ms results in an actual beacon period of 104 ms.

Many applications cannot tolerate a long time between broadcast and multicast messages, which results in poor protocol and application performance. We recommend that you set a low DTIM value for 802.11a/n and 802.11b/g/n networks that support such clients.

Related Topics

Configuring Advanced WLAN Properties (CLI), on page 966

Session Timeout

You can configure a WLAN with a session timeout. The session timeout is the maximum time for a client session to remain active before requiring reauthorization.

Related Topics

Configuring Advanced WLAN Properties (CLI), on page 966

Cisco Client Extensions

The Cisco Client Extensions (CCX) software is licensed to manufacturers and vendors of third-party client devices. The CCX code resident on these clients enables them to communicate wirelessly with Cisco access points and to support Cisco features that other client devices do not, including those features that are related to increased security, enhanced performance, fast roaming, and power management.

- The software supports CCX versions 1 through 5, which enables controllers and their access points to communicate wirelessly with third-party client devices that support CCX. CCX support is enabled automatically for every WLAN on the controller and cannot be disabled. However, you can configure Aironet information elements (IEs).
- If Aironet IE support is enabled, the access point sends an Aironet IE 0x85 (which contains the access point name, load, number of associated clients, and so on) in the beacon and probe responses of this WLAN, and the controller sends Aironet IEs 0x85 and 0x95 (which contains the management IP address of the controller and the IP address of the access point) in the reassociation response if it receives Aironet IE 0x85 in the reassociation request.

Related Topics

Configuring Advanced WLAN Properties (CLI), on page 966

Peer-to-Peer Blocking

Peer-to-peer blocking is applied to individual WLANs, and each client inherits the peer-to-peer blocking setting of the WLAN to which it is associated. Peer-to-Peer enables you to have more control over how traffic is directed. For example, you can choose to have traffic bridged locally within the controller, dropped by the controller, or forwarded to the upstream VLAN.

Peer-to-peer blocking is supported for clients that are associated with the local switching WLAN.

Related Topics

Configuring Advanced WLAN Properties (CLI), on page 966

Diagnostic Channel

You can choose a diagnostic channel to troubleshoot why the client is having communication problems with a WLAN. You can test the client and access points to identify the difficulties that the client is experiencing and allow corrective measures to be taken to make the client operational on the network. You can use the controller GUI or CLI to enable the diagnostic channel, and you can use the controller CLI to run the diagnostic tests.

Note We recommend that you enable the diagnostic channel feature only for nonanchored SSIDs that use the management interface.

Related Topics

Configuring Advanced WLAN Properties (CLI), on page 966

Client Count Per WLAN

You can set a limit to the number of clients that can connect to a WLAN, which is useful in scenarios where you have a limited number of clients that can connect to a controller. For example, consider a scenario where the controller can serve up to 256 clients on a WLAN and these clients can be shared between enterprise users (employees) and guest users. You can set a limit on the number of guest clients that can access a given WLAN. The number of clients that you can configure for each WLAN depends on the platform that you are using.

Related Topics

Configuring Advanced WLAN Properties (CLI), on page 966

Per-WLAN RADIUS Source Support

By default, the controller sources all RADIUS traffic from the IP address on its management interface, which means that even if a WLAN has specific RADIUS servers configured instead of the global list, the identity used is the management interface IP address.

If you want to filter WLANs, you can use the callStationID that is set by RFC 3580 to be in the APMAC:SSID format. You can also extend the filtering on the authentication server to be on a per-WLAN source interface by using the NAS-IP-Address attribute.

When you enable the per-WLAN RADIUS source support, the controller sources all RADIUS traffic for a particular WLAN by using the dynamic interface that is configured. Also, RADIUS attributes are modified accordingly to match the identity. This feature virtualizes the controller on the per-WLAN RADIUS traffic, where each WLAN can have a separate layer 3 identity. This feature is useful in ACS Network Access Restrictions and Network Access Profiles.

You can be combine per-WLAN RADIUS source support with the normal RADIUS traffic source and some WLANs that use the management interface and others using the per-WLAN dynamic interface as the address source.

How to Configure WLANs

Creating WLANs (CLI)

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan-name wlan-id* [*ssid*]
3. **end**
4. (Optional) **show wlan summary**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *wlan-name wlan-id* [*ssid*] **Example:** `Controller(config)# wlan mywlan 34 mywlan-ssid`	Specifies the WLAN name and ID: • For the *wlan-name*, enter the profile name. The range is from 1 to 32 alphanumeric characters. • For the *wlan-id*, enter the WLAN ID. The range is from 1 to 512. • For the *ssid*, enter the Service Set Identifier (SSID) for this WLAN. If the SSID is not specified, the WLAN profile name is set as the SSID. **Note** By default, the WLAN is disabled.

	Command or Action	Purpose
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.
Step 4	**show wlan summary** **Example:** `Controller# show wlan summary`	(Optional) Displays a summary of WLANs that are created on the device.

Deleting WLANs (CLI)

SUMMARY STEPS

1. **configure terminal**
2. **no wlan** *wlan-name wlan-id ssid*
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**no wlan** *wlan-name wlan-id ssid* **Example:** `Controller(config)# no wlan test2`	Deletes the WLAN. The arguments are as follows: • The *wlan-name* is the WLAN profile name. • The *wlan-id* is the WLAN ID. • The *ssid* is the WLAN SSID name configured for the WLAN.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Searching WLANs (CLI)

SUMMARY STEPS

1. **show wlan summary**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show wlan summary** **Example:** `Controller# show wlan summary`	Displays the list of all WLANs configured on the device. You can search for the WLAN in the output.

```
Controller# show wlan summary
Number of WLANs: 4

WLAN Profile Name                     SSID                                VLAN Status
------------------------------------------------------------------------------------
1    test1                             test1-ssid                          137  UP
3    test2                             test2-ssid                          136  UP
2    test3                             test3-ssid                          1    UP
45   test4                             test4-ssid                          1    DOWN
```

You can also use wild cards to search WLANs. For example **show wlan summary include** | *variable*. Where variable is any search string in the output.

```
Controller# show wlan summary | include test-wlan-ssid
1    test-wlan                       test-wlan-ssid                    137   UP
```

Configuring General WLAN Properties (CLI)

You can configure the following properties:

- Media stream
- Broadcast SSID
- Call Snooping
- Radio
- Interface
- Status

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan-name*
3. **shutdown**
4. **media-stream multicast-direct**
5. **broadcast-ssid**
6. **call-snoop**
7. **radio** {**all** | **dot11a** | **dot11ag** | **dot11bg** | **dot11g**}
8. **client vlan** *vlan-identifier*
9. **ip multicast vlan** *vlan-name*
10. **no shutdown**
11. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *wlan-name* **Example:** `Controller# wlan test4`	Enters the WLAN configuration submode. The *wlan-name* is the profile name of the configured WLAN.
Step 3	**shutdown** **Example:** `Controller# shutdown`	Disables the WLAN before configuring the parameters.
Step 4	**media-stream multicast-direct** **Example:** `Controller(config-wlan)# media-stream multicast-direct`	Enables multicast VLANs on this WLAN.
Step 5	**broadcast-ssid** **Example:** `Controller(config-wlan)# broadcast-ssid`	Broadcasts the SSID for this WLAN. This field is enabled by default.
Step 6	**call-snoop** **Example:** `Controller(config-wlan)# call-snoop`	Enables call-snooping support.

	Command or Action	Purpose
Step 7	**radio** {**all** | **dot11a** | **dot11ag** | **dot11bg** | **dot11g**} **Example:** `Controller# radio all`	Enables radios on the WLAN. The keywords are as follows: • **all—**Configures the WLAN on all radio bands. • **dot1a—**Configures the WLAN on only 802.11a radio bands. • **dot11g—**Configures the WLAN on 802.11ag radio bands. • **dot11bg—**Configures the WLAN on only 802.11b/g radio bands (only 802.11b if 802.11g is disabled). • **dot11ag—** Configures the wireless LAN on 802.11g radio bands only.
Step 8	**client vlan** *vlan-identifier* **Example:** `Controller# client vlan test-vlan`	Enables an interface group on the WLAN. The arguments are as follows: • *vlan-identifier***—**Specifies the VLAN ID. • **name—**Specifies the VLAN name.
Step 9	**ip multicast vlan** *vlan-name* **Example:** `Controller(config-wlan)# ip multicast vlan test`	Enables IP multicast on a WLAN. The keywords are as follows: • **vlan—**Specifies the VLAN ID. • *vlan-name***—**Specifies the VLAN name.
Step 10	**no shutdown** **Example:** `Controller(config-wlan)# no shutdown`	Enables the WLAN.
Step 11	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring Advanced WLAN Properties (CLI)

You can configure the following advanced properties:

- AAA Override
- Coverage Hole Detection
- Session Timeout
- Cisco Client Extensions
- Diagnostic Channels
- Interface Override ACLs

- P2P Blocking
- Client Exclusion
- Maximum Clients Per WLAN
- Off Channel Scan Defer

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan-name*
3. **aaa-override**
4. **chd**
5. **session-timeout** *time-in-seconds*
6. **ccx aironet-iesupport**
7. **diag-channel**
8. **ip access-group** *acl-name*
9. **peer-blocking** [**drop** | **forward-upstream**]
10. **exclusionlist** *time-in-seconds*
11. **client association limit** *max-number-of-clients*
12. **channel-scan defer-priority** { [**0-7**] | **defer-value**}
13. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *wlan-name* **Example:** `Controller# wlan test4`	Enters the WLAN configuration submode. The *wlan-name* is the profile name of the configured WLAN.
Step 3	**aaa-override** **Example:** `Controller(config-wlan)# aaa-override`	Enables AAA override.
Step 4	**chd** **Example:** `Controller(config-wlan)# chd`	Enables coverage hole detection for this WLAN. This field is enabled by default.

	Command or Action	Purpose
Step 5	**session-timeout** *time-in-seconds* **Example:** `Controller(config-wlan)# session-timeout 450`	Sets the session timeout in seconds. The range and default values vary according to the security configuration. If the WLAN security is configured to dot1x, the range is 300 to 86400 seconds and the default value is 1800 seconds. For all other WLAN security configurations, the range is 1 to 65535 seconds and the default value is 0 seconds. A value of 0 indicates no session timeout.
Step 6	**ccx aironet-iesupport** **Example:** `Controller(config-wlan)# ccx aironet-iesupport`	Enables support for Aironet IEs for this WLAN. This field is enabled by default.
Step 7	**diag-channel** **Example:** `Controller(config-wlan)# diag-channel`	Enables diagnostic channel support to troubleshoot client communication issues on a WLAN.
Step 8	**ip access-group** *acl-name* **Example:** `Controller(config)# ip access-group test-acl-name`	Configures the WLAN ACL group. The variable *acl-name* specifies the user-defined IPv4 ACL name.
Step 9	**peer-blocking** [**drop** | **forward-upstream**] **Example:** `Controller(config)# peer-blocking drop`	Configures peer to peer blocking parameters. The keywords are as follows: • **drop—** Enables peer-to-peer blocking on the drop action. • **forward-upstream—**Enables peer-to-peer blocking on the forward upstream action.
Step 10	**exclusionlist** *time-in-seconds* **Example:** `Controller(config)# exclusionlist`	Specifies the timeout in seconds. The valid range is from 0 to 2147483647. Enter 0 for no timeout. A zero (0) timeout indicates that the client is permanently added to the exclusion list.
Step 11	**client association limit** *max-number-of-clients* **Example:** `Controller(config)# client association limit 200`	Sets the maximum number of clients that can be configured on a WLAN.
Step 12	**channel-scan defer-priority** { [**0-7**] | **defer-value**} **Example:** `Controller(config)# channel-scan defer-priority 6`	Sets the channel scan defer priority and defer time. The arguments are as follows: • *defer-priority*—Specifies the priority markings for packets that can defer off-channel scanning. The range is from 0 to 7. The default is 3. • *defer-time*—Deferral time in milliseconds. The range is from 0 to 60000. The default is 100.

	Command or Action	Purpose
Step 13	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Related Topics

Band Selection, on page 958

Off-Channel Scanning Defer, on page 959

DTIM Period, on page 959

Session Timeout, on page 960

Cisco Client Extensions, on page 960

Peer-to-Peer Blocking, on page 961

Diagnostic Channel, on page 961

Client Count Per WLAN, on page 961

Information About AAA Override, on page 982

Monitoring WLAN Properties (CLI)

Command	Description
show wlan id *wlan-id*	Displays WLAN properties based on the WLAN ID.
show wlan name *wlan-name*	Displays WLAN properties based on the WLAN name.
show wlan all	Displays WLAN properties of all configured WLANs.
show wlan summary	Displays a summary of all WLANs. The summary details includes the following information: • WLAN ID • Profile name • SSID • VLAN • Status
show running-config wlan *wlan-name*	Displays the running configuration of a WLAN based on the WLAN name.

Command	Description
show running-config *wlan*	Displays the running configuration of all WLANs.

Where to Go Next

Proceed to configure DHCP for WLANs.

Additional References

Related Documents

Related Topic	Document Title
WLAN command reference	*WLAN Command Reference, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*

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

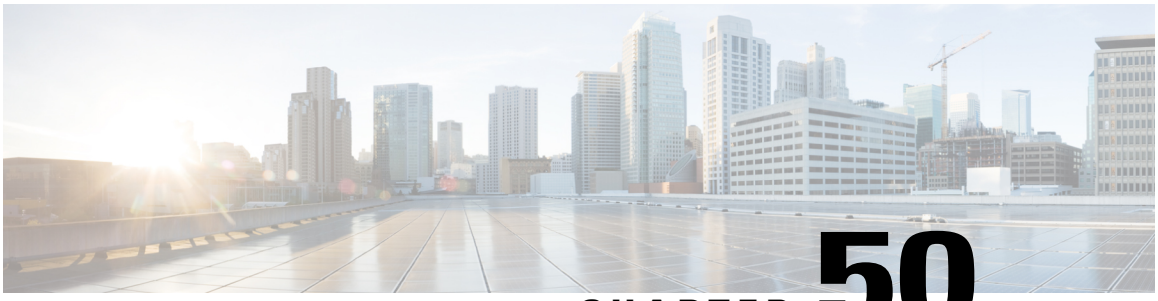
Technical Assistance

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 Information for WLANs

This table lists the features in this module and provides links to specific configuration information:

Feature Name	Release	Feature Information
WLAN functionality		This feature was introduced.



CHAPTER 50

Configuring DHCP for WLANs

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see 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 document.

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 DHCP for WLANs

- To be able to use the DHCP option 82, you must configure DHCP on Cisco IOS software. By default, DHCP option 82 is enabled for all clients. You can control the wireless client behavior using the WLAN suboptions.

Restrictions for Configuring DHCP for WLANs

- If you override the DHCP server in a WLAN, you must ensure that you configure the underlying Cisco IOS configuration to make sure that the DHCP server is reachable.

- WLAN DHCP override works only if DHCP service is enabled on the controller.

  You can configure DHCP service in the following ways:

  - Configuring the DHCP pool on the controller.
  - Configuring a DHCP relay agent on the SVI. Note: the VLAN of the SVI must be mapped to the WLAN where DHCP override is configured.

Information About the Dynamic Host Configuration Protocol

You can configure WLANs to use the same or different Dynamic Host Configuration Protocol (DHCP) servers or no DHCP server. Two types of DHCP servers are available: internal and external.

Internal DHCP Servers

The controllers contain an internal DHCP server. This server is typically used in branch offices that do not already have a DHCP server. The wireless network generally contains a maximum of 10 access points or fewer, with the access points on the same IP subnet as the controller. The internal server provides DHCP addresses to wireless clients, direct-connect access points, and DHCP requests that are relayed from access points. Only lightweight access points are supported. When you want to use the internal DHCP server, you must set the management interface IP address of the controller as the DHCP server IP address.

DHCP option 43 is not supported on the internal server. Therefore, the access point must use an alternative method to locate the management interface IP address of the controller, such as local subnet broadcast, Domain Name System (DNS), or priming.

An internal DHCP server pool only serves the wireless clients of that controller, not clients of other controllers. Also, an internal DHCP server can serve only wireless clients, not wired clients.

When clients use the internal DHCP server of the controller, IP addresses are not preserved across reboots. As a result, multiple clients can be assigned with the same IP address. To resolve any IP address conflicts, clients must release their existing IP address and request a new one.

Note DHCPv6 is not supported in the internal DHCP servers.

External DHCP Servers

The operating system is designed to appear as a DHCP Relay to the network and as a DHCP server to clients with industry-standard external DHCP servers that support DHCP Relay, which means that each controller appears as a DHCP Relay agent to the DHCP server and as a DHCP server at the virtual IP address to wireless clients.

Because the controller captures the client IP address that is obtained from a DHCP server, it maintains the same IP address for that client during intra controller, inter controller, and inter-subnet client roaming.

Note External DHCP servers can support DHCPv6.

DHCP Assignments

You can configure DHCP on a per-interface or per-WLAN basis. We recommend that you use the primary DHCP server address that is assigned to a particular interface.

You can assign DHCP servers for individual interfaces. You can configure the management interface, AP-manager interface, and dynamic interface for a primary and secondary DHCP server, and you can configure the service-port interface to enable or disable DHCP servers.

Security Considerations

For enhanced security, we recommend that you require all clients to obtain their IP addresses from a DHCP server. To enforce this requirement, you can configure all WLANs with a DHCP Addr. Assignment Required setting, which disallows client static IP addresses. If DHCP Addr. Assignment Required is selected, clients must obtain an IP address via DHCP. Any client with a static IP address is not allowed on the network. The controller monitors DHCP traffic because it acts as a DHCP proxy for the clients.

Note WLANs that support management over wireless must allow management (device-servicing) clients to obtain an IP address from a DHCP server.

If slightly less security is tolerable, you can create WLANs with DHCP Addr. Assignment Required disabled. Clients then have the option of using a static IP address or obtaining an IP address from a designated DHCP server.

Note DHCP Addr. Assignment Required is not supported for wired guest LANs.

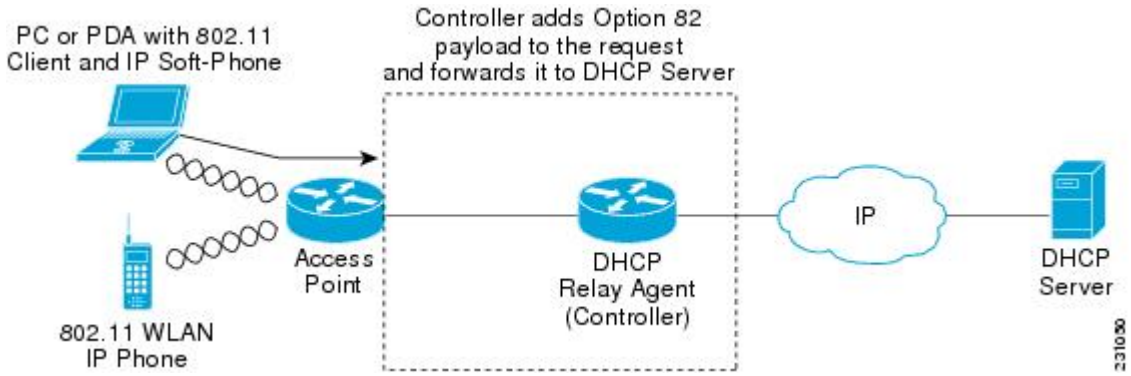
You can create separate WLANs with DHCP Addr. Assignment Required configured as disabled. This is applicable only if DHCP proxy is enabled for the controller. You must not define the primary/secondary configuration DHCP server you should disable the DHCP proxy. These WLANs drop all DHCP requests and force clients to use a static IP address. These WLANs do not support management over wireless connections.

Information About DHCP Option 82

DHCP option 82 provides additional security when DHCP is used to allocate network addresses. It enables the controller to act as a DHCP relay agent to prevent DHCP client requests from untrusted sources. You can

configure the controller to add option 82 information to DHCP requests from clients before forwarding the requests to the DHCP server.

Figure 45: DHCP Option 82



The access point forwards all DHCP requests from a client to the controller. The controller adds the DHCP option 82 payload and forwards the request to the DHCP server. The payload can contain the MAC address or the MAC address and SSID of the access point, depending on how you configure this option.

Note Any DHCP packets that already include a relay agent option are dropped at the controller.

Configuring DHCP Scopes

Information About DHCP Scopes

Controllers have built-in DHCP relay agents. However, when you desire network segments that do not have a separate DHCP server, the controllers can have built-in DHCP scopes that assign IP addresses and subnet masks to wireless clients. Typically, one controller can have one or more DHCP scopes that each provide a range of IP addresses.

DHCP scopes are needed for internal DHCP to work. Once DHCP is defined on the controller, you can then point the primary DHCP server IP address on the management, AP-manager, and dynamic interfaces to the controller's management interface.

How to Configure DHCP for WLANs

Configuring DHCP for WLANs (CLI)

Use this procedure to configure the following DHCP parameters on a WLAN:

- DHCP Option 82 Payload
- DHCP Required
- DHCP Override

Before You Begin

- You must have admin privileges for configuring the WLAN.
- To configure the DHCP override, you must have the IP address of the DHCP server.

SUMMARY STEPS

1. **configure terminal**
2. **shutdown**
3. **wlan** *wlan-name*
4. **ip dhcp opt82** { **ascii** | **format** {*add-ssid* | *ap-ethmac*} | **rid**}
5. **ip dhcp required**
6. **ip dhcp server** *ip-address*
7. **no shutdown**
8. **end**
9. **show wlan** *wlan-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**shutdown** **Example:** `Controller(config)# shutdown`	Shut down the WLAN.
Step 3	**wlan** *wlan-name* **Example:** `Controller# wlan test4`	Enters the WLAN configuration submode. The *wlan-name* is the profile name of the configured WLAN.

Step 4	**ip dhcp opt82** { **ascii** | **format** {*add-ssid* | *ap-ethmac*} | **rid**}
	
	Example:
	`Controller(config)#ip dhcp opt82 format add-ssid`
		Specifies the DHCP82 payload on the WLAN. The keyword and arguments are as follows:
		• **ascii—**Configures ASCII for DHCP Option 82. If this is not configured, the option 82 format is set to ASCII format.
		• **format—**Specifies the DHCP option 82 format. The following options are available:
		  • *add-ssid*—Set RemoteID format that is the AP radio MAC address and SSID.
		  • *ap-ethmac*—Set RemoteID format that is the AP Ethernet MAC address.

	Command or Action	Purpose
		Note If the format option is not configured, only the AP radio MAC address is used. • rid—Adds the Cisco 2 byte RID for DHCP option 82.
Step 5	**ip dhcp required** **Example:** `Controller(config-wlan)# ip dhcp required`	Makes it mandatory for clients to get their IP address from the DHCP server. Static clients are not allowed.
Step 6	**ip dhcp server** *ip-address* **Example:** `Controller(config-wlan)# ip dhcp server 200.1.1.2`	Defines a DHCP server on the WLAN that overrides the DHCP server address on the interface assigned to the WLAN.
Step 7	**no shutdown** **Example:** `Controller(config-wlan)# no shutdown`	Restarts the WLAN.
Step 8	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.
Step 9	**show wlan** *wlan-name* **Example:** `Controller(config-wlan)# show wlan test-wlan`	Verifies the DHCP configuration.

Additional References

Related Documents

Related Topic	Document Title
System Management	*System Management Configuration Guide (Cisco WLC 5700 Series)*

MIBs

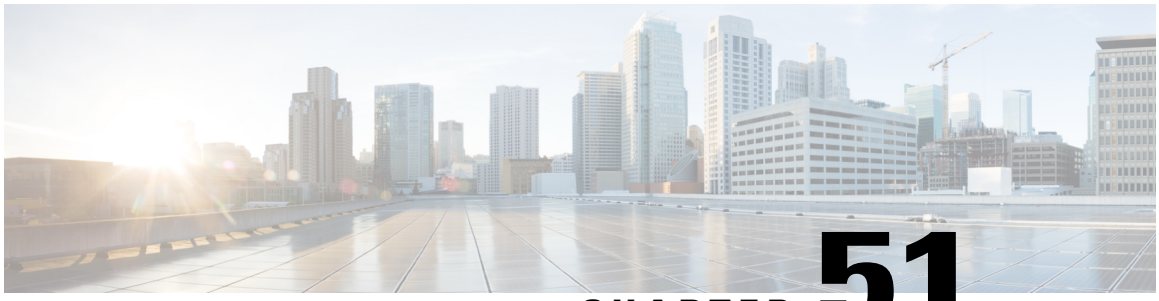
MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

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 Information for DHCP for WLANs

Feature Name	Release	Feature Information
DHCP functionality for WLAN		This feature was introduced.



CHAPTER 51

Configuring WLAN Security

This chapter contains the following sections:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see 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 document.

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 WLANs

- You can associate up to 16 WLANs with each access point group and assign specific access points to each group. Each access point advertises only the enabled WLANs that belong to its access point group. The access point (AP) does not advertise disabled WLANs in its access point group or WLANs that belong to another group.
- We recommend that you assign one set of VLANs for WLANs and a different set of VLANs for management interfaces to ensure that controllers properly route VLAN traffic.
- The controller uses different attributes to differentiate between WLANs with the same Service Set Identifier (SSID).

- WLANs with the same SSID and same Layer 2 policy cannot be created if the WLAN ID is lower than 17.
- Two WLANs with IDs that are greater than 17 and that have the same SSID and same Layer 2 policy is allowed if WLANs are added in different AP groups.

Note This requirement ensures that clients never detect the SSID present on the same access point radio.

Information About AAA Override

The AAA Override option of a WLAN enables you to configure the WLAN for identity networking. It enables you to apply VLAN tagging, Quality of Service (QoS), and Access Control Lists (ACLs) to individual clients based on the returned RADIUS attributes from the AAA server.

Related Topics

Configuring Advanced WLAN Properties (CLI), on page 966

How to Configure WLAN Security

Configuring Static WEP + 802.1X Layer 2 Security Parameters (CLI)

Before You Begin

You must have administrator privileges.

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan-name*
3. **security static-wep-key** {**authentication** {**open** | **sharedkey**} | **encryption** {**104** | **40**} [**ascii** | **hex**] { **0** | **8** } } *wep-key wep-key-index1-4*
4. **end**

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	**wlan** *wlan-name* **Example:** `Controller# wlan test4`	Enters the WLAN configuration submode. The *wlan-name* is the profile name of the configured WLAN.
Step 3	**security static-wep-key** {**authentication** {**open** | **sharedkey**} | **encryption** {**104** | **40**} [**ascii** | **hex**] { **0** | **8** } } *wep-key wep-key-index1-4* **Example:** `Controller(config-wlan)# security static-wep-key encryption 40 hex 0 test 2`	Configures static WEP security on a WLAN. The keywords and arguments are as follows: • **authentication—**Configures 802.11 authentication. • **encryption—**Sets the static WEP keys and indices. • **open—** Configures open system authentication. • **sharedkey—**Configures shared key authentication. • **104, 40 —** Specifies the WEP key size. • **hex, ascii—**Specifies the input format of the key. • *wep-key-index***—**Type of password that follows. A value of 0 indicates that an unencrypted password follows. A value of 8 indicates that an AES encrypted follows.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring Static WEP Layer 2 Security Parameters (CLI)

Before You Begin

You must have administrator privileges.

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan-name*
3. **security static-wep-key** [**authentication** {**open** | **shared**} | **encryption** {**104** | **40**} {**ascii** | **hex**} [**0** | **8**]]
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *wlan-name* **Example:** `Controller# wlan test4`	Enters the WLAN configuration submode. The *wlan-name* is the profile name of the configured WLAN.
Step 3	**security static-wep-key** [**authentication** {**open** | **shared**} | **encryption** {**104** | **40**} {**ascii** | **hex**} [**0** | **8**]] **Example:** `Controller(config-wlan)# security static-wep-key authentication open`	The keywords are as follows: • **static-wep-key—**Configures Static WEP Key authentication. • **authentication—**Specifies the authentication type you can set. The values are open and shared. • **encryption—**Specifies the encryption type that you can set. The valid values are 104 and 40. 40-bit keys must contain 5 ASCII text characters or 10 hexadecimal characters. 104-bit keys must contain 13 ASCII text characters or 26 hexadecimal characters • **ascii—**Specifies the key format as ASCII. • **hex—**Specifies the key format as HEX.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring WPA + WPA2 Layer 2 Security Parameters (CLI)

Note The default security policy is WPA2.

Before You Begin

You must have administrator privileges.

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan-name*
3. **security wpa**
4. **security wpa wpa1**
5. **security wpa wpa1 ciphers** [**aes** | **tkip**]
6. **security wpa wpa2**
7. **security wpa wpa2 ciphers** [**aes** | **tkip**]
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *wlan-name* **Example:** `Controller# wlan test4`	Enters the WLAN configuration submode. The *wlan-name* is the profile name of the configured WLAN.
Step 3	**security wpa** **Example:** `Controller(config-wlan)# security wpa`	Enables WPA.
Step 4	**security wpa wpa1** **Example:** `Controller(config-wlan)# security wpa wpa1`	Enables WPA1.
Step 5	**security wpa wpa1 ciphers** [**aes**	**tkip**] **Example:** `Controller(config-wlan)# security wpa wpa1 ciphers aes`	Specifies the WPA1 cipher. Choose one of the following encryption types: • **aes—**Specifies WPA/AES support. • **tkip—**Specifies WPA/TKIP support.
Step 6	**security wpa wpa2** **Example:** `Controller(config-wlan)# security wpa`	Enables WPA 2.
Step 7	**security wpa wpa2 ciphers** [**aes**	**tkip**] **Example:** `Controller(config-wlan)# security wpa wpa2 ciphers tkip`	Configure WPA2 cipher. Choose one of the following encryption types: • **aes—**Specifies WPA/AES support. • **tkip—**Specifies WPA/TKIP support.

	Command or Action	Purpose
Step 8	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring 802.1X Layer 2 Security Parameters (CLI)

Before You Begin

You must have administrator privileges.

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan-name*
3. **security dot1x**
4. **security** [**authentication-list** {*auth-list-name*} | **encryption** { **0** | **104** | **40** }
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *wlan-name* **Example:** `Controller# wlan test4`	Enters the WLAN configuration submode. The *wlan-name* is the profile name of the configured WLAN.
Step 3	**security dot1x** **Example:** `Controller(config-wlan)# security dot1x`	Specifies 802.1X security.
Step 4	**security** [**authentication-list** {*auth-list-name*}	**encryption** { **0**	**104**	**40** } **Example:** `Controller(config-wlan)# security encryption 104`	The keywords and arguments are as follows: • **authentication-list—**Specifies the authentication list for IEEE 802.1X. • **encryption—**Specifies the length of the CKIP encryption key. The valid values are 0, 40, and 104. Zero (0) signifies no encryption. This is the default.

	Command or Action	Purpose
		Note All keys within a WLAN must be of the same size.
Step 5	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Additional References

Related Documents

Related Topic	Document Title
WLAN command reference	*WLAN Command Reference, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*
Security configuration guide	*Security Configuration Guide (Cisco WLC 5700 Series)*

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

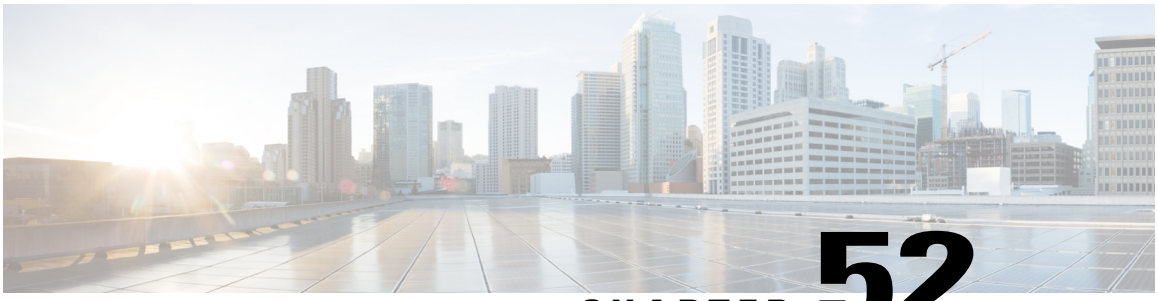
Technical Assistance

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 Information about WLAN Layer 2 Security

This table lists the features in this module and provides links to specific configuration information.

Feature Name	Release	Feature Information
WLAN Security functionality		This feature was introduced.



CHAPTER 52

Configuring Access Point Groups

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see 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 document.

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 AP Groups

The following are the prerequisites for creating access point groups on a controller:

- The required access control list (ACL) must be defined on the router that serves the VLAN or subnet.
- Multicast traffic is supported with access point group VLANs. However, if the client roams from one access point to another, the client might stop receiving multicast traffic, unless IGMP snooping is enabled.

Related Topics

Information About Access Point Groups, on page 990
Restrictions for Configuring Access Point Groups, on page 990

Restrictions for Configuring Access Point Groups

- Suppose that the interface mapping for a WLAN in the AP group table is the same as the WLAN interface. If the WLAN interface is changed, the interface mapping for the WLAN in the AP group table also changes to the new WLAN interface.

  Suppose that the interface mapping for a WLAN in the AP group table is different from the one defined for the WLAN. If the WLAN interface is changed, then the interface mapping for the WLAN in the AP group table does not change to the new WLAN interface.

- If you clear the configuration on the controller, all of the access point groups disappear except for the default access point group "default-group," which is created automatically.

Related Topics

Information About Access Point Groups, on page 990
Prerequisites for Configuring AP Groups, on page 989

Information About Access Point Groups

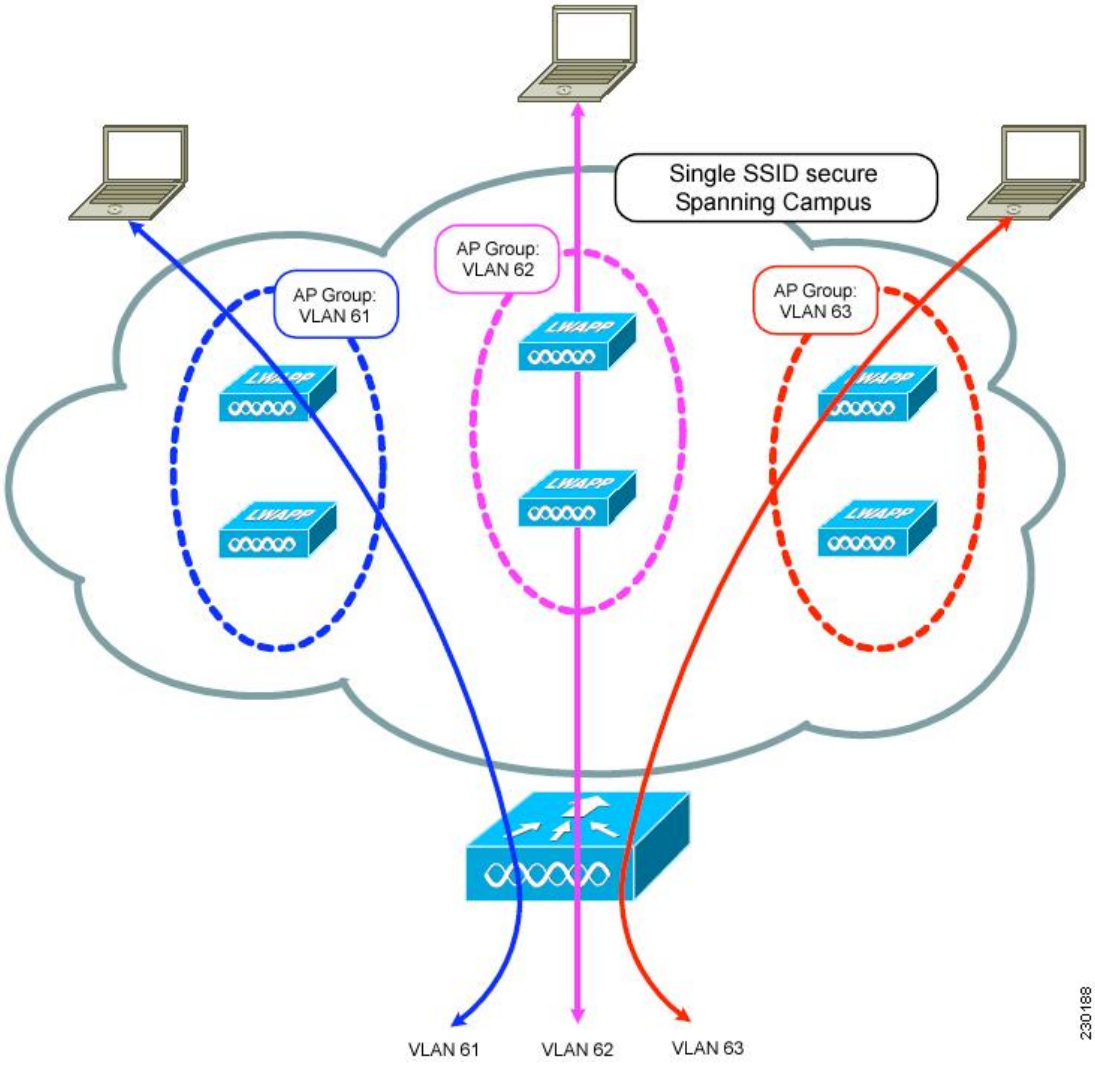
After you create up to 512 WLANs on the controller, you can selectively publish them (using access point groups) to different access points to better manage your wireless network. In a typical deployment, all users on a WLAN are mapped to a single interface on the controller. Therefore, all users that are associated with that WLAN are on the same subnet or VLAN. However, you can choose to distribute the load among several interfaces or to a group of users based on specific criteria such as individual departments (such as Marketing) by creating access point groups. Additionally, these access point groups can be configured in separate VLANs to simplify network administration.

In the figure, three configured dynamic interfaces are mapped to three different VLANs (VLAN 61, VLAN 62, and VLAN 63). Three access point groups are defined, and each is a member of a different VLAN, but all are members of the same SSID. A client within the wireless SSID is assigned an IP address from the VLAN subnet on which its access point is a member. For example, any user that associates with an access point that is a member of access point group VLAN 61 is assigned an IP address from that subnet.

In the figure, the controller internally treats roaming between access points as a Layer 3 roaming event. In this way, WLAN clients maintain their original IP addresses.

After all access points have joined the controller, you can create access point groups and assign up to 16 WLANs to each group. Each access point advertises only the enabled WLANs that belong to its access point group. The access point does not advertise disabled WLANs in its access point group or WLANs that belong to another group.

Figure 46: Access Point Groups



Related Topics

Creating Access Point Groups, on page 992

Viewing Access Point Group, on page 993

Assigning an Access Point to an AP Group, on page 993

Prerequisites for Configuring AP Groups, on page 989

Restrictions for Configuring Access Point Groups, on page 990

How to Configure Access Point Groups

Creating Access Point Groups

Before You Begin

You must have administrator privileges to perform this operation.

SUMMARY STEPS

1. **configure terminal**
2. **ap group** *ap-group-name*
3. **wlan** *wlan-name*
4. (Optional) **vlan** *vlan-name*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**ap group** *ap-group-name* **Example:** `Controller(config)# ap group my-ap-group`	Creates an access point group.
Step 3	**wlan** *wlan-name* **Example:** `Controller(config-apgroup)# wlan wlan-name`	Associates the AP group to a WLAN.
Step 4	**vlan** *vlan-name* **Example:** `Controller(config-apgroup)# vlan test-vlan`	(Optional) Assigns the access point group to a VLAN.
Step 5	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

This example shows how to create an AP group:

```
Controller# configure terminal
Controller(config-apgroup)# ap group test-ap-group-16
```

```
Controller(config-wlan-apgroup)# wlan test-ap-group-16
Controller(config-wlan-apgroup)# vlan VLAN1300
```

Related Topics

Information About Access Point Groups, on page 990

Assigning an Access Point to an AP Group

Before You Begin

You must have administrator privileges to perform this operation.

SUMMARY STEPS

1. **ap name** *ap-name* **ap-group-name** *ap-group*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**ap name** *ap-name* **ap-group-name** *ap-group* **Example:** `Controller# ap name 1240-101 ap-groupname apgroup_16`	Assigns the access point to the access point group. The keywords and arguments are as follows: • **name**—Specifies that the argument following this keyword is the name of an AP that is associated to the controller. • *ap-name*—AP that you want to associate to the AP group. • **ap-group-name**—Specifies that the argument following this keyword is the name of the AP group that is configured on the controller. • *ap-group*—Name of the access point group that is configured on the controller.

Related Topics

Information About Access Point Groups, on page 990

Viewing Access Point Group

Before You Begin

You must have administrator privileges to perform this operation.

SUMMARY STEPS

1. **show ap groups**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show ap groups** **Example:** `Controller# show ap groups`	Displays the AP groups configured on the controller.

Related Topics

Information About Access Point Groups, on page 990

Additional References

Related Documents

Related Topic	Document Title
WLAN commands	*WLAN Command Reference, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*
Lightweight Access Point configuration	*Lightweight Access Point Configuration Guide, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*
Lightweight Access Point commands	*Lightweight Access Point Command Reference, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*

MIBs

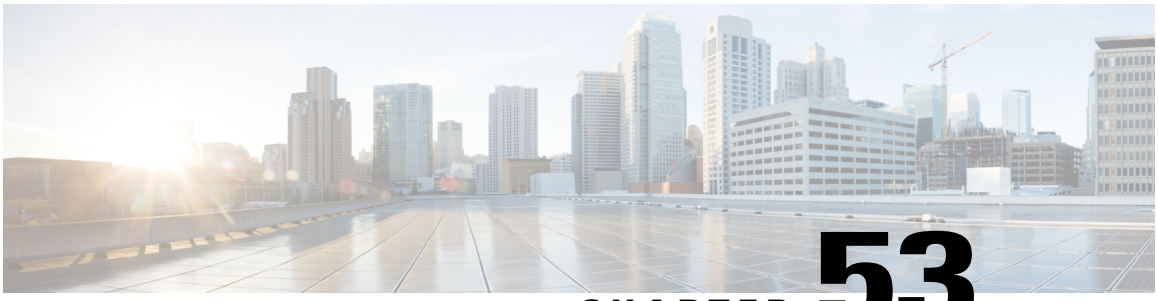
MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

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

PART X

Radio Resource Management



CHAPTER 53

Configuring Radio Resource Management

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Radio Resource Management

The Controller should be configured as a mobility controller and not a mobility anchor to configure radio resource management. It may require dynamic channel assignment functionality for the home APs to be supported.

Restrictions for Radio Resource Management

The number of APs in a RF-group is limited to 2000.

If a AP tries to join the RF-group that already holds the maximum number of APs it can support, the device rejects the application and throws an error.

Information About Radio Resource Management

The Radio Resource Management (RRM) software embedded in the controller acts as a built-in RF engineer to consistently provide real-time RF management of your wireless network. RRM enables controllers to continually monitor their associated lightweight access points for the following information:

- Traffic load—The total bandwidth used for transmitting and receiving traffic. It enables wireless LAN managers to track and plan network growth ahead of client demand.
- Interference—The amount of traffic coming from other 802.11 sources.
- Noise—The amount of non-802.11 traffic that is interfering with the currently assigned channel.
- Coverage—The received signal strength (RSSI) and signal-to-noise ratio (SNR) for all connected clients.
- Other —The number of nearby access points.

RRM performs these functions:

- Radio resource monitoring
- Transmit power control
- Dynamic channel assignment
- Coverage hole detection and correction

Radio Resource Monitoring

RRM automatically detects and configures new controllers and lightweight access points as they are added to the network. It then automatically adjusts associated and nearby lightweight access points to optimize coverage and capacity.

Lightweight access points can simultaneously scan all valid 802.11a/b/g channels for the country of operation as well as for channels available in other locations. The access points go “off-channel” for a period not greater than 60 ms to monitor these channels for noise and interference. Packets collected during this time are analyzed to detect rogue access points, rogue clients, ad-hoc clients, and interfering access points.

Note In the presence of voice traffic (in the last 100 ms), the access points defer off-channel measurements.

Each access point spends only 0.2 percent of its time off-channel. This activity is distributed across all access points so that adjacent access points are not scanning at the same time, which could adversely affect wireless LAN performance.

Transmit Power Control

The controller dynamically controls access point transmit power based on real-time wireless LAN conditions. Typically in TPCv1, power can be kept low to gain extra capacity and reduce interference.

The transmit power control (TPC) algorithm both increases and decreases an access point’s power in response to changes in the RF environment. In most instances, TPC seeks to lower an access point's power to reduce interference, but in the case of a sudden change in the RF coverage—for example, if an access point fails or

becomes disabled—TPC can also increase power on surrounding access points. This feature is different from coverage hole detection, which is primarily concerned with clients. TPC provides enough RF power to achieve desired coverage levels while avoiding channel interference between access points.

Overriding the TPC Algorithm with Minimum and Maximum Transmit Power Settings

The TPC algorithm balances RF power in many diverse RF environments. However, it is possible that automatic power control will not be able to resolve some scenarios in which an adequate RF design was not possible to implement due to architectural restrictions or site restrictions—for example, when all access points must be mounted in a central hallway, placing the access points close together, but requiring coverage out to the edge of the building.

In these scenarios, you can configure maximum and minimum transmit power limits to override TPC recommendations. The maximum and minimum TPC power settings apply to all access points through RF profiles in a RF network.

To set the Maximum Power Level Assignment and Minimum Power Level Assignment text boxes, enter the maximum and minimum transmit power used by RRM on the Tx Power Control page. The range for these parameters is -10 to 30 dBm. The minimum value cannot be greater than the maximum value; the maximum value cannot be less than the minimum value.

If you configure a maximum transmit power, RRM does not allow any access point attached to the controller to exceed this transmit power level (whether the power is set by RRM TPC or by coverage hole detection). For example, if you configure a maximum transmit power of 11 dBm, then no access point would transmit above 11 dBm, unless the access point is configured manually.

Dynamic Channel Assignment

Two adjacent access points on the same channel can cause either signal contention or signal collision. In a collision, data is not received by the access point. This functionality can become a problem, for example, when someone reading e-mail in a café affects the performance of the access point in a neighboring business. Even though these are completely separate networks, someone sending traffic to the café on channel 1 can disrupt communication in an enterprise using the same channel. Controllers can dynamically allocate access point channel assignments to avoid conflict and to increase capacity and performance. Channels are “reused” to avoid wasting scarce RF resources. In other words, channel 1 is allocated to a different access point far from the café, which is more effective than not using channel 1 altogether.

The controller's dynamic channel assignment (DCA) capabilities are also useful in minimizing adjacent channel interference between access points. For example, two overlapping channels in the 802.11b/g band, such as 1 and 2, cannot both simultaneously use 11/54 Mbps. By effectively reassigning channels, the controller keeps adjacent channels separated.

Note We recommend that you use only non-overlapping channels (1, 6, 11, and so on).

The controller examines a variety of real-time RF characteristics to efficiently handle channel assignments as follows:

- Access point received energy—The received signal strength measured between each access point and its nearby neighboring access points. Channels are optimized for the highest network capacity.
- Noise—Noise can limit signal quality at the client and access point. An increase in noise reduces the effective cell size and degrades user experience. By optimizing channels to avoid noise sources, the

controller can optimize coverage while maintaining system capacity. If a channel is unusable due to excessive noise, that channel can be avoided.

- 802.11 Interference—Interference is any 802.11 traffic that is not part of your wireless LAN, including rogue access points and neighboring wireless networks. Lightweight access points constantly scan all channels looking for sources of interference. If the amount of 802.11 interference exceeds a predefined configurable threshold (the default is 10 percent), the access point sends an alert to the controller. Using the RRM algorithms, the controller may then dynamically rearrange channel assignments to increase system performance in the presence of the interference. Such an adjustment could result in adjacent lightweight access points being on the same channel, but this setup is preferable to having the access points remain on a channel that is unusable due to an interfering foreign access point.

  In addition, if other wireless networks are present, the controller shifts the usage of channels to complement the other networks. For example, if one network is on channel 6, an adjacent wireless LAN is assigned to channel 1 or 11. This arrangement increases the capacity of the network by limiting the sharing of frequencies. If a channel has virtually no capacity remaining, the controller may choose to avoid this channel. In very dense deployments in which all nonoverlapping channels are occupied, the controller does its best, but you must consider RF density when setting expectations.

- Utilization—When utilization monitoring is enabled, capacity calculations can consider that some access points are deployed in ways that carry more traffic than other access points (for example, a lobby versus an engineering area). The controller can then assign channels to improve the access point with the worst performance reported.

- Load—The load is taken into account when changing the channel structure to minimize the impact on clients currently in the wireless LAN. This metric keeps track of every access point's transmitted and received packet counts to determine how busy the access points are. New clients avoid an overloaded access point and associate to a new access point. This parameter is disabled by default.

The controller combines this RF characteristic information with RRM algorithms to make system-wide decisions. Conflicting demands are resolved using soft-decision metrics that guarantee the best choice for minimizing network interference. The end result is optimal channel configuration in a three-dimensional space, where access points on the floor above and below play a major factor in an overall wireless LAN configuration.

Note Radios using 40-MHz channels in the 2.4-GHz band are not supported by DCA.

The RRM startup mode is invoked in the following conditions:

- In a single-controller environment, the RRM startup mode is invoked after the controller is rebooted.
- In a multiple-controller environment, the RRM startup mode is invoked after an RF Group leader is elected.

RRM startup mode runs for 100 minutes (10 iterations at 10-minute intervals). The duration of the RRM startup mode is independent of the DCA interval, sensitivity, and network size. The startup mode consists of 10 DCA runs with high sensitivity (making channel changes easy and sensitive to the environment) to converge to a steady state channel plan. After the startup mode is finished, DCA continues to run at the specified interval and sensitivity.

Coverage Hole Detection and Correction

The RRM coverage hole detection algorithm can detect areas of radio coverage in a wireless LAN that are below the level needed for robust radio performance. This feature can alert you to the need for an additional (or relocated) lightweight access point.

If clients on a lightweight access point are detected at threshold levels (RSSI, failed client count, percentage of failed packets, and number of failed packets) lower than those specified in the RRM configuration, the access point sends a "coverage hole" alert to the controller. The alert indicates the existence of an area where clients are continually experiencing poor signal coverage, without having a viable access point to which to roam. The controller discriminates between coverage holes that can and cannot be corrected. For coverage holes that can be corrected, the controller mitigates the coverage hole by increasing the transmit power level for that specific access point. The controller does not mitigate coverage holes caused by clients that are unable to increase their transmit power or are statically set to a power level because increasing their downstream transmit power might increase interference in the network.

How to Configure RRM

Configuring Advanced RRM CCX Parameters

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 24ghz | 5ghz rrm ccx location-measurement** *interval*
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 24ghz | 5ghz rrm ccx location-measurement** *interval* **Example:** `Controller(config)# ap dot11 24ghz rrm ccx location-measurement 15`	Configures the interval for 802.11 CCX client location measurements. The range is from 10 to 32400 seconds.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring Advanced 802.11 Channel Assignment Parameters

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 24ghz | 5ghz rrm channel cleanair-event**{**high | low | medium**}
3. **ap dot11 24ghz | 5ghz rrm channel dca**{*channel number*| **anchor-time | global**{**auto| once**}| **interval | min-metric | sensitivity**{**high | low | medium**}}
4. **ap dot11 24ghz | 5ghz rrm channel device**
5. **ap dot11 24ghz | 5ghz rrm channel foreign**
6. **ap dot11 24ghz | 5ghz rrm channel load**
7. **ap dot11 24ghz | 5ghz rrm channel noise**
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 24ghz | 5ghz rrm channel cleanair-event**{**high | low | medium**} **Example:** `Controller(config)#ap dot11 24ghz rrm channel cleanair-event sensitivity high`	Configures cleanair event-driven RRM parameters. • **High–** Specifies the most sensitivity to non-WiFi interference as indicated by the air quality (AQ) value. • **Low–** Specifies the least sensitivity to non-WiFi interference as indicated by the AQ value. • **Medium–** Specifies medium sensitivity to non-WiFi interference as indicated by the AQ value.
Step 3	**ap dot11 24ghz | 5ghz rrm channel dca**{*channel number*| **anchor-time | global**{**auto| once**}| **interval | min-metric | sensitivity**{**high | low | medium**}} **Example:** `Controller(config)#ap dot11 24ghz rrm channel dca interval 2`	Configures dynamic channel assignment (DCA) algorithm parameters for the 802.11 band. • ***<1-14>–*** Enter a channel number to be added to the DCA list. • **anchor-time–** Configures the anchor time for the DCA. The range is between 0 and 23 hours. • **global–** Configures the DCA mode for all 802.11 Cisco APs. ◦ **auto–** Enables auto-RF. ◦ **once–** Enables auto-RF only once. • **interval–** Configures the DCA interval value. The values are 1, 2, 3, 4, 6, 8, 12 and 24 hours and the default value 0 denotes 10 minutes.

	Command or Action	Purpose
		• **min-metric–** Configures the DCA minimum RSSI energy metric. The range is between -100 and -60. • **sensitivity–** Configures the DCA sensitivity level to changes in the environment. ◦ **high–** Specifies the most sensitivity. ◦ **low–** Specifies the least sensitivity. ◦ **medium–** Specifies medium sensitivity.
Step 4	**ap dot11 24ghz | 5ghz rrm channel device** **Example:** `Controller(config)#ap dot11 24ghz rrm channel device`	Configures the persistent non-WiFi device avoidance in the 802.11 channel assignment.
Step 5	**ap dot11 24ghz | 5ghz rrm channel foreign** **Example:** `Controller(config)#ap dot11 24ghz rrm channel foreign`	Configures the foreign AP 802.11 interference avoidance in the channel assignment.
Step 6	**ap dot11 24ghz | 5ghz rrm channel load** **Example:** `Controller(config)#ap dot11 24ghz rrm channel load`	Configures the Cisco AP 802.11 load avoidance in the channel assignment.
Step 7	**ap dot11 24ghz | 5ghz rrm channel noise** **Example:** `Controller(config)#ap dot11 24ghz rrm channel noise`	Configures the 802.11 noise avoidance in the channel assignment.
Step 8	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring 802.11 Coverage Hole Detection

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 24ghz | 5ghz rrm coverage data**{**fail-percentage | packet-count | rssi-threshold**}
3. **ap dot11 24ghz | 5ghz rrm coverage exception global** *exception level*
4. **ap dot11 24ghz | 5ghz rrm coverage level global** *cli_min exception level*
5. **ap dot11 24ghz | 5ghz rrm coverage voice**{**fail-percentage | packet-count | rssi-threshold**}
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 24ghz | 5ghz rrm coverage data**{**fail-percentage | packet-count | rssi-threshold**} **Example:** `Controller(config)#ap dot11 24ghz rrm coverage data fail-percentage 90`	Configures the 802.11 coverage hole detection for data packets. • **fail-percentage—** Configures the 802.11 coverage failure-rate threshold for uplink data packets as a percentage that ranges from 1 to 100%. • **packet-count—** Configures the 802.11 coverage minimum failure count threshold for uplink data packets that ranges from 1 to 255. • **rssi-threshold—** Configures the 802.11 minimum receive coverage level for data packets that range from –90 to –60 dBm.
Step 3	**ap dot11 24ghz | 5ghz rrm coverage exception global** *exception level* **Example:** `Controller(config)#ap dot11 24ghz rrm coverage exception global 50`	Configures the 802.11 Cisco AP coverage exception level as a percentage that ranges from 0 to 100%.
Step 4	**ap dot11 24ghz | 5ghz rrm coverage level global** *cli_min exception level* **Example:** `Controller(config)#ap dot11 24ghz rrm coverage level global 10`	Configures the 802.11 Cisco AP client minimum exception level that ranges from 1 to 75 clients.

	Command or Action	Purpose
Step 5	**ap dot11 24ghz | 5ghz rrm coverage voice**{**fail-percentage | packet-count | rssi-threshold**} **Example:** `Controller(config)#ap dot11 24ghz rrm coverage voice packet-count 200`	Configures the 802.11 coverage hole detection for voice packets. • **fail-percentage—** Configures the 802.11 coverage failure-rate threshold for uplink voice packets as a percentage that ranges from 1 to 100%. • **packet-count—** Configures the 802.11 coverage minimum failure count threshold for uplink voice packets that ranges from 1 to 255. • **rssi-threshold—** Configures the 802.11 minimum receive coverage level for voice packets that range from –90 to –60 dBm.
Step 6	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring Members in a 802.11 Static RF Group

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 24ghz | 5ghz rrm group-member** *group_name ip_addr*
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 24ghz | 5ghz rrm group-member** *group_name ip_addr* **Example:** `Controller(config)#ap dot11 24ghz rrm group-member Grpmem01 10.1.1.1`	Configures members in a 802.11 static RF group. The group mode should be set as leader for the group member to be active.

	Command or Action	Purpose
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring RF Group Selection Mode

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 24ghz | 5ghz rrm group-mode**{**auto | leader | off**}
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 24ghz	5ghz rrm group-mode**{**auto	leader	off**} **Example:** `Controller(config)#ap dot11 24ghz rrm group-mode leader`	Configures RF group selection mode for 802.11 bands. • **auto—** Sets the 802.11 RF group selection to automatic update mode. • **leader—** Sets the 802.11 RF group selection to leader mode. • **off—** Disables the 802.11 RF group selection.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring 802.11 Event Logging

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 24ghz | 5ghz rrm logging**{**channel | coverage | foreign | load | noise | performance | txpower**}
3. **end**

DETAILED STEPS

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

Step 2	**ap dot11 24ghz | 5ghz rrm logging**{**channel | coverage | foreign | load | noise | performance | txpower**}
	
	Example:
	`Controller(config)#ap dot11 24ghz rrm logging channel`
	`Controller(config)#ap dot11 24ghz rrm logging coverage`
	`Controller(config)#ap dot11 24ghz rrm logging foreign`
	`Controller(config)#ap dot11 24ghz rrm logging load`
	`Controller(config)#ap dot11 24ghz rrm logging noise`
	`Controller(config)#ap dot11 24ghz rrm logging performance`
	`Controller(config)#ap dot11 24ghz rrm logging txpower`
		Configures event-logging for various parameters.
		• **channel—** Configures the 802.11 channel change logging mode.
		• **coverage—** Configures the 802.11 coverage profile logging mode.
		• **foreign—** Configures the 802.11 foreign interference profile logging mode.
		• **load—** Configures the 802.11 load profile logging mode.
		• **noise—** Configures the 802.11 noise profile logging mode.
		• **performance—** Configures the 802.11 performance profile logging mode.
		• **txpower—** Configures the 802.11 transmit power change logging mode.

Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring 802.11 Statistics Monitoring

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 24ghz | 5ghz rrm monitor channel-list**{**all | country | dca**}
3. **ap dot11 24ghz | 5ghz rrm monitor coverage** *interval*
4. **ap dot11 24ghz | 5ghz rrm monitor load** *interval*
5. **ap dot11 24ghz | 5ghz rrm monitor noise** *interval*
6. **ap dot11 24ghz | 5ghz rrm monitor signal** *interval*
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 24ghz | 5ghz rrm monitor channel-list**{**all | country | dca**} **Example:** `Controller(config)#ap dot11 24ghz rrm monitor channel-list all`	Sets the 802.11 monitoring channel-list for parameters such as noise/interference/rogue. • **all—** Monitors all channels • **country—** Monitor channels used in configured country code • **dca—** Monitor channels used by dynamic channel assignment
Step 3	**ap dot11 24ghz | 5ghz rrm monitor coverage** *interval* **Example:** `Controller(config)#ap dot11 24ghz rrm monitor coverage 600`	Configures the 802.11 coverage measurement interval in seconds that ranges from 60 to 3600.
Step 4	**ap dot11 24ghz | 5ghz rrm monitor load** *interval* **Example:** `Controller(config)#ap dot11 24ghz rrm monitor load 180`	Configures the 802.11 load measurement interval in seconds that ranges from 60 to 3600.
Step 5	**ap dot11 24ghz | 5ghz rrm monitor noise** *interval* **Example:** `Controller(config)#ap dot11 24ghz rrm monitor noise 360`	Configures the 802.11 noise measurement interval (channel scan interval) in seconds that ranges from 60 to 3600.

	Command or Action	Purpose
Step 6	**ap dot11 24ghz | 5ghz rrm monitor signal** *interval* **Example:** `Controller(config)#ap dot11 24ghz rrm monitor signal 480`	Configures the 802.11 signal measurement interval (neighbor packet frequency) in seconds that ranges from 60 to 3600.
Step 7	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring Neighbor Discovery Type

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 24ghz | 5ghz rrm ndp-type** {**protected | transparent**}
3. **end**

DETAILED STEPS

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

Step 2	**ap dot11 24ghz | 5ghz rrm ndp-type** {**protected | transparent**}
	
	Example:
	`Controller(config)#ap dot11 24ghz rrm ndp-type protected`
	`Controller(config)#ap dot11 24ghz rrm ndp-type transparent`
		Configures the neighbor discovery type.
		• **protected—** Sets the neighbor discover type to protected.
		• **transparent—** Sets the neighbor discover type to transparent.

Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring the 802.11 Performance Profile

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 24ghz | 5ghz rrm profile clients** *cli_threshold_value*
3. **ap dot11 24ghz | 5ghz rrm profile foreign** *int_threshold_value*
4. **ap dot11 24ghz | 5ghz rrm profile noise** *for_noise_threshold_value*
5. **ap dot11 24ghz | 5ghz rrm profile throughput** *throughput_threshold_value*
6. **ap dot11 24ghz | 5ghz rrm profile utilization** *rf_util_threshold_value*
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 24ghz | 5ghz rrm profile clients** *cli_threshold_value* **Example:** `Controller(config)#ap dot11 24ghz rrm profile clients 20`	Sets the threshold value for 802.11 Cisco AP clients that range between 1 and 75 clients.
Step 3	**ap dot11 24ghz | 5ghz rrm profile foreign** *int_threshold_value* **Example:** `Controller(config)#ap dot11 24ghz rrm profile foreign 50`	Sets the threshold value for 802.11 foreign interference that ranges between 0 and 100%.
Step 4	**ap dot11 24ghz | 5ghz rrm profile noise** *for_noise_threshold_value* **Example:** `Controller(config)#ap dot11 24ghz rrm profile noise -10`	Sets the threshold value for 802.11 foreign noise ranges between –127 and 0 dBm.
Step 5	**ap dot11 24ghz | 5ghz rrm profile throughput** *throughput_threshold_value* **Example:** `Controller(config)#ap dot11 24ghz rrm profile throughput 10000`	Sets the threshold value for 802.11 Cisco AP throughput that ranges between 1000 and 10000000 bytes per second.

	Command or Action	Purpose
Step 6	**ap dot11 24ghz | 5ghz rrm profile utilization** *rf_util_threshold_value* **Example:** `Controller(config)#ap dot11 24ghz rrm profile utilization 50`	Sets the threshold value for 802.11 RF utilization that ranges between 0 to 100%
Step 7	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring the Tx-Power Control Threshold

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 24ghz | 5ghz rrm tpc-threshold** *threshold_value*
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 24ghz | 5ghz rrm tpc-threshold** *threshold_value* **Example:** `Controller(config)#ap dot11 24ghz rrm tpc-threshold -60`	Configures the Tx-power control threshold used by RRM for auto power assignment. The range is from –80 to –50.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring the Tx-Power Level

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 24ghz | 5ghz rrm txpower**{*trans_power_level* | **auto** | **max** | **min** | **once**}
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 24ghz | 5ghz rrm txpower**{*trans_power_level* | **auto** | **max** | **min** | **once**} **Example:** `Controller(config)#ap dot11 24ghz rrm txpower auto`	Configures the 802.11 tx-power level • **trans_power_level**- Sets the transmit power level. • **auto**- Enables auto-RF. • **max**- Configures the maximum auto-RF tx-power. • **min**- Configures the minimum auto-RF tx-power. • **once**- Enables one-time auto-RF.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Monitoring RRM Parameters

Table 108: Commands for monitoring Radio Resource Management

Commands	Description
show ap dot11 24ghz ccx	Displays the 802.11b CCX information for all Cisco APs.
show ap dot11 24ghz channel	Displays the configuration and statistics of the 802.11b channel assignment.
show ap dot11 24ghz coverage	Displays the configuration and statistics of the 802.11b coverage.

Commands	Description
show ap dot11 24ghz group	Displays the configuration and statistics of the 802.11b grouping.
show ap dot11 24ghz l2roam	Displays 802.11b l2roam information.
show ap dot11 24ghz logging	Displays the configuration and statistics of the 802.11b event logging.
show ap dot11 24ghz monitor	Displays the configuration and statistics of the 802.11b monitoring.
show ap dot11 24ghz profile	Displays 802.11b profiling information for all Cisco APs.
show ap dot11 24ghz receiver	Displays the configuration and statistics of the 802.11b receiver.
show ap dot11 24ghz summary	Displays the configuration and statistics of the 802.11b Cisco APs.
show ap dot11 24ghz txpower	Displays the configuration and statistics of the 802.11b transmit power control.
show ap dot11 5ghz ccx	Displays 802.11a CCX information for all Cisco APs.
show ap dot11 5ghz channel	Displays the configuration and statistics of the 802.11a channel assignment.
show ap dot11 5ghz coverage	Displays the configuration and statistics of the 802.11a coverage.
show ap dot11 5ghz group	Displays the configuration and statistics of the 802.11a grouping.
show ap dot11 5ghz l2roam	Displays 802.11a l2roam information.
show ap dot11 5ghz logging	Displays the configuration and statistics of the 802.11a event logging.
show ap dot11 5ghz monitor	Displays the configuration and statistics of the 802.11a monitoring.
show ap dot11 5ghz profile	Displays 802.11a profiling information for all Cisco APs.
show ap dot11 5ghz receiver	Displays the configuration and statistics of the 802.11a receiver.

Commands	Description
show ap dot11 5ghz summary	Displays the configuration and statistics of the 802.11a Cisco APs.
show ap dot11 5ghz txpower	Displays the configuration and statistics of the 802.11a transmit power control.

Additional References

Related Documents

Related Topic	Document Title
RRM commands and their details	*RRM Command Reference, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*

MIBs

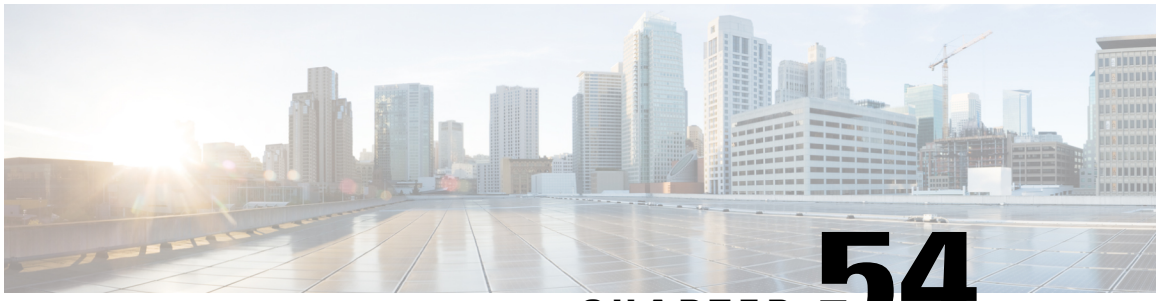
MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

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

PART XI

Lightweight Access Points



CHAPTER 54

Configuring the Controller for Access Point Discovery

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Controller for Access Point Discovery

- Ensure that the Control and Provisioning of Wireless Access Points (CAPWAP) UDP ports 5246 and 5247 (similar to the Lightweight Access Point Protocol (LWAPP) UDP ports 12222 and 12223) are enabled and are not blocked by an intermediate device that could prevent an access point from joining the controller.
- If access control lists (ACLs) are in the control path between the controller and its access points, you must open new protocol ports to prevent access points from being stranded.
- If an access point is in the UP state and its IP address changes, the access point tears down the existing CAPWAP tunnel and rejoins the controller.

- Access points must be discovered by a controller before they can become an active part of the network. The lightweight access points support the following controller discovery processes:
  - Layer 3 CAPWAP discovery—You can enable this feature on different subnets from the access point. This feature uses IP addresses and UDP packets rather the MAC addresses used by Layer 2 discovery.
  - Locally stored controller IP address discovery—If the access point was previously associated to a controller, the IP addresses of the primary, secondary, and tertiary controllers are stored in the access point's nonvolatile memory. This process of storing controller IP addresses on an access point for later deployment is called *priming the access point*.
  - DHCP server discovery—This feature uses DHCP option 43 to provide controller IP addresses to the access points. Cisco switches support a DHCP server option that is typically used for this capability.
  - DNS discovery—The access point can discover controllers through your domain name server (DNS). You must configure your DNS to return controller IP addresses in response to CISCO-CAPWAP-CONTROLLER.*localdomain*, where *localdomain* is the access point domain name. When an access point receives an IP address and DNS information from a DHCP server, it contacts the DNS to resolve CISCO-CAPWAP-CONTROLLER.*localdomain*. When the DNS sends a list of controller IP addresses, the access point sends discovery requests to the controllers.

Restrictions for Configuring the Controller for Access Point Discovery

- Ensure that the controllers are configured with the correct date and time. If the date and time configured on the controller precedes the creation and installation date of certificates on the access points, the access point fails to join the controller.
- During the discovery process, access points that are supported by the Cisco controller, such as the 1140, 1260, 3500, 1040,1600, 2600, or 3600 query only for Cisco controllers.

Information About Configuring the Controller for Access Point Discovery

In a CAPWAP environment, a lightweight access point discovers a controller by using CAPWAP discovery mechanisms and then sends a CAPWAP join request to the controller. The controller sends a CAPWAP join response to the access point that allows the access point to join the controller. When the access point joins the controller, the controller manages its configuration, firmware, control transactions, and data transactions.

Access Point Communication Protocols

Cisco lightweight access points use the IETF standard CAPWAP to communicate with the controller and other lightweight access points on the network.

CAPWAP, which is based on LWAPP, is a standard, interoperable protocol that enables a controller to manage a collection of wireless access points. CAPWAP is implemented in controller for these reasons:

- To provide an upgrade path from Cisco products that use LWAPP to next-generation Cisco products that use CAPWAP

- To manage RFID readers and similar devices
- To enable controllers to interoperate with third-party access points in the future

Viewing Access Point Join Information

Join statistics for an access point that sends a CAPWAP discovery request to the controller at least once are maintained on the controller even if the access point is rebooted or disconnected. These statistics are removed only when the controller is rebooted or when you choose to clear the statistics.

Troubleshooting the Access Point Join Process

Access points can fail to join a controller for many reasons such as a RADIUS authorization is pending, self-signed certificates are not enabled on the controller, the access point and controller's regulatory domains do not match, and so on.

You can configure the access points to send all CAPWAP-related errors to a syslog server. You do not need to enable any debug commands on the controller because all of the CAPWAP error messages can be viewed from the syslog server itself.

The state of the access point is not maintained on the controller until it receives a CAPWAP join request from the access point, so it can be difficult to determine why the CAPWAP discovery request from a certain access point was rejected. In order to troubleshoot such joining issues without enabling CAPWAP debug commands on the controller, the controller collects information for all access points that send a discovery message to this controller and maintains information for any access points that have successfully joined this controller.

The controller collects all join-related information for each access point that sends a CAPWAP discovery request to the controller. Collection begins when the first discovery message is received from the access point and ends when the last configuration payload is sent from the controller to the access point.

When the controller is maintaining join-related information for the maximum number of access points, it does not collect information for any more access points.

You can also configure a DHCP server to return a syslog server IP address to the access point using option 7 on the server. The access point then starts sending all syslog messages to this IP address.

You can configure the syslog server IP address through the access point CLI, if the access point is not connected to the controller by entering the **capwap ap log-server** *syslog_server_IP_address* command.

When the access point joins a controller for the first time, the controller pushes the global syslog server IP address (the default is 255.255.255.255) to the access point. After that, the access point sends all syslog messages to this IP address, until it is overridden by one of the following scenarios:

- The access point is still connected to the same controller, and you changed the global syslog server IP address configuration on the controller by using the **ap syslog host** *Syslog_Server_IP_Address* command. In this case, the controller pushes the new global syslog server IP address to the access point.
- The access point is still connected to the same controller, and you configured a specific syslog server IP address for the access point on the controller by using the **ap name** *Cisco_AP* **syslog host** *Syslog_Host_IP_Address* command. In this case, the controller pushes the new specific syslog server IP address to the access point.
- The access point gets disconnected from the controller, and you configured the syslog server IP address from the access point CLI by using the **capwap ap log-server** *syslog_server_IP_address* command. This command works only if the access point is not connected to any controller.

- The access point gets disconnected from the controller and joins another controller. In this case, the new controller pushes its global syslog server IP address to the access point.

Whenever a new syslog server IP address overrides the existing syslog server IP address, the old address is erased from persistent storage, and the new address is stored in its place. The access point also starts sending all syslog messages to the new IP address, if the access point can reach the syslog server IP address.

How to Configure Access Point Discovery

Configuring the Controller for Access Point Discovery (CLI)

Note The procedure to perform this task using the controller GUI is not currently available.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ap capwap master**
4. **end**
5. (Optional) **clear arp-cache**
6. (Optional) **clear mac address-table dynamic** [**address** *mac-address*]
7. **configure terminal**
8. **no ap capwap master**
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Controller# enable`	Enters privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 3	**ap capwap master** **Example:** `Controller(config)# ap capwap master`	Configures the new controller as a master controller.

	Command or Action	Purpose
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.
Step 5	**clear arp-cache** **Example:** `Controller# clear arp-cache`	(Optional) Clears the entire Address Resolution Protocol (ARP) cache.
Step 6	**clear mac address-table dynamic** [**address** *mac-address*] **Example:** `Controller# clear mac address-table dynamic`	(Optional) Clears the MAC forwarding table. **Note** After you clear the MAC forwarding table, you must restart the access point.
Step 7	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 8	**no ap capwap master** **Example:** `Controller(config)# no ap capwap master`	Configures the controller not to be a master controller. **Note** You must enter this command after all the access points have joined the new controller.
Step 9	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring the Syslog Server for Access Points

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ap syslog host** *host_ip_address*
4. **end**
5. **show ap config dot11 24ghz global**
6. **show ap name** *Cisco_AP* **config general**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Controller# enable`	Enters privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 3	**ap syslog host** *host_ip_address* **Example:** `Controller(config)# ap syslog host 10.9.9.16`	Configures the global syslog server for all access points that join this controller. **Note** By default, the global syslog server IP address for all access points is 255.255.255.255. Make sure that the access points can reach the subnet on which the syslog server resides before configuring the syslog server on the controller. If the access points cannot reach this subnet, the access points are unable to send out syslog messages.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.
Step 5	**show ap config dot11 24ghz global** **Example:** `Controller# show ap config dot11 24ghz global`	Displays the global syslog server settings for all access points that join the controller.
Step 6	**show ap name** *Cisco_AP* **config general** **Example:** `Controller# show ap name AP03 config general`	Displays the syslog server settings for a specific access point.

Monitoring Access Point Join Information (CLI)

Note The procedure to perform this task using the controller GUI is not currently available.

SUMMARY STEPS

1. **enable**
2. **show ap join stats summary**
3. **show ap mac-address** *mac_address* **join stats summary**
4. **show ap mac-address** *mac_address* **join stats detailed**
5. **clear ap join statistics**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Controller# enable`	Enters privileged EXEC mode.
Step 2	**show ap join stats summary** **Example:** `Controller# show ap join stats summary`	Displays the MAC addresses of all the access points that are joined to the controller or that have tried to join.
Step 3	**show ap mac-address** *mac_address* **join stats summary** **Example:** `Controller# show ap mac-address 000.2000.0400 join stats summary`	Displays the last join error detail for a specific access point.
Step 4	**show ap mac-address** *mac_address* **join stats detailed** **Example:** `Controller# show ap mac-address 000.2000.0400 join stats detailed`	Displays all join-related statistics collected for a specific access point.
Step 5	**clear ap join statistics** **Example:** `Controller# clear ap join statistics`	Clears the join statistics for all access points. **Note** To clear the join statistics that correspond to specific access points, enter the **clear ap mac-address** *mac_address* **join statistics** command.

Searching for Access Point Radios (GUI)

Step 1 Perform either of the following:

- Choose **Monitor > AP Summary** and then click the 802.11a/n Radios link or the 802.11b/g/n Radios link.
- Choose **Monitor > 802.11a/n** or **Monitor > 802.11b/g/n**.

Depending on the option that you choose to perform, either the **802.11a/n Radios** page or the **802.11b/g/n Radios** page appears. These pages show all of the 802.11a/n or 802.11b/g/n access point radios that are joined to the controller and their current settings in a tabular format.

Note In a Cisco converged access environment, the 802.11a and 802.11b/g radios should not be differentiated based on their Base Radio MAC addresses, because they might have the same addresses. Instead, the radios should be differentiated based on their physical addresses.

Step 2 From the **Show** drop-down list that appears at the top right corner of the page, choose **Quick Filter**.
The filter options (text boxes) appear in each of the column header in the table.

Step 3 Enter a keyword in the following text boxes to specify the filtering criteria based on which you want to perform the search operation:

- **AP Name—**Enter an access point name as the filtering criteria to perform the search operation..

  Note When you enable the MAC address filter, the other filters are disabled automatically. When you enable any of the other filters, the MAC address filter is disabled automatically.

- **Base MAC Radio—**Enter the Base radio MAC address of an access point radio as the filtering criteria to perform the search operation.
- **Operational Status—**Enter the operational status that corresponds to the radio.
- **Channel—**Enter the channel information that corresponds to the radio as the filtering criteria to perform the search operation.
- **Power Level—**Enter the power level of the radio as the filtering criteria to perform the search operation.
- **Admin Status—**Enter the administrative status that corresponds to the radio as the filtering criteria to perform the search operation.

Step 4 Click the **Filter** icon that appears next to the **Show** drop-down list to commit your changes.

Note If you want to remove the filter and display the entire access point radio list, click **Clear Filter**.

Only the access point radios that match your search criteria appear on the **802.11a/n Radios** page or the **802.11b/g/n Radios** page, and the Current Filter parameter at the top of the page specifies the filter used to generate the list (for example, MAC Address:00:1e:f7:75:0a:a0 or AP Name:pmsk-ap).

Monitoring the Interface Details

Step 1 Choose **Configuration > AP Summary**.
The **All APs** page appears and displays a list of access points that are associated to the controller.

Step 2 Select the access point for which you want to monitor the interface details.
The **AP > Edit** page appears.

Step 3 Click the **Interface** tab.
The interface details appear and displays the following parameters:

Table 109: Interfaces Parameters Details

Button	Description
Interface	Name of the interface.
Operational Status	Operational state of the physical Ethernet interface on the access point.
RX Bytes	Total number of bytes in the error-free packets received on the interface.
RX Unicast Packets	Total number of unicast packets received on the interface.
RX Non-Unicast Packets	Total number of non-unicast or multicast packets received on the interface.
TX Bytes	Number of bytes in the error-free packets transmitted on the interface.
TX Unicast Packets	Total number of unicast packets transmitted on the interface.
TX Non-Unicast Packets	Total number of non-unicast or multicast packets transmitted on the interface.
Radio Slot	Slot number of the radio interface.
Radio Interface Type	Type of the radio interface.
Sub Band	Sub-band details of the radio interface.
Admin Status	Administrative status of the radio interface.
Oper Status	Operational status of the radio interface.
CleanAir Admin Status	Administrative status that corresponds to the CleanAir feature on the radio interface.
CleanAir Oper Status	Operational status that corresponds to the CleanAir feature on the radio interface.
Regulatory Domain	Information that corresponds to the regulatory domain of the radio interface.

Configuration Examples for Configuring the Controller for Access Point Discovery

Displaying the MAC Addresses of all Access Points: Example

This example shows how to display MAC addresses of all the access points that are joined to the controller:

```
Controller# show ap join stats summary
Number of APs.......................................... 4

Base Mac          EthernetMac       AP Name IP Address    Status
----------------- ----------------- ------- ------------- ----------
00:0b:85:57:bc:c0 00:0b:85:57:bc:c0 AP1130  10.10.163.217 Joined
00:1c:0f:81:db:80 00:1c:63:23:ac:a0 AP1140  10.10.163.216 Not joined
00:1c:0f:81:fc:20 00:1b:d5:9f:7d:b2 AP1     10.10.163.215 Joined
00:21:1b:ea:36:60 00:0c:d4:8a:6b:c1 AP2     10.10.163.214 Not joined
```

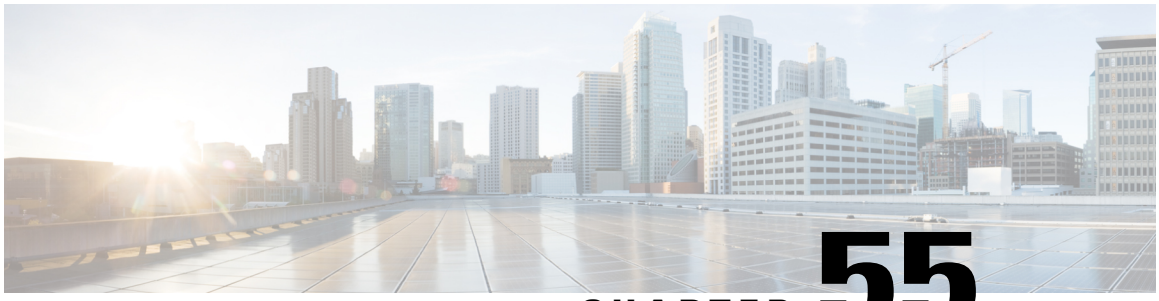
This example shows how to display the last join error details for a specific access point:

```
Controller# show ap mac-address 000.2000.0400 join stats summary
Is the AP currently connected to controller..... Yes
Time at which the AP joined this
controller last time................................ Aug 21 12:50:36.061
Type of error
that occurred last.................. AP got or has been disconnected
Reason for error
that occurred last........... The AP has been reset by the controller
Time at which the last join error occurred......... Aug 21 12:50:34.374
```

This example shows how to display all join-related statistics collected for a specific access point:

```
Controller# show ap mac-address 000.2000.0400 join stats detailed
Discovery phase statistics
- Discovery requests received.......................... 2
- Successful discovery responses sent.................. 2
- Unsuccessful discovery request processing.......... 0
- Reason for last unsuccessful discovery attempt..... Not applicable
- Time at last successful discovery attempt.......... Aug 21 12:50:23.335
- Time at last unsuccessful discovery attempt........ Not applicable

Join phase statistics
- Join requests received............................... 1
- Successful join responses sent....................... 1
- Unsuccessful join request processing................ 1
- Reason for last unsuccessful join attempt.....      RADIUS authorization
                                                      is pending
                                                      for the AP
- Time at last successful join attempt................ Aug 21 12:50:34.481
- Time at last unsuccessful join attempt.............. Aug 21 12:50:34.374

Configuration phase statistics
- Configuration requests received...................... 1
- Successful configuration responses sent.............. 1
- Unsuccessful configuration request processing....... 0
- Reason for last unsuccessful configuration attempt.. Not applicable
- Time at last successful configuration attempt....... Aug 21 12:50:34.374
- Time at last unsuccessful configuration attempt..... Not applicable

Last AP message decryption failure details
- Reason for last message decryption failure.......... Not applicable

Last AP disconnect details
- Reason for last AP connection failure............... The AP has been reset by
                                                      the controller
Last join error summary
- Type of error that occurred last.................... AP got or has been
                                                      disconnected
```

```
- Reason for error that occurred last.................. The AP has been reset
                                                       by the controller
- Time at which the last join error occurred.......... Aug 21 12:50:34.374
```



CHAPTER 55

Configuring Data Encryption

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Data Encryption

- Cisco 1260, 3500, 3600, 801, 1140, 1310, and 1520 series access points support Datagram Transport Layer Security (DTLS) data encryption.
- You can use the controller to enable or disable DTLS data encryption for a specific access point or for all access points.
- Non-Russian customers who use the Cisco controller do not need a data DTLS license. However, all customers who use WISM2 and Cisco 2500 Series controllers must enable data DTLS.

Restrictions for Configuring Data Encryption

- Encryption limits throughput at both the controller and the access point, and maximum throughput is desired for most enterprise networks.
- If your controller does not have a data DTLS license and if the access point associated with the controller has DTLS enabled, the data path will be unencrypted.
- In images that do not have a DTLS license, the **config** or **show** commands are not available.

Information About Data Encryption

The controller enables you to encrypt Control and Provisioning of Wireless Access Points (CAPWAP) control packets (and optionally, CAPWAP data packets) that are sent between the access point and the controller using DTLS. DTLS is a standards-track Internet Engineering Task Force (IETF) protocol based on TLS. CAPWAP control packets are management packets exchanged between a controller and an access point while CAPWAP data packets encapsulate forwarded wireless frames. CAPWAP control and data packets are sent over separate UDP ports: 5246 (control) and 5247 (data). If an access point does not support DTLS data encryption, DTLS is enabled only for the control plane, and a DTLS session for the data plane is not established.

How to Configure Data Encryption

Configuring Data Encryption (CLI)

SUMMARY STEPS

1. **configure terminal**
2. **ap link-encryption**
3. **end**
4. **show ap link-encryption**
5. **show wireless dtls connections**

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	**ap link-encryption** **Example:** `Controller(config)# ap link-encryption`	Enables data encryption for all access points or a specific access point by entering this command. The default value is disabled. Changing the data encryption mode requires the access points to rejoin the controller.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.
Step 4	**show ap link-encryption** **Example:** `Controller# show ap link-encryption`	Displays the encryption state of all access points or a specific access point. This command also shows authentication errors, which track the number of integrity check failures and replay errors. Relay errors help in tracking the number of times the access point receives the same packet.
Step 5	**show wireless dtls connections** **Example:** `Controller# show wireless dtls connections`	Displays a summary of all active DTLS connections. **Note** If you experience any problems with DTLS data encryption, enter the **debug dtls ap** {**all** | **event** | **trace**} command to debug all DTLS messages, events, or traces.

Configuring Data Encryption (GUI)

Before You Begin

Ensure that the base license is installed on the Cisco controller. After the license is installed, you can enable data encryption for the access points.

Step 1 Choose **Configuration** > **Wireless** > **AP Summary** to open the **All APs** page.

Step 2 Click the name of the access point for which you want to enable data encryption.

Step 3 Choose the **Advanced** tab to open the **AP > Edit (Advanced)** page.

Step 4 Select the **Data Encryption** check box to enable data encryption for this access point or unselect it to disable this feature. The default value is unselected.

Note Changing the data encryption mode requires the access points to rejoin the controller.

Step 5 Click **Apply** to commit your changes.

Configuration Examples for Configuring Data Encryption

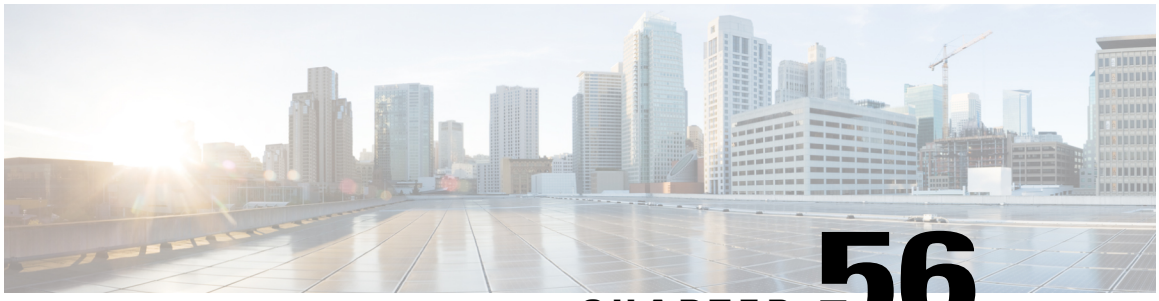
Displaying Data Encryption States for all Access Points: Examples

This example shows how to display the encryption state of all access points or a specific access point. This command also shows authentication errors, which track the number of integrity check failures and replay errors. Relay errors help in tracking the number of times the access point receives the same packet:

```
Controller# show ap link-encryption
                  Encryption  Dnstream  Upstream    Last
AP Name                State     Count     Count  Update
------------------  ---------  --------  --------  ------
3602a                 Enabled         0         0   Never
```

This example shows how to display a summary of all active DTLS connections:

```
Controller# show wireless dtls connections
AP Name         Local Port   Peer IP       Peer Port  Ciphersuite
--------------- ------------ ------------- ---------- --------------------
3602a           Capwap_Ctrl  10.10.21.213  46075      TLS_RSA_WITH_AES_128_CBC_SHA
3602a           Capwap_Data  10.10.21.213  46075      TLS_RSA_WITH_AES_128_CBC_SHA
```



CHAPTER 56

Configuring the Retransmission Interval and Retry Count

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Access Point Retransmission Interval and Retry Count

- You can configure the retransmission intervals and retry count both at a global and a specific access point level. A global configuration applies these configuration parameters to all the access points. Alternatively, when you configure the retransmission level and retry count at a specific access point level, the values are applied to that particular access point. The access point specific configuration has a higher precedence than the global configuration.

Information About Retransmission Interval and Retry Count

The controller and the access points exchange packets using the Control and Provisioning of Wireless Access Points (CAPWAP) reliable transport protocol. For each request, a response is defined. This response is used to acknowledge the receipt of the request message. Response messages are not explicitly acknowledged; therefore, if a response message is not received, the original request message is retransmitted after the retransmit interval. If the request is not acknowledged after a maximum number of retransmissions, the session is closed and the access points reassociate with another controller.

How to Configure Access Point Retransmission Interval and Retry Count

Configuring the Access Point Retransmission Interval and Retry Counts (CLI)

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ap capwap retransmit interval** *interval_time*
4. **ap capwap retransmit count** *count_value*
5. **end**
6. **ap name** *Cisco_AP* **capwap retransmit interval** *interval_time*
7. **ap name** *Cisco_AP* **capwap retransmit count** *count_value*
8. **show ap capwap retransmit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Controller# enable`	Enters privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 3	**ap capwap retransmit interval** *interval_time* **Example:** `Controller(config)# ap capwap retransmit interval 2`	Configures the control packet retransmit interval for all access points globally. **Note** The range for the interval parameter is from 2 to 5.

	Command or Action	Purpose
Step 4	**ap capwap retransmit count** *count_value* **Example:** `Controller(config)# ap capwap retransmit count 3`	Configures the control packet retry count for all access points globally. **Note** The range for the count is from 3 to 8.
Step 5	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.
Step 6	**ap name** *Cisco_AP* **capwap retransmit interval** *interval_time* **Example:** `Controller# ap name AP02 capwap retransmit interval 2`	Configures the control packet retransmit interval for the individual access point that you specify. **Note** The range for the interval is from 2 to 5.
Step 7	**ap name** *Cisco_AP* **capwap retransmit count** *count_value* **Example:** `Controller# ap name AP02 capwap retransmit count 3`	Configures the control packet retry count for the individual access point that you specify. **Note** The range for the retry count is from 3 to 8.
Step 8	**show ap capwap retransmit** **Example:** `Controller# show ap capwap retransmit`	Displays the CAPWAP retransmit details.

Configuring the Access Point Retransmission Interval and Retry Counts (GUI)

Step 1 Configure the controller to set the retransmission interval and retry count globally using the controller GUI as follows:

a) Choose **Monitor > AP Summary > Global AP Configuration**
The **Global Configuration** page appears.

b) Choose one of the following options under the **AP Transmit Config Parameters** area:

- **AP Retransmit Count—**Enter the number of times you want the access point to retransmit the request to the controller. The range is from 3 to 8.
- **AP Retransmit Interval—**Enter the time duration between the retransmission of requests. The range is from 2 to 5.

Step 2 Configure the controller to set the retransmission interval and retry count for a specific access point as follows:

a) Choose **Configuration** > **Wireless** > **AP Summary**.
The **All APs** page appears with a list of access points.

b) Click the access point for which you want to configure retransmit interval and retry count parameters.
The **AP > Edit** page appears.

c) Click **Advanced**.

d) In the **AP Retransmit Config Parameters** area, select the check box that corresponds to the **AP Retransmit Count** field.
The **AP Retransmit Count** field is enabled.

e) In the **AP Retransmit Count** field, enter the number of times that you want the access point to retransmit the request to the controller.
Note The range is from 2 to 5.

f) In the **AP Retransmit Config Parameters** area, select the check box that corresponds to the **AP Retransmit Interval** field.
The **AP Retransmit Interval** field is enabled.

g) In the **AP Retransmit Interval** field, enter the time duration between the retransmission of requests.
Note The range is from 3 to 8.

h) Click **Apply**.

Monitoring CAPWAP Maximum Transmission Unit Information (CLI)

Note The procedure to perform this task using the controller GUI is not currently available.

SUMMARY STEPS

1. **enable**
2. **show ap name** *Cisco_AP* **config general**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Controller# enable`	Enters privileged EXEC mode.
Step 2	**show ap name** *Cisco_AP* **config general** **Example:** `Controller# show ap name Maria-1250 config general | include MTU`	Displays the maximum transmission unit (MTU) for the CAPWAP path on the controller. The MTU specifies the maximum size of any packet (in bytes) in a transmission.

Debugging CAPWAP

Note The procedure to perform this task using the controller GUI is not currently available.

SUMMARY STEPS

1. **enable**
2. **debug capwap** {**ap all** | **cert-management** | **critical** | **detail** | **dtls-keepalive** | **error** | **events** | **ha** [**detail** | **dump** | **event** | **hex-dump** | **information** | **packet** | **payload** | **state**} [**switch** *switch_number*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Controller# enable`	Enters privileged EXEC mode.
Step 2	**debug capwap** {**ap all**	**cert-management**	**critical**	**detail**	**dtls-keepalive**	**error**	**events**	**ha** [**detail**	**dump**	**event**	**hex-dump**	**information**	**packet**	**payload**	**state**} [**switch** *switch_number*] **Example:** `Controller# debug capwap ap all`	Enables debugging of CAPWAP data. You can use any one of the following keywords with the command to enable debugging of various CAPWAP parameters: • **all—**Enables debugging all CAPWAP data. • **cert-management—**Enables debugging of CAPWAP certificate management. • **critical—**Enables debugging of critical CAPWAP data. • **details—**Enables debugging of CAPWAP details. • **dtls-keepalive—**Enables debugging of dtls keepalive packets. • **error—**Enables debugging of CAPWAP errors. • **events—**Enables debugging of CAPWAP events. • **ha—**Enables debugging of high availability in CAPWAP. • **hexdump—**Enables debugging of a CAPWAP hexadecimal dump. • **information—**Enables debugging of CAPWAP information. • **packets—**Enables debugging of CAPWAP packets. • **payload—**Enables debugging of a CAPWAP payload. • **state—**Enables debugging of CAPWAP states.

	Command or Action	Purpose

Configuration Examples for Configuring Access Point Retransmission Interval and Retry Count

Displaying the CAPWAP Retransmission Details: Example

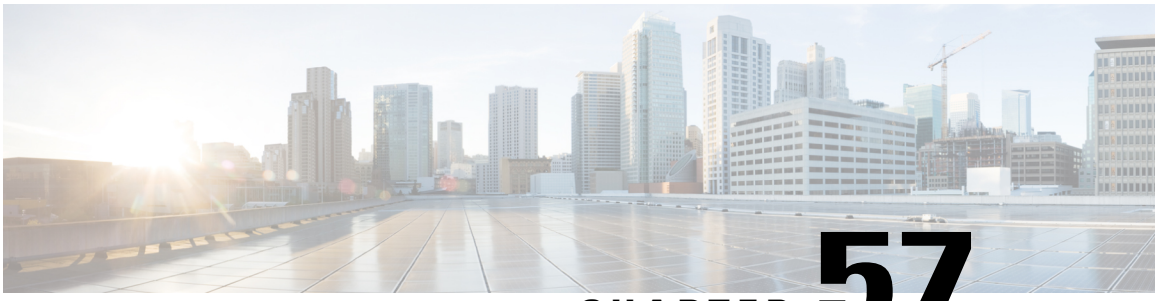
This example shows how to display the CAPWAP retransmit details:

```
Controller# show ap capwap retransmit
Global control packet retransmit interval : 3
Global control packet retransmit count : 5

AP Name                          Retransmit Interval             Retransmit Count

-------------------------------- --------------------------------
--------------------------------
3602a                            5                               3
```

Displaying Maximum Transmission Unit Information: Example

This example shows how to display the maximum transmission unit (MTU) for the CAPWAP path on the controller. The MTU specifies the maximum size of any packet (in bytes) in a transmission:

```
Controller# show ap name Maria-1250 config general | include MTU
CAPWAP Path MTU.................................. 1500
```



CHAPTER 57

Configuring Adaptive Wireless Intrusion Prevention System

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 wIPS

- The regular local mode access point has been extended with a subset of Wireless Intrusion Prevention System (wIPS) capabilities. This feature enables you to deploy your access points to provide protection without needing a separate overlay network.

How to Configure wIPS on Access Points

Configuring wIPS on an Access Point (CLI)

SUMMARY STEPS

1. **ap name** *Cisco_AP* **mode local**
2. **ap name** *Cisco_AP* **dot11 5ghz shutdown**
3. **ap name** *Cisco_AP* **dot11 24ghz shutdown**
4. **ap name** *Cisco_AP* **mode monitor submode wips**
5. **ap name** *Cisco_AP* **monitor-mode wips-optimized**
6. **show ap dot11 24ghz monitor**
7. **ap name** *Cisco_AP* **no dot11 5ghz shutdown**
8. **ap name** *Cisco_AP* **no dot11 24ghz shutdown**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**ap name** *Cisco_AP* **mode local** **Example:** `Controller# ap name AP01 mode local`	Configures an access point for monitor mode. A message appears that indicates that changing the AP's mode causes the access point to reboot. This message also displays a prompt that enables you to specify whether or not you want to continue with changing the AP mode. Enter **y** at the prompt to continue.
Step 2	**ap name** *Cisco_AP* **dot11 5ghz shutdown** **Example:** `Controller# ap name AP01 dot11 5ghz shutdown`	Disables the 802.11a radio on the access point.
Step 3	**ap name** *Cisco_AP* **dot11 24ghz shutdown** **Example:** `Controller# ap name AP02 dot11 24ghz shutdown`	Disables the 802.11b radio on the access point.
Step 4	**ap name** *Cisco_AP* **mode monitor submode wips** **Example:** `Controller# ap name AP01 mode monitor submode wips`	Configures the wIPS submode on the access point. **Note** To disable wIPS on the access point, enter the **ap name** *Cisco_AP* **modemonitor submode none** command.
Step 5	**ap name** *Cisco_AP* **monitor-mode wips-optimized**	Enables wIPS optimized channel scanning for the access point.

	Command or Action	Purpose
	Example: `Controller# ap name AP01 monitor-mode wips-optimized`	The access point scans each channel for 250 milliseconds. It derives the list of channels to be scanned from the monitor configuration. You can choose the following options: • **All—**All channels supported by the access point's radio. • **Country—**Only the channels supported by the access point's country of operation. • **DCA—**Only the channel set used by the dynamic channel assignment (DCA) algorithm, which by default includes all of the nonoverlapping channels allowed in the access point's country of operation.
Step 6	**show ap dot11 24ghz monitor** **Example:** `Controller# show ap dot11 24ghz monitor`	Displays the monitor configuration channel set. **Note** The 802.11b Monitor Channels value in the output of the command indicates the monitor configuration channel set.
Step 7	**ap name** *Cisco_AP* **no dot11 5ghz shutdown** **Example:** `Controller# ap name AP01 no dot11 5ghz shutdown`	Enables the 802.11a radio on the access point.
Step 8	**ap name** *Cisco_AP* **no dot11 24ghz shutdown** **Example:** `Controller# ap name AP01 no dot11 24ghz shutdown`	Enables the 802.11b radio on the access point.

Configuring wIPS on an Access Point (GUI)

Step 1 Choose **Configuration > AP Summary**.
The **All APs** page appears with a list of access points that are joined to the controller.

Step 2 Click the access point that you want to choose from the list.
The **AP > Edit** page appears.

Step 3 From the **AP Mode** drop-down list, choose one of the following options to configure the AP mode parameters:

- **Local**
- **Monitor**
- **Rogue Detector**
- **Sniffer**

- **Spectrum Expert**

Step 4 From the **AP Sub Mode** drop-down list, choose **WIPS** to set the AP sub mode parameters.

Step 5 Click **Apply**.

Monitoring wIPS Information

Note The procedure to perform this task using the controller GUI is not currently available.

SUMMARY STEPS

1. **show ap name** *Cisco_AP* **config general**
2. **show ap monitor-mode summary**
3. **show wireless wps wips summary**
4. **show wireless wps wips statistics**
5. **clear wireless wips statistics**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show ap name** *Cisco_AP* **config general** **Example:** `Controller# show ap name AP01 config general`	Displays information on the wIPS submode on the access point.
Step 2	**show ap monitor-mode summary** **Example:** `Controller# show ap monitor-mode summary`	Displays the wIPS optimized channel scanning configuration on the access point.
Step 3	**show wireless wps wips summary** **Example:** `Controller# show wireless wps wips summary`	Displays the wIPS configuration forwarded by NCS or Prime to the controller.
Step 4	**show wireless wps wips statistics** **Example:** `Controller# show wireless wps wips statistics`	Displays the current state of wIPS operation on the controller.
Step 5	**clear wireless wips statistics** **Example:** `Controller# clear wireless wips statistics`	Clears the wIPS statistics on the controller.

Configuration Examples for Configuring wIPS on Access Points

Displaying the Monitor Configuration Channel Set: Example

This example shows how to display the monitor configuration channel set:

```
Controller# show ap dot11 24ghz monitor
Default 802.11b AP monitoring
802.11b Monitor Mode.......................... enable
802.11b Monitor Channels...................... Country channels
802.11b AP Coverage Interval.................. 180 seconds
802.11b AP Load Interval...................... 60 seconds
802.11b AP Noise Interval..................... 180 seconds
802.11b AP Signal Strength Interval............ 60 seconds
```

Displaying wIPS Information: Examples

This example shows how to display information on the wIPS submode on the access point:

```
Controller# show ap name AP01 config general
Cisco AP Identifier.............. 3
Cisco AP Name.................... AP1131:46f2.98ac
...
AP Mode ......................... Monitor
Public Safety ................... Disabled Disabled
AP SubMode ...................... WIPS
```

This example shows how to display the wIPS optimized channel scanning configuration on the access point:

```
Controller# show ap monitor-mode summary
AP Name       Ethernet MAC   Status   Scanning
                                      Channel
                                      List
------------- -------------- -------- ---------
AP1131:4f2.9a 00:16:4:f2:9:a WIPS     1,6,NA,NA
```

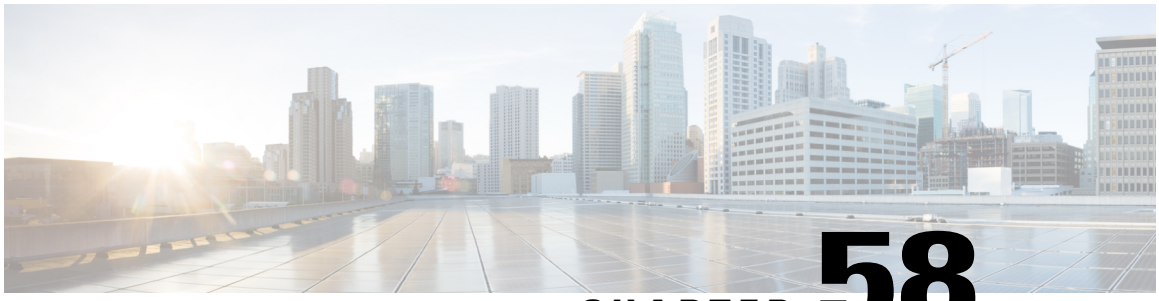
This example shows how to display the wIPS configuration forwarded by WCS to the controller:

```
Controller# show wireless wps wips summary
Policy Name.............. Default
Policy Version........... 3
```

This example shows how to display the current state of wIPS operation on the controller:

```
Controller# show wireless wps wips statistics
Policy Assignment Requests............ 1
Policy Assignment Responses........... 1
Policy Update Requests................ 0
Policy Update Responses............... 0
Policy Delete Requests................ 0
Policy Delete Responses............... 0
Alarm Updates......................... 13572
Device Updates........................ 8376
Device Update Requests................ 0
Device Update Responses............... 0
Forensic Updates...................... 1001
Invalid WIPS Payloads................. 0
Invalid Messages Received............. 0
CAPWAP Enqueue Failed................. 0
NMSP Enqueue Failed................... 0
NMSP Transmitted Packets.............. 22950
```

```
NMSP Transmit Packets Dropped..... 0
NMSP Largest Packet.................. 1377
```



CHAPTER 58

Configuring Authentication for Access Points

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Authentication for Access Points

- You can set a global username, password, and enable password for all access points that are currently joined to the controller and any that join in the future inherit as they join the controller. If desired, you can override the global credentials and assign a unique username, password, and enable password for a specific access point.
- After an access point joins the controller, the access point enables console port security, and you are prompted for your username and password whenever you log into the access point's console port. When you log in, you are in nonprivileged mode, and you must enter the enable password in order to use the privileged mode.
- The global credentials that you configure on the controller are retained across controller and access point reboots. They are overwritten only if the access point joins a new controller that is configured with a

global username and password. If the new controller is not configured with global credentials, the access point retains the global username and password configured for the first controller.

- You must track the credentials used by the access points. Otherwise, you might not be able to log into an access point's console port. If you need to return the access points to the default *Cisco/Cisco* username and password, you must clear the controller's configuration and the access point's configuration to return them to factory-default settings. To reset the default access point configuration, enter the **ap name** *Cisco_AP* **mgmtuser username** *Cisco* **password** *Cisco* command. Entering the command does not clear the static IP address of the access point. Once the access point rejoins a controller, it adopts the default *Cisco/Cisco* username and password.
- You can configure global authentication settings for all access points that are currently joined to the controller and any that join in the future. If desired, you can override the global authentication settings and assign unique authentication settings for a specific access point.
- This feature is supported on the following hardware:
  - All Cisco switches that support authentication.
  - Cisco Aironet 1260, 3500, 3600, 1140, 1310, and 1520 access points

Restrictions for Configuring Authentication for Access Points

- The controller name in the AP configuration is case sensitive. Therefore, make sure to configure the exact system name on the AP configuration. Failure to do this results in the AP fallback not working.
- The following access points are not supported by the Cisco controller:
  - All non 802.11 access points (also AP 1120) and AP 1250.

Information about Configuring Authentication for Access Points

Cisco IOS access points are shipped from the factory with *Cisco* as the default enable password. This password allows users to log into the nonprivileged mode and enter the **show** and **debug** commands that pose a security threat to your network. You must change the default enable password to prevent unauthorized access and to enable users to enter configuration commands from the access point's console port.

You can configure 802.1X authentication between a lightweight access point and a Cisco switch. The access point acts as an 802.1X supplicant and is authenticated by the switch where it uses EAP-FAST with anonymous PAC provisioning.

How to Configure Authentication for Access Points

Configuring Global Credentials for Access Points (CLI)

Note The procedure to perform this task using the controller GUI is not currently available.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ap mgmtuser username** *user_name* **password 0** *passsword* **secret 0** *secret_value*
4. **end**
5. **ap name** *Cisco_AP* **mgmtuser username** *user_name* **password** *password* **secret** *secret*
6. **show ap summary**
7. **show ap name** *Cisco_AP* **config dot11 24ghz general**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Controller# enable`	Enters privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.

Step 3	**ap mgmtuser username** *user_name* **password 0** *passsword* **secret 0** *secret_value*
	
	Example:
	`Controller(config)# ap mgmtuser apusr1`
	
	`password appass 0 secret 0 appass1`
		Configures the global username and password and enables the password for all access points that are currently joined to the controller and any access points that join the controller in the future. In the command, the parameter 0 specifies that an unencrypted password will follow and 8 specifies that an AES encrypted password will follow.

Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.
Step 5	**ap name** *Cisco_AP* **mgmtuser username** *user_name* **password** *password* **secret** *secret*	Overrides the global credentials for a specific access point and assigns a unique username and password and enables password to this access point.

	Command or Action	Purpose
	Example: `Controller(config)# ap name TSIM_AP-2 mgmtuser apusr1 password appass secret secret`	The credentials that you enter in this command are retained across controller and access point reboots and if the access point joins a new controller. **Note** If you want to force this access point to use the controller's global credentials, enter the **ap name** *Cisco_AP* **no mgmtuser** command. The following message appears after you execute this command: "AP reverted to global username configuration."
Step 6	**show ap summary** **Example:** `Controller# show ap summary`	Displays the global credentials configuration information that corresponds to all access points that join the controller. **Note** If global credentials are not configured, the Global AP User Name text box shows "Not Configured."
Step 7	**show ap name** *Cisco_AP* **config dot11 24ghz general** **Example:** `Controller# show ap name AP02 config dot11 24ghz general`	Displays the global credentials configuration for a specific access point. **Note** If this access point is configured for global credentials, the AP User Mode text boxes shows "Automatic." If the global credentials have been overwritten for this access point, the AP User Mode text box shows "Customized."

Configuring Authentication for Access Points (CLI)

Note The procedure to perform this task using the controller GUI is not currently available.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ap dot1x username** *user_name_value* **password 0** *password_value*
4. **end**
5. **ap name** *Cisco_AP* **dot1x-user username** *username_value* **password** *password_value*
6. **configure terminal**
7. **no ap dot1x username** *user_name_value* **password 0** *password_value*
8. **end**
9. **show ap summary**
10. **show ap name** *Cisco_AP* **config general**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Controller# enable`	Enters privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 3	**ap dot1x username** *user_name_value* **password 0** *password_value* **Example:** `Controller(config)# ap dot1x username AP3 password 0 password`	Configures the global authentication username and password for all access points that are currently joined to the controller and any access points that join the controller in the future. This command contains the following keywords and arguments: • **username—**Specifies an 802.1X username for all access points. • *user-id***—**Username. • **password—**Specifies an 802.1X password for all access points. • **0—**Specifies an unencrypted password. • **8—**Specifies an AES encrypted password. • *passwd***—**Password. **Note** You must enter a strong password for the password parameter. Strong passwords are at least eight characters long, contain a combination of uppercase and lowercase letters, numbers, and symbols, and are not a word in any language.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.
Step 5	**ap name** *Cisco_AP* **dot1x-user username** *username_value* **password** *password_value* **Example:** `Controller# ap name AP03 dot1x-user username apuser1 password appass`	Overrides the global authentication settings and assigns a unique username and password to a specific access point. This command contains the following keywords and arguments: • **username—**Specifies to add a username. • *user-id***—**Username. • **password—**Specifies to add a password. • **0—**Specifies an unencrypted password. • **8—**Specifies an AES encrypted password. • *passwd***—**Password.

	Command or Action	Purpose
		Note You must enter a strong password for the password parameter. See the note in Step 2 for the characteristics of strong passwords. The authentication settings that you enter in this command are retained across controller and access point reboots and whenever the access point joins a new controller.
Step 6	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 7	**no ap dot1x username** *user_name_value* **password 0** *password_value* **Example:** `Controller(config)# no ap dot1x username dot1xusr password 0 dot1xpass`	Disables 802.1X authentication for all access points or for a specific access point. The following message appears after you execute this command: "AP reverted to global username configuration." **Note** You can disable 802.1X authentication for a specific access point only if global 802.1X authentication is not enabled. If global 802.1X authentication is enabled, you can disable 802.1X for all access points only.
Step 8	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.
Step 9	**show ap summary** **Example:** `Controller# show ap summary`	Displays the authentication settings for all access points that join the controller. **Note** If global authentication settings are not configured, the Global AP Dot1x User Name text box shows "Not Configured."
Step 10	**show ap name** *Cisco_AP* **config general** **Example:** `Controller# show ap name AP02 config general`	Displays the authentication settings for a specific access point. **Note** If this access point is configured for global authentication, the AP Dot1x User Mode text boxes shows "Automatic." If the global authentication settings have been overwritten for this access point, the AP Dot1x User Mode text box shows "Customized."

Configuring the Switch for Authentication (CLI)

Note The procedure to perform this task using the controller GUI is not currently available.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **dot1x system-auth-control**
4. **aaa new-model**
5. **aaa authentication dot1x default group radius**
6. **radius-server host** *host_ip_adress* **acct-port** *port_number* **auth-port** *port_number* **key 0** *unencryptied_server_key*
7. **interface TenGigabitEthernet1/0/1**
8. **switch mode access**
9. **dot1x pae authenticator**
10. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Controller# enable`	Enters privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 3	**dot1x system-auth-control** **Example:** `Controller(config)# dot1x system-auth-control`	Enables system authentication control.
Step 4	**aaa new-model** **Example:** `Controller(config)# aaa new-model`	Enables new access control commands and functions.
Step 5	**aaa authentication dot1x default group radius** **Example:** `Controller(config)# aaa authentication dot1x default group radius`	Sets the default authentications lists for IEEE 802.1X by using all the radius hosts in a server group.
Step 6	**radius-server host** *host_ip_adress* **acct-port** *port_number* **auth-port** *port_number* **key 0** *unencryptied_server_key* **Example:** `Controller(config)# radius-server host 10.1.1.1 acct-port 1813 auth-port 6225 key 0 encryptkey`	Sets a clear text encryption key for the RADIUS authentication server.

	Command or Action	Purpose
Step 7	**interface TenGigabitEthernet1/0/1** **Example:** `Controller(config)# interface TenGigabitEthernet1/0/1`	Sets the 10-Gigbit Ethernet interface. The command prompt changes from Controller(config)# to Controller(config-if)#.
Step 8	**switch mode access** **Example:** `Controller(config-if)# switch mode access`	Sets the unconditional truncking mode access to the interface.
Step 9	**dot1x pae authenticator** **Example:** `Controller(config-if)# dot1x pae authenticator`	Sets the 802.1X interface PAE type as the authenticator.
Step 10	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuration Examples for Configuring Authentication for Access Points

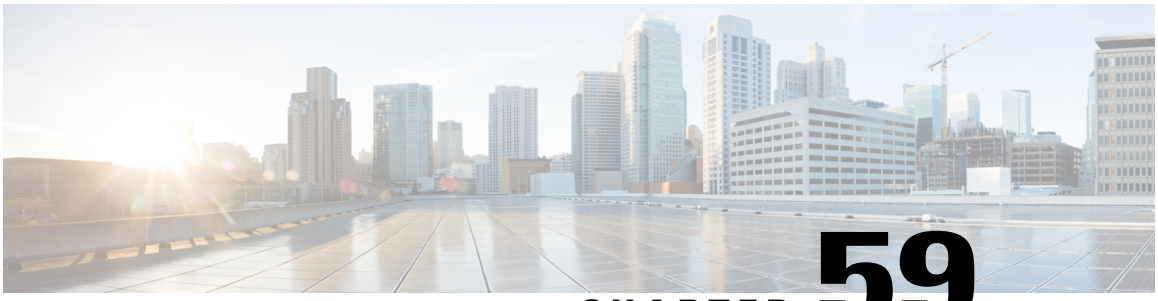
Displaying the Authentication Settings for Access Points: Examples

This example shows how to display the authentication settings for all access points that join the controller:

```
Controller# show ap summary
Number of APs.................................. 1
Global AP User Name............................ globalap
Global AP Dot1x User Name...................... globalDot1x
```

This example shows how to display the authentication settings for a specific access point:

```
Controller# show ap name AP02 config dot11 24ghz general
Cisco AP Identifier............................ 0
Cisco AP Name.................................. TSIM_AP2
...
AP Dot1x User Mode............................. AUTOMATIC
AP Dot1x User Name............................. globalDot1x
```



CHAPTER 59

Converting Autonomous Access Points to Lightweight Mode

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Converting Autonomous Access Points to Lightweight Mode

- Access points that are converted to lightweight mode do not support Wireless Domain Services (WDS). Converted access points communicate only with Cisco wireless LAN controllers and cannot communicate with WDS devices. However, the controller provides functionality that is equivalent to WDS when the access point associates to it.
- All Cisco lightweight access points support 16 Basic Service Set Identifiers (BSSIDs) per radio and a total of 16 wireless LANs per access point. When a converted access point associates to a controller, only wireless LANs with IDs 1 through 16 are pushed to the access point unless the access point is a member of an access point group.
- Access points that are converted to lightweight mode must get an IP address and discover the controller using DHCP, DNS, or IP subnet broadcast.

Information About Autonomous Access Points Converted to Lightweight Mode

You can convert autonomous Cisco Aironet access points to lightweight mode. When you upgrade the access points to lightweight mode, the access point communicates with the controller and receives a configuration and software image from the controller.

See the *Upgrading Autonomous Cisco Aironet Access Points to Lightweight Mode* document for instructions to upgrade an autonomous access point to lightweight mode:

http://www.cisco.com/en/US/docs/wireless/access_point/conversion/lwapp/upgrade/guide/lwapnote.html

Reverting from Lightweight Mode to Autonomous Mode

After you use the upgrade tool to convert an autonomous access point to lightweight mode, you can convert the access point from a lightweight unit back to an autonomous unit by loading a Cisco IOS release that supports autonomous mode (Cisco IOS Release 12.3(7)JA or earlier releases). If the access point is associated to a controller, you can use the controller to load the Cisco IOS release. If the access point is not associated to a controller, you can load the Cisco IOS release using TFTP. In either method, the access point must be able to access a TFTP server that contains the Cisco IOS release to be loaded.

Using DHCP Option 43 and DHCP Option 60

Cisco Aironet access points use the type-length-value (TLV) format for DHCP option 43. You must program the DHCP servers to return the option based on the access point's DHCP Vendor Class Identifier (VCI) string (DHCP option 60).

The following table lists the VCI strings for Cisco access points that can operate in lightweight mode.

Access Point	VCI String
Cisco Aironet 1140 Series	Cisco AP c1140
Cisco Aironet 3500 Series	Cisco AP c3500

Access Point	VCI String
Cisco Aironet 3600 Series	Cisco AP c3600

The format of the TLV block is as follows:

- Type: 0xf1 (decimal 241)
- Length: Number of controller IP addresses * 4
- Value: List of the IP addresses of controller management interfaces

See the product documentation for your DHCP server for instructions on configuring DHCP option 43. The *Upgrading Autonomous Cisco Aironet Access Points to Lightweight Mode* document contains example steps for configuring option 43 on a DHCP server.

If the access point is ordered with the Service Provider Option - AIR-OPT60-DHCP selected, the VCI string for that access point will be different than those strings listed in the previous table. The VCI string has the following suffix: ServiceProvider. For example, a 1260 with this option returns this VCI string: Cisco AP c1260-ServiceProvider.

Note The controller IP address that you obtain from the DHCP server should be a unicast IP address. Do not configure the controller IP address as a multicast address when configuring DHCP option 43.

How Converted Access Points Send Crash Information to the Controller

When a converted access point unexpectedly reboots, the access point stores a crash file on its local flash memory at the time of the crash. After the unit reboots, it sends the reason for the reboot to the controller. If the unit rebooted because of a crash, the controller pulls up the crash file using existing CAPWAP messages and stores it in the controller flash memory. The crash information copy is removed from the access point flash memory when the controller pulls it from the access point.

How Converted Access Points Send Radio Core Dump Information to the Controller

When a radio module in a converted access point generates a core dump, the access point stores the core dump file of the radio on its local flash memory at the time of the radio crash. It sends a notification message to the controller indicating which radio generated a core dump file. The controller sends a trap that alerts you so that you can retrieve the radio core file from the access point.

The retrieved core file is stored in the controller flash and can be uploaded through TFTP or FTP to an external server for analysis. The core file is removed from the access point flash memory when the controller pulls it from the access point.

Uploading Memory Core Dumps from Converted Access Points

By default, access points converted to lightweight mode do not send memory core dumps to the controller. This section provides instructions to upload access point core dumps using the controller GUI or CLI.

Displaying MAC Addresses for Converted Access Points

There are some differences in the way that controllers display the MAC addresses of converted access points on information pages in the controller GUI:

- On the AP Summary page, the controller lists the Ethernet MAC addresses of converted access points.
- On the AP Detail page, the controller lists the BSS MAC addresses and Ethernet MAC addresses of converted access points.
- On the Radio Summary page, the controller lists converted access points by the radio MAC address.

Configuring a Static IP Address for a Lightweight Access Point

If you want to specify an IP address for an access point rather than having one assigned automatically by a DHCP server, you can use the controller GUI or CLI to configure a static IP address for the access point. Static IP addresses are generally used only for deployments with a limited number of users.

An access point cannot discover the controller using domain name system (DNS) resolution if a static IP address is configured for the access point, unless you specify a DNS server and the domain to which the access point belongs. You can configure these parameters using either the controller CLI or the GUI.

Note If you configure an access point to use a static IP address that is not on the same subnet on which the access point's previous DHCP address was, the access point falls back to a DHCP address after the access point reboots. If the access point falls back to a DHCP address, enter the **show ap config general** *Cisco_AP* CLI command to show that the access point is using a fallback IP address. However, the GUI shows both the static IP address and the DHCP address, but it does not identify the DHCP address as a fallback address.

How to Revert to a Previous Release

Reverting to a Previous Release (CLI)

SUMMARY STEPS

1. **enable**
2. **ap name** *Cisco_AP* **tftp-downgrade** *tftp_server_ip_address tftp_server_image_filename*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Controller# enable`	Enters privileged EXEC mode.

	Command or Action	Purpose
Step 2	**ap name** *Cisco_AP* **tftp-downgrade** *tftp_server_ip_address tftp_server_image_filename* **Example:** `Controller# ap name AP02 tftp-downgrade 10.0.0.1 tsrvname`	Reverts the access point converted to lightweight mode to autonomous mode. **Note** After entering this command, you must wait until the access point reboots and then reconfigure the access point using the CLI or GUI.

Reverting to a Previous Release (Using the Mode Button and a TFTP Server)

Step 1 Configure the PC on which your TFTP server software runs with a static IP address in the range of 10.0.0.2 to 10.0.0.30.

Step 2 Make sure that the PC contains the access point image file (such as *c1140-k9w7-tar.123-7.JA.tar* for a 1140 series access point) in the TFTP server folder and that the TFTP server is activated.

Step 3 Rename the access point image file in the TFTP server folder to **c1140-k9w7-tar.default** for a 1140 series access point.

Step 4 Connect the PC to the access point using a Category 5 (CAT5) Ethernet cable.

Step 5 Disconnect power from the access point.

Step 6 Press and hold the **MODE** button while you reconnect power to the access point.

Note The **MODE** button on the access point must be enabled.

Step 7 Hold the **MODE** button until the status LED turns red (approximately 20 to 30 seconds), and release the MODE button.

Step 8 Wait until the access point reboots as indicated by all LEDs turning green followed by the Status LED blinking green.

Step 9 After the access point reboots, reconfigure the access point using the GUI or the CLI.

Authorizing Access Points (CLI)

Note The procedure to perform this task using the controller GUI is not currently available.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ap auth-list ap-policy authorize-ap**
4. **ap auth-list ap-policy mic**
5. **username** *user_name* **mac aaa attribute list** *list_name*
6. **aaa new-model**
7. **aaa authorization credential-download** *auth_list* **local**
8. **aaa attribute list** *list*
9. **aaa session-id common**
10. **aaa local authentication default authorization default**
11. **show ap name** *Cisco_AP* **config general**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Controller# enable`	Enters privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 3	**ap auth-list ap-policy authorize-ap** **Example:** `Controller(config)# ap auth-list ap-policy authorize-ap`	Configures an access point authorization policy.
Step 4	**ap auth-list ap-policy mic** **Example:** `Controller(config)# ap auth-list ap-policy mic`	Configures an access point to accept manufactured-installed certificates (MICs), self-signed certificates (SSCs), or local significant certificates (LSCs).
Step 5	**username** *user_name* **mac aaa attribute list** *list_name* **Example:** `Controller(config)# username aaa.bbb.ccc mac aaa attribute list attrlist`	Configures the MAC address of an access point locally.
Step 6	**aaa new-model** **Example:** `Controller(config)# aaa new-model`	Enables new access control commands and functions.

	Command or Action	Purpose
Step 7	**aaa authorization credential-download** *auth_list* **local** **Example:** `Controller(config)# aaa authorization credential-download auth_download local`	Downloads EAP credentials from the local server.
Step 8	**aaa attribute list** *list* **Example:** `Controller(config)# aaa attribute list alist`	Configures AAA attribute list definitions.
Step 9	**aaa session-id common** **Example:** `Controller(config)# aaa session-id common`	Configures the AAA common session ID.
Step 10	**aaa local authentication default authorization default** **Example:** `Controller(config)# aaa local authentication default authorization default`	Configures the local authentication method list.
Step 11	**show ap name** *Cisco_AP* **config general** **Example:** `Controller(config)# show ap name AP01 config general`	Displays the configuration information that corresponds to a specific access point.

Retrieving Radio Core Dumps (CLI)

Note The procedure to perform this task using the controller GUI is not currently available.

SUMMARY STEPS

1. **enable**
2. **ap name** *Cisco_AP* **crash-file get-radio-core-dump slot 0**
3. **show ap crash-file**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Controller# enable`	Enters privileged EXEC mode.
Step 2	**ap name** *Cisco_AP* **crash-file get-radio-core-dump slot 0** **Example:** `Controller# ap name AP02 crash-file get-radio-core-dump slot 0`	Transfers the radio core dump file from the access point to the controller. For the slot parameter, enter the slot ID of the radio that crashed.
Step 3	**show ap crash-file** **Example:** `Controller# show ap crash-file`	Displays access point crash file information. Using this command, you can verify whether the file is downloaded to the controller.

How to Upload Access Point Core Dumps

Uploading Access Point Core Dumps (CLI)

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ap core-dump** *tftp_server_ip_address tftp_server_image_filename* **compress**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Controller# enable`	Enters privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**ap core-dump** *tftp_server_ip_address tftp_server_image_filename* **compress** **Example:** `Controller(config)# ap core-dump 10.0.0.1 cdpname compress`	Uploads a core dump of the access point. The following parameters must be specified with the command: • *tftp_server_ip_address*—IP address of the TFTP server to which the access point sends core dump files. • *filename*—Name that the access points uses to label the core file. • **compress**—Configures the access point to send compressed core files. **Note** When you choose compress, the file is saved with a .gz extension (for example, dump.log.gz). This file can be opened with WinZip. • **uncompress**—Configures the access point to send uncompressed core files.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Uploading Access Point Core Dumps (GUI)

Step 1 Choose **Configuration > AP Summary.**
The **All APs** page appears with a list of access points.

Step 2 Click the access point for which you want to upload the core dumps.
The **AP > Edit** page appears.

Step 3 Click the **Advanced** tab.

Step 4 In the **AP Core Dump** area, select the **AP Core Dump** check box to upload a core dump of the access point.

Step 5 In the **TFTP Server IP** text box, enter the IP address of the TFTP server.

Step 6 In the **File Name** text box, enter a name of the access point core dump file (such as dump.log).

Step 7 Select the **File Compression** check box to compress the access point core dump file.
When you enable this option, the file is saved with a .gz extension (such as dump.log.gz). This file can be opened with WinZip.

Step 8 Click **Apply** to commit your changes.

Disabling the Reset Button on Converted Access Points (CLI)

You can enable or disable the Reset button on access points that are converted to lightweight mode. The Reset button is labeled MODE on the outside of the access point.

Note The procedure to perform this task using the controller GUI is not currently available.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no ap reset-button**
4. **end**
5. **ap name** *Cisco_AP* **reset-button**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Controller# enable`	Enters privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 3	**no ap reset-button** **Example:** `Controller(config)# no ap reset-button`	Disables the Reset buttons on all converted access points that are associated to the controller. **Note** To enable the Reset buttons on all converted access points that are associated to the controller, enter the **ap reset-button** command.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.
Step 5	**ap name** *Cisco_AP* **reset-button** **Example:** `Controller# ap name AP02 reset-button`	Enables the Reset button on the converted access point that you specify.

Monitoring the AP Crash Log Information

Note The procedure to perform this task using the controller GUI is not currently available.

SUMMARY STEPS

1. **enable**
2. **show ap crash-file**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Controller# enable`	Enters privileged EXEC mode.
Step 2	**show ap crash-file** **Example:** `Controller# show ap crash-file`	Verifies whether the crash file is downloaded to the controller.

How to Configure a Static IP Address on an Access Point

Configuring a Static IP Address on an Access Point (CLI)

SUMMARY STEPS

1. **enable**
2. **ap name** *Cisco_AP* **static-ip ip-address** *static_ap_address* **netmask** *static_ip_netmask* **gateway** *static_ip_gateway*
3. **enable**
4. **configure terminal**
5. **ap static-ip name-server** *nameserver_ip_address*
6. **ap static-ip domain** *static_ip_domain*
7. **end**
8. **show ap name** *Cisco_AP* **config dot11 24ghz general**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Controller# enable`	Enters privileged EXEC mode.
Step 2	**ap name** *Cisco_AP* **static-ip ip-address** *static_ap_address* **netmask** *static_ip_netmask* **gateway** *static_ip_gateway* **Example:** `Controller# ap name AP03 static-ip ip-address 9.9.9.16 netmask 255.255.0.0 gateway 9.9.9.2`	Configures a static IP address on the access point. This command contains the following keywords and arguments: • **ip-address—** Specifies the Cisco access point static IP address. • *ip-address***—** Cisco access point static IP address. • **netmask—**Specifies the Cisco access point static IP netmask. • *netmask***—** Cisco access point static IP netmask. • **gateway—**Specifies the Cisco access point gateway. • *gateway***—** IP address of the Cisco access point gateway. The access point reboots and rejoins the controller, and the static IP address that you specify is pushed to the access point. After the static IP address has been sent to the access point, you can configure the DNS server IP address and domain name. You must perform Steps 3 and 4 after the access points reboot.
Step 3	**enable** **Example:** `Controller# enable`	Enters privileged EXEC mode.
Step 4	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 5	**ap static-ip name-server** *nameserver_ip_address* **Example:** `Controller(config)# ap static-ip name-server 10.10.10.205`	Configures a DNS server so that a specific access point or all access points can discover the controller using DNS resolution. **Note** To undo the DNS server configuration, enter the **no ap static-ip name-server** *nameserver_ip_address* command.
Step 6	**ap static-ip domain** *static_ip_domain* **Example:** `Controller(config)# ap static-ip domain domain1`	Configures the domain to which a specific access point or all access points belong. **Note** To undo the domain name configuration, enter the **no ap static-ip domain** *static_ip_domain* command.

	Command or Action	Purpose
Step 7	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Step 8	**show ap name** *Cisco_AP* **config dot11 24ghz general**
	
	Example:
	`Controller# show ap name AP03 dot11 24ghz`
	`config general`
		Displays the IP address configuration for the access point.

Configuring a Static IP Address on an Access Point (GUI)

Step 1 Choose **Configuration** > **Wireless** > **AP Summary**
The **All APs** page appears with a list of all access points that are associated with the controller.

Step 2 Click the name of the access point for which you want to configure a static IP address.
The **AP > Edit** page appears.

Step 3 In the **IP Config** area, select the **Static IP** check box if you want to assign a static IP address to the access point. The default value is unselected.
Options that enable you to configure a static IP address for the access point appear in the **IP Config** area.

Step 4 In the **Netmask** field, enter the network mask.

Step 5 In the **Gateway** field, enter the default gateway address.

Step 6 Click **Apply** to commit your changes.
The access point reboots and rejoins the controller, and the static IP address that you specified in Step 4 is sent to the access point.

Step 7 After the static IP address has been sent to the access point, you can configure the DNS server IP address and domain name by performing the following steps:

a) In the **DNS IP Address** field, enter the IP Address of the DNS server.
b) In the **Domain Name** field, enter the name of the domain to which the access points belongs.
c) Click **Apply** to commit the changes.

Recovering the Access Point Using the TFTP Recovery Procedure

Step 1 Download the required recovery image from Cisco.com (ap3g2-k9w8-tar.152-2.JA.tar) and install it in the root directory of your TFTP server.

Step 2 Connect the TFTP server to the same subnet as the target access point and power-cycle the access point. The access point boots from the TFTP image and then joins the controller to download the oversized access point image and complete the upgrade procedure.

Step 3 After the access point has been recovered, you can remove the TFTP server.

Configuration Examples for Converting Autonomous Access Points to Lightweight Mode

Displaying LSC Information: Example

This example shows how to display the LSC summary:

```
Controller# show wireless certificate lsc summary
<?xml version="1.0"?>
 <iossr-response
  xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'>
  <cmd-response>
   <res0>
    <properties>
     <lscEnable type="boolean">false</lscEnable>
     <key_size type="unsignedInt">2048</key_size>
     <lscApProvision type="unsignedByte">0</lscApProvision>
     <rebootNum type="unsignedByte">3</rebootNum>
     <trustpoint
      type="string">default_lsc_trustpoint</trustpoint>
     <country type="string"></country>
     <state type="string"></state>
     <city type="string"></city>
     <orgn type="string"></orgn>
     <dept type="string"></dept>
     <email type="string"></email>
    </properties>
   </res0>
  </cmd-response>
 </iossr-response>
LSC Enabled                     : No
LSC AP-Provisioning             : No
TrustPoint                      : default_lsc_trustpoint
LSC Params:
    Country                     :
    State                       :
    City                        :
    Orgn                        :
    Dept                        :
    Email                       :
    KeySize                     : 2048
```

This example shows how to display details about the access points that are provisioned using LSC:

```
Controller# show wireless certificate lsc ap-provision
<?xml version="1.0"?>
 <iossr-response
  xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'>
  <cmd-response>
   <res0>
    <properties>
     <lscApProvision type="unsignedByte">0</lscApProvision>
    </properties>
   </res0>
   <res1>  </res1>
  </cmd-response>
 </iossr-response>
LSC AP-Provisioning                  : No
```

Displaying the IP Address Configuration for Access Points: Example

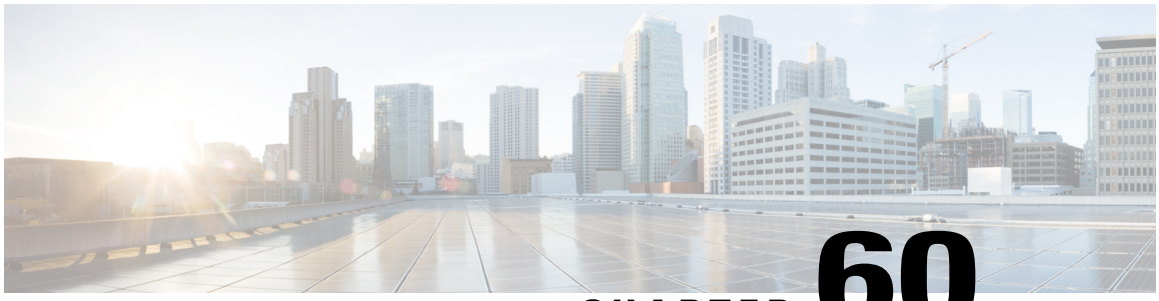
This example shows how to display the IP address configuration for the access point:

```
Controller# show ap name AP03 dot11 24ghz config general
Cisco AP Identifier.............. 4
Cisco AP Name.................................. AP6
IP Address Configuration................... Static IP assigned
IP Address..................................... 10.10.10.118
IP NetMask..................................... 255.255.255.0
Gateway IP Addr................................ 10.10.10.1
Domain......................................... Domain1
Name Server.................................... 10.10.10.205
...
```

Displaying Access Point Crash File Information: Example

This example shows how to display access point crash file information. Using this command, you can verify whether the file is downloaded to the controller:

```
Controller# show ap crash-file
Local Core Files:
lrad_AP1130.rdump0 (156)

The number in parentheses indicates the size of the file. The size should
be greater than zero if a core dump file is available.
```



CHAPTER 60

Using Cisco Workgroup Bridges

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Cisco Workgroup Bridges and non-Cisco Workgroup bridges

A WGB is a mode that can be configured on an autonomous Cisco IOS access point to provide wireless connectivity to a lightweight access point on behalf of clients that are connected by Ethernet to the WGB access point. A WGB connects a wired network over a single wireless segment by learning the MAC addresses of its wired clients on the Ethernet interface and reporting them to the lightweight access point using Internet Access Point Protocol (IAPP) messaging. The WGB provides wireless access connectivity to wired clients by establishing a single wireless connection to the lightweight access point.

When a Cisco WGB is used, the WGB informs the access points of all the clients that it is associated with. The controller is aware of the clients that are associated with the access point. When non-Cisco WGBs are used, the controller has no information about the IP address of the clients on the wired segment behind the WGB. Without this information, the controller drops the following types of messages:

- ARP REQ from the distribution system for the WGB client.
- ARP RPLY from the WGB client.
- DHCP REQ from the WGB client.

• DHCP RPLY for the WGB client.

Monitoring the Status of Workgroup Bridges

Note The procedure to perform this task using the controller GUI is not currently available.

SUMMARY STEPS

1. **enable**
2. **show wireless wgb summary**
3. **show wireless wgb mac-address** *wgb_mac_address* **detail**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Controller# enable`	Enters privileged EXEC mode.
Step 2	**show wireless wgb summary** **Example:** `Controller# show wireless wgb summary`	Displays the WGBs on your network.
Step 3	**show wireless wgb mac-address** *wgb_mac_address* **detail** **Example:** `Controller# show wireless wgb mac-address 00:0d:ed:dd:25:82 detail`	Displays the details of any wired clients that are connected to a particular WGB.

Debugging WGB Issues (CLI)

Note The procedure to perform this task using the controller GUI is not currently available.

SUMMARY STEPS

1. **enable**
2. **debug iapp all**
3. **debug iapp error**
4. **debug iapp packet**
5. **debug mobility handoff** [**switch** *switch_number*]
6. **debug dhcp**
7. **debug dot11 mobile**
8. **debug dot11 state**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Controller# enable`	Enters privileged EXEC mode.
Step 2	**debug iapp all** **Example:** `Controller# debug iapp all`	Enables debugging for IAPP messages.
Step 3	**debug iapp error** **Example:** `Controller# debug iapp error`	Enables debugging for IAPP error events.
Step 4	**debug iapp packet** **Example:** `Controller# debug iapp packet`	Enables debugging for IAPP packets.
Step 5	**debug mobility handoff** [**switch** *switch_number*] **Example:** `Controller# debug mobility handoff`	Enables debugging for any roaming issues.
Step 6	**debug dhcp** **Example:** `Controller# debug dhcp`	Debug an IP assignment issue when DHCP is used.
Step 7	**debug dot11 mobile** **Example:** `Controller# debug dot11 mobile`	Enables dot11/mobile debugging. Debug an IP assignment issue when static IP is used.

	Command or Action	Purpose
Step 8	**debug dot11 state** **Example:** `Controller# debug dot11 state`	Enables dot11/state debugging. Debug an IP assignment issue when static IP is used.

Configuration Examples for Configuring Workgroup Bridges

WGB Configuration: Example

This example shows how to configure a WGB access point using static WEP with a 40-bit WEP key:

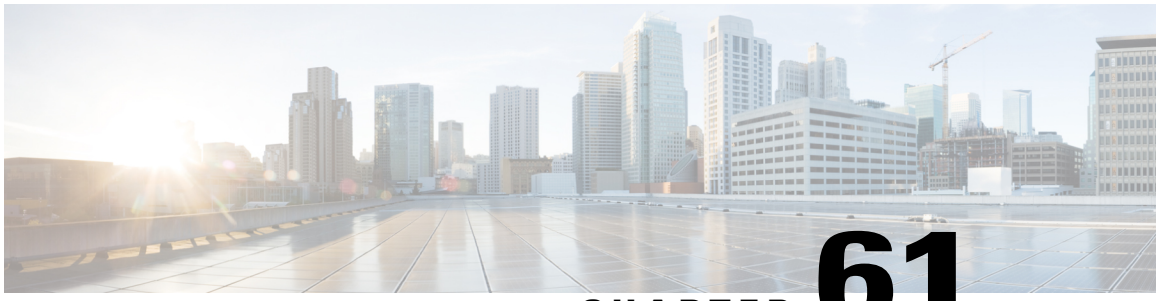
```
Controller# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Controller(config)# dot11 ssid WGB_with_static_WEP
Controller(config-ssid)# authentication open
Controller(config-ssid)# guest-mode
Controller(config-ssid)# exit
Controller(config)# interface  dot11Radio 0
Controller(config)# station-role workgroup-bridge
Controller(config-if)# encry mode wep 40
Controller(config-if)# encry key 1 size 40 0 1234567890
Controller(config-if)# ssid WGB_with_static_WEP
Controller(config-if)# end
```

Verify that the WGB is associated to an access point by entering this command on the WGB:

show dot11 association

Information similar to the following appears:

```
Controller# show dot11 associations
802.11 Client Stations on Dot11Radio0:
SSID [FCVTESTING] :
MAC Address    IP address      Device        Name            Parent         State
000b.8581.6aee 10.11.12.1      WGB-client    map1            -              Assoc
ap#
```



CHAPTER 61

Configuring Backup Controllers and Failover Priority for Access Points

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Backup Controllers and Failover Priority for Access Points

- You can configure primary and secondary backup controllers (which are used if primary, secondary, or tertiary controllers are not specified or are not responsive) for all access points that are connected to the controller as well as various timers, including heartbeat timers and discovery request timers. To reduce the controller failure detection time, you can configure the fast heartbeat interval (between the controller and the access point) with a smaller timeout value. When the fast heartbeat timer expires (at every

heartbeat interval), the access point determines if any data packets have been received from the controller within the last interval. If no packets have been received, the access point sends a fast echo request to the controller.

- The access point maintains a list of backup controllers and periodically sends primary discovery requests to each entry on the list. When the access point receives a new discovery response from a controller, the backup controller list is updated. Any controller that fails to respond to two consecutive primary discovery requests is removed from the list. If the access point's local controller fails, it chooses an available controller from the backup controller list in this order: primary, secondary, tertiary, primary backup, and secondary backup. The access point waits for a discovery response from the first available controller in the backup list and joins the controller if it receives a response within the time configured for the primary discovery request timer. If the time limit is reached, the access point assumes that the controller cannot be joined and waits for a discovery response from the next available controller in the list.
- When an access point's primary controller comes back online, the access point disassociates from the backup controller and reconnects to its primary controller. The access point falls back only to its primary controller and not to any available secondary controller for which it is configured. For example, if an access point is configured with primary, secondary, and tertiary controllers, it fails over to the tertiary controller when the primary and secondary controllers become unresponsive. If the secondary controller comes back online while the primary controller is down, the access point does not fall back to the secondary controller and stays connected to the tertiary controller. The access point waits until the primary controller comes back online to fall back from the tertiary controller to the primary controller. If the tertiary controller fails and the primary controller is still down, the access point then falls back to the available secondary controller.
- You can configure your wireless network so that the backup controller recognizes a join request from a higher-priority access point and if necessary disassociates a lower-priority access point as a means to provide an available port.
- You must enable failover priority on your network and assign priorities to the individual access points before you can configure this feature.

Restrictions for Configuring Backup Controllers and Failover Priority for Access Points

- You can configure the fast heartbeat timer only for access points in local mode.
- Failover priority is not in effect during the regular operation of your wireless network. It takes effect only if there are more association requests after a controller failure than there are available backup controller ports.
- By default, all access points are set to priority level 1, which is the lowest priority level. Therefore, you must assign a priority level only to those access points that warrant a higher priority.

Information About Configuring Backup Controllers

A single controller at a centralized location can act as a backup for access points when they lose connectivity with the primary controller in the local region. Centralized and regional controllers do not need to be in the same mobility group. You can specify a primary, secondary, and tertiary controller for specific access points

in your network. Using the controller CLI, you can specify the IP addresses of the backup controllers, which allows the access points to fail over to controllers outside of the mobility group.

Configuring Failover Priority for Access Points

Each controller has a defined number of communication ports for access points. When multiple controllers with unused access point ports are deployed on the same network and one controller fails, the dropped access points automatically poll for unused controller ports and associate with them.

Optimizing RFID Tracking on Access Points

To optimize the monitoring and location calculation of RFID tags, you can enable tracking optimization on up to four channels within the 2.4-GHz band of an 802.11b/g access point radio. This feature allows you to scan only the channels on which tags are usually programmed to operate (such as channels 1, 6, and 11).

Retrieving the Unique Device Identifier on Controllers and Access Points

The Unique Device Identifier (UDI) standard uniquely identifies products across all Cisco hardware product families, enabling customers to identify and track Cisco products throughout their business and network operations and to automate their asset management systems. The standard is consistent across all electronic, physical, and standard business communications. The UDI consists of five data elements:

- The orderable product identifier (PID)
- The version of the product identifier (VID)
- The serial number (SN)
- The entity name
- The product description

The UDI is burned into the EEPROM of controllers and lightweight access points at the factory. It can be retrieved through either the GUI or the CLI.

How to Configure Backup Controllers for Access Points

Configuring Backup Controllers for Access Points (CLI)

SUMMARY STEPS

1. **enable**
2. **ap name** *Cisco_AP* **controller primary** *primary_controller_name* [*primary_controller_ip_address*]
3. **ap name** *Cisco_AP* **controller secondary** *secondary_controller_name* [*secondary_controller_ip_address*]
4. **ap name** *Cisco_AP* **controller tertiary** *tertiary_controller_name* [*tertiary_controller_ip_adress*]
5. **configure terminal**
6. **ap capwap backup primary** *primary_backup_controller_name primary_backup_controller_ip_address*
7. **ap capwap backup secondary** *secondary_backup_controller_name secondary_backup_controller_ip_address*
8. **ap capwap timers fast-heartbeat-timeout** {**local** *timeout_interval*}
9. **ap capwap timers heartbeat-timeout** [*interval*].
10. **ap capwap timers primary-discovery-timeout** [*interval*].
11. **ap capwap timers discovery-timeout** [*interval*].
12. **end**
13. **show ap name** *Cisco_AP* **config general**
14. **show wireless client timers**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Controller# enable`	Enters privileged EXEC mode.

Step 2	**ap name** *Cisco_AP* **controller primary** *primary_controller_name* [*primary_controller_ip_address*]
	
	Example:
	`Controller# ap name AP02 controller primary`
	`pricon 10.0.0.1`
		Configures a primary controller for a specific access point.
		Note The *controller_ip_address* argument in Step 2 and Step 4 is optional. If the backup controller is outside the mobility group to which the access point is connected (the primary controller), you must provide the IP address of the primary, secondary, or tertiary controller, respectively. In each command, the *controller_name* and *controller_ip_address* must belong to the same primary, secondary, or tertiary controller. Otherwise, the access point cannot join the backup controller.

Step 3	**ap name** *Cisco_AP* **controller secondary** *secondary_controller_name* [*secondary_controller_ip_address*]	Configures a secondary controller for a specific access point.

	Command or Action	Purpose

	Example:
	`Controller# ap name AP02 controller secondary`
	`seccon 10.0.0.2`	

Step 4	**ap name** *Cisco_AP* **controller tertiary** *tertiary_controller_name* [*tertiary_controller_ip_adress*]
	Example:
	`Controller# ap name AP02 controller tertiary`
	`tercon 10.0.0.3`
		Configures a tertiary controller for a specific access point.

Step 5	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.

Step 6	**ap capwap backup primary** *primary_backup_controller_name primary_backup_controller_ip_address*
	Example:
	`Controller(config)# ap capwap backup primary`
	`advbackuppricon 10.0.0.3`
		Configures a primary backup controller for all access points.
		Note To delete the primary backup controller, enter the **no ap capwap backup primary** *primary_backup_controller_name primary_backup_controller_ip_address* command.

Step 7	**ap capwap backup secondary** *secondary_backup_controller_name secondary_backup_controller_ip_address*
	Example:
	`Controller(config)# ap capwap backup secondary`
	`advbackupseccon 10.0.0.4`
		Configures a secondary backup controller for all access points.
		Note To delete a secondary backup controller, enter the **no ap capwap backup secondary** *secondary_backup_controller_name secondary_backup_controller_ip_address* command.

Step 8	**ap capwap timers fast-heartbeat-timeout** {**local** *timeout_interval*} **Example:** `Controller(config)# ap capwap timers fast-heartbeat-timeout local 5`	Enables the fast heartbeat timer for local access points. **Note** The *timeout_Interval* is from 1 to 10 seconds (inclusive). Specifying a small heartbeat interval reduces the amount of time that it takes to detect a controller failure. The default value is disabled. **Note** To disable the fast heartbeat timer for local access points, enter the **no ap capwap timers fast-heartbeat-timeout** {**local** *timeout_interval*} command.
Step 9	**ap capwap timers heartbeat-timeout** [*interval*]. **Example:** `Controller(config)# ap capwap timers heartbeat-timeout 15`	Configures the access point heartbeat timer. **Note** The timeout *interval* is from 1 to 30 seconds (inclusive). This value should be at least three times larger than the fast heartbeat timer. The default value is 30 seconds. **Note** To disable the access point heartbeat timer, enter the **no ap capwap timers heartbeat-timeout** [*interval*] command.

	Command or Action	Purpose
		Caution Do not enable the fast heartbeat timer with the high latency link. If you have to enable the fast heartbeat timer, the timer value must be greater than the latency.
Step 10	**ap capwap timers primary-discovery-timeout** [*interval*]. **Example:** `Controller(config)# ap capwap timers primary-discovery-timeout 90`	Configures the access point primary discovery request timer. **Note** The timeout *interval* is from 30 to 3600 seconds. The default is 120 seconds. **Note** To disable the access point primary discovery request timer, enter the **no ap capwap timers primary-discovery-timeout** [*interval*] command.
Step 11	**ap capwap timers discovery-timeout** [*interval*]. **Example:** `Controller(config)# ap capwap timers discovery-timeout 9`	Configures the access point discovery timer. **Note** The timeout *interval* is from 1 to 10 seconds (inclusive). The default is 10 seconds. **Note** To disable the access point discovery timer, enter the **no ap capwap timers discovery-timeout** [*interval*] command.
Step 12	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.
Step 13	**show ap name** *Cisco_AP* **config general** **Example:** `Controller# show ap name AP02 config general`	Displays access point configuration information.
Step 14	**show wireless client timers** **Example:** `Controller# show wireless client timers`	Displays the wireless client timer information.

Configuring Backup Controllers for Access Points (GUI)

Step 1 Choose **Monitor > AP Summary > Global AP Configuration**.
The **Global Configuration** page appears.

Step 2 From the **Local Mode AP Fast Heartbeat Timer State** drop-down list, choose **Enable** to enable the fast heartbeat timer for access points in local mode or choose **Disable** to disable this timer. The default value is Disable.

Note If you choose **Enable** from the **Local Mode AP Fast Heartbeat Timer State** drop-down list, enter a value in the **Local Mode AP Fast Heartbeat Timeout** text box to configure the fast heartbeat timer for access points in local mode. Specifying a small heartbeat interval reduces the amount of time it takes to detect a controller failure.

The range for the AP Fast Heartbeat Timeout value is from 1 to 10 (inclusive) for controllers and the default is 1 second.

Step 3 In the **AP Primary Discovery Timeout** text box, enter a value from 30 to 3600 seconds (inclusive) to configure the access point primary discovery request timer. The default is 120 seconds.

Step 4 Configure primary, secondary, and tertiary backup controllers for a specific access point as follows:

a) Choose **Configuration > AP Summary**.
The **ALL APs** page appears with a list of access points that are associated to the controller.

b) Click the name of the access point for which you want to configure primary, secondary, and tertiary backup controllers.
The **AP > Edit** page appears.

c) Click the **High Availability** tab.

d) In the **Primary Controller** text box, enter the name and IP address of the primary controller for this access point.

Note Entering an IP address for the backup controller is optional in this step and the next two steps. If the backup controller is outside the mobility group to which the access point is connected (the primary controller), you must provide the IP address of the primary, secondary, or tertiary controller, respectively. The controller name and IP address must belong to the same primary, secondary, or tertiary controller. Otherwise, the access point cannot join the backup controller.

e) In the **Secondary Controller** text box, enter the name and IP address of the secondary controller for the access point.

f) In the **Tertiary Controller** text box, enter the name and IP address of the tertiary controller for the access point.

g) Click **Apply** to commit your changes.

How to Configure Failover Priority for Access Points

Configuring Failover Priority for Access Points (CLI)

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ap capwap priority**
4. **end**
5. **ap name** *Cisco_AP* {**priority** *priority_value*}

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Controller# enable`	Enters privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 3	**ap capwap priority** **Example:** `Controller(config)# ap capwap priority`	Enables the access point failover priority. **Note** To disable access point failover priority, enter the **no ap capwap priority** command.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.
Step 5	**ap name** *Cisco_AP* {**priority** *priority_value*} **Example:** `Controller# ap name AP02 priority 140`	Specifies the priority of an access point. **Note** You can enter a value from 1 to 4 for the priority value parameter.

Configuring Failover Priority for Access Points (GUI)

Step 1 Choose **Configuration > AP Summary**.
The **ALL APs** page appears.

Step 2 Click the name of the access point for which you want to configure failover priority.
The **AP > Edit** page appears.

Step 3 Click the **High Availability** tab.
The High Availability options appear.

Step 4 From the **AP Failover Priority** drop-down list, choose one of the following options to specify the priority of the access point:

- **Low—**Assigns the access point to the level 1 priority, which is the lowest priority level. This is the default value.
- **Medium—**Assigns the access point to the level 2 priority.
- **High—**Assigns the access point to the level 3 priority.
- **Critical—**Assigns the access point to the level 4 priority, which is the highest priority level.

Step 5 Click **Apply** to commit your changes.

Monitoring Failover Priority Settings (CLI)

Note The procedure to perform this task using the controller GUI is not currently available.

SUMMARY STEPS

1. **enable**
2. **show ap capwap summary**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Controller# enable`	Enters privileged EXEC mode.
Step 2	**show ap capwap summary** **Example:** `Controller# show ap capwap summary`	Displays access point capwap summary. Using this command, you can confirm whether the access point failover priority is enabled on your network.

Configuration Examples for Configuring Backup Controllers and Failover Priority for Access Points

Displaying Access Point Configuration Information: Examples

This example shows how to display access point configuration information:

```
Controller# show ap name AP01 config general

Cisco AP Identifier : 0
Cisco AP Name : AP01
Country Code : US - United States
Regulatory Domain Allowed by Country : 802.11bg:-A
802.11a:-A
AP Country Code : US - United States
AP Regulatory Domain : Unconfigured
Switch Port Number : Te1/0/1
MAC Address : 0000.2000.03f0
```

```
IP Address Configuration : Static IP assigned
IP Address : 9.9.9.16
...... ........
...... ........
Primary Cisco Switch Name : 1-4404
Primary Cisco Switch IP Address : 2.2.2.2
Secondary Cisco Switch Name : 1-4404
Secondary Cisco Switch IP Address : 2.2.2.2
Tertiary Cisco Switch Name : 2-4404
Tertiary Cisco Switch IP Address : 1.1.1.4
```

Displaying Wireless Client Timer Information

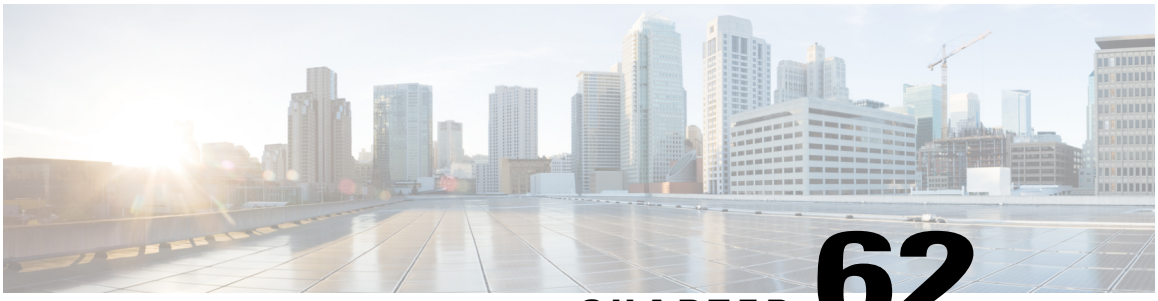
This example shows how to display wireless client timer information:

```
Controller# show wireless client timers

Authentication Response Timeout (seconds) : 10
Rogue Entry Timeout (seconds) : 1300
AP Heart Beat Timeout (seconds) : 30
AP Discovery Timeout (seconds) : 10
AP Local mode Fast Heartbeat (seconds) : 10 (enable)
AP flexconnect mode Fast Heartbeat (seconds) : disable
AP Primary Discovery Timeout (seconds) : 120
```

Displaying Access Point CAPWAP Summary: Example

This example shows how to display access point CAPWAP summary. Using this command, you can confirm whether or not the access point failover priority is enabled on your network.

```
Controller# show ap capwap summary

AP Fallback : Enabled
AP Join Priority : Disabled
AP Master : Disabled
Primary backup Controller Name :
Primary backup Controller IP : 0.0.0.0
Secondary backup Controller Name :
Secondary backup Controller IP : 0.0.0.0
```



CHAPTER 62

Configuring Probe Request Forwarding

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Configuring Probe Request Forwarding

Probe requests are 802.11 management frames that are sent by clients to request information about the capabilities of Service Set Identifiers (SSIDs). By default, access points forward acknowledged probe requests to the controller for processing. Acknowledged probe requests are probe requests for SSIDs that are supported by the access point. If desired, you can configure access points to forward both acknowledged and unacknowledged probe requests to the controller. The controller can use the information from unacknowledged probe requests to improve the location accuracy.

How to Configure Probe Request Forwarding (CLI)

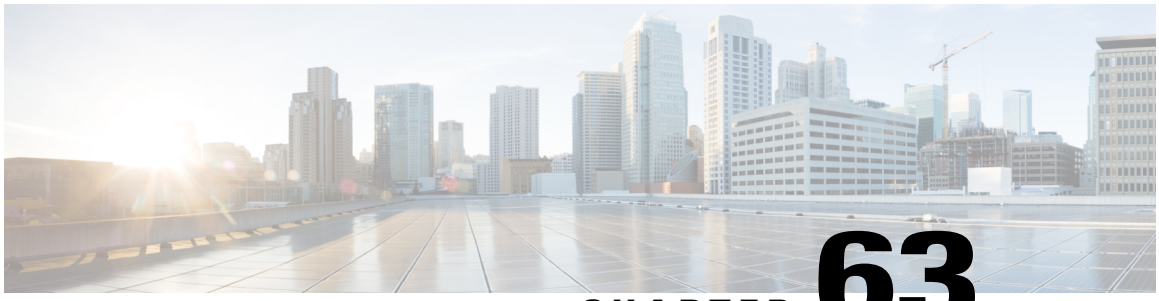
Note The procedure to perform this task using the controller GUI is not currently available.

SUMMARY STEPS

1. **configure terminal**
2. **wireless probe filter**
3. **wireless probe filter** *num_probes interval*
4. **end**
5. **show wireless probe**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**wireless probe filter** **Example:** `Controller(config)# wireless probe filter`	Enables or disables the filtering of probe requests forwarded from an access point to the controller. **Note** If you enable probe filtering, the default filter setting, the access point forwards only acknowledged probe requests to the controller. If you disable probe filtering, the access point forwards both acknowledged and unacknowledged probe requests to the controller.
Step 3	**wireless probe filter** *num_probes interval* **Example:** `Controller(config)# wireless probe filter 5 5`	Limits the number of probe requests sent to the controller per client per access point radio in a given interval. You must specify the following arguments with this command: • *num_probes*—Number of probe requests forwarded to the controller per client per access point radio in a given interval. The range is from 1 to 100. • *interval*—Probe limit interval in milliseconds. The range is from 100 to 10000.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.
Step 5	**show wireless probe** **Example:** `Controller# show wireless probe`	Displays the advanced probe request configuration.



CHAPTER 63

Optimizing RFID Tracking

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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.

Optimizing RFID Tracking on Access Points

To optimize the monitoring and location calculation of RFID tags, you can enable tracking optimization on up to four channels within the 2.4-GHz band of an 802.11b/g access point radio. This feature allows you to scan only the channels on which tags are usually programmed to operate (such as channels 1, 6, and 11).

How to Optimize RFID Tracking on Access Points

Optimizing RFID Tracking on Access Points (CLI)

SUMMARY STEPS

1. **ap name** *Cisco_AP* **mode monitor submode none**
2. **ap name** *Cisco_AP* **dot11 24ghz shutdown**
3. **ap name** *Cisco_AP* **monitor-mode tracking-opt**
4. **ap name** *Cisco_AP* **monitor-mode dot11b** {**fast-channel** [*first_channel second_channel third_channel fourth_channel*]}
5. **ap name** *Cisco_AP* **no dot11 24ghz shutdown**
6. **show ap monitor-mode summary**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**ap name** *Cisco_AP* **mode monitor submode none** **Example:** `Controller# ap name 3602a mode monitor submode none`	Specifies the monitor submode for the access point as none. **Note** A warning message indicates that changing the access point's mode will cause the access point to reboot and prompts you to specify whether you want to continue by entering **Y**. After you enter **Y**, the access point reboots.
Step 2	**ap name** *Cisco_AP* **dot11 24ghz shutdown** **Example:** `Controller# ap name AP01 dot11 24ghz shutdown`	Disables the access point radio.
Step 3	**ap name** *Cisco_AP* **monitor-mode tracking-opt** **Example:** `Controller# ap name TSIM_AP1 monitor-mode tracking-opt`	Configures the access point to scan only the Dynamic Channel Assignment (DCA) channels supported by its country of operation. **Note** To disable tracking optimization for an access point, enter the **ap name** *Cisco_AP* **monitor-mode tracking-opt no-optimization** command.
Step 4	**ap name** *Cisco_AP* **monitor-mode dot11b** {**fast-channel** [*first_channel second_channel third_channel fourth_channel*]} **Example:** `Controller# ap name AP01 monitor-mode dot11b fast-channel 1 2 3 4`	Chooses up to four specific 802.11b channels to be scanned by the access point. **Note** In the United States, you can assign any value from 1 to 11 (inclusive) to the channel variable. Other countries support additional channels. You must assign at least one channel.

	Command or Action	Purpose
Step 5	**ap name** *Cisco_AP* **no dot11 24ghz shutdown** **Example:** `Controller# ap name AP01 no dot11 24ghz shutdown`	Enables the access point radio.
Step 6	**show ap monitor-mode summary** **Example:** `Controller# show ap monitor-mode summary`	Displays all the access points in monitor mode.

Optimizing RFID Tracking on Access Points (GUI)

Step 1 Choose **Configuration>Wireless> AP Summary**.
The **All APs** page appears with the access points that are associated with the controller.

Step 2 Click the access point for which you want to configure monitoring mode.
The **AP > Edit** page that corresponds to the access point that you choose appears.

Step 3 From the **AP Status drop-down** list in the **General** area, choose **Monior**.

Step 4 Click **Apply** to commit your changes.
A warning message indicates that the access point will reboot. Click **Ok**.

Step 5 Choose **Configuration>Wireless>AP Summary>802.11 b/g/n Radios**.
The **802.11/g/n Radios** page appears with a list of access points.

Step 6 Select the checkbox that corresponds to the access point that you want to choose and click **Configure**.
The **802.11b/g/n Radios > Edit** page appears.

Step 7 From the **Admin Status** drop-down list in the **General** area, choose **Disable** to disable the access point radio.

Step 8 Click **Apply**.

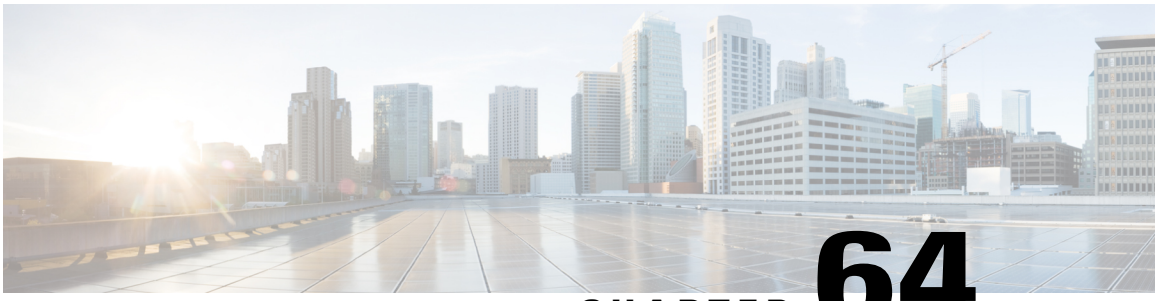
Configuration Examples for Optimizing RFID Tracking

Displaying all the Access Points in Monitor Mode: Example

This example shows how to display all the access points in monitor mode:

```
Controller# show ap monitor-mode summary

AP Name       Ethernet MAC   Status   Scanning
                                      Channel
                                      List
```

```
------------- -------------- -------- ---------
AP1131:4f2.9a 00:16:4:f2:9:a Tracking 1,6,NA,NA
```



CHAPTER 64

Configuring Country Codes

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Country Codes

- Generally, you configure one country code per controller; you configure one code that matches the physical location of the controller and its access points. You can configure up to 20 country codes per controller. This multiple-country support enables you to manage access points in various countries from a single controller.
- When the multiple-country feature is used, all controllers that are going to join the same RF group must be configured with the same set of countries, configured in the same order.
- Access points are capable of using all the available legal frequencies. However, access points are assigned to the frequencies that are supported in their relevant domains.
- The country list configured on the RF group leader determines which channels the members would operate on. This list is independent of which countries have been configured on the RF group members.
- For controllers in the Japan regulatory domain, you must have had one or more Japan country codes (JP, J2, or J3) configured on your controller at the time you last booted your controller.

- For controllers in the Japan regulatory domain, you must have at least one access point with a -J regulatory domain joined to your controller.

Information About Configuring Country Codes

Controllers and access points are designed for use in many countries with varying regulatory requirements. The radios within the access points are assigned to a specific regulatory domain at the factory (such as -E for Europe), but the country code enables you to specify a particular country of operation (such as FR for France or ES for Spain). Configuring a country code ensures that each radio's broadcast frequency bands, interfaces, channels, and transmit power levels are compliant with country-specific regulations.

Information About Migrating Access Points from the -J Regulatory Domain to the -U Regulatory Domain

The Japanese government has changed its 5-GHz radio spectrum regulations. These regulations allow a text box upgrade of 802.11a 5-GHz radios. Japan allows three frequency sets:

- J52 = 34 (5170 MHz), 38 (5190 MHz), 42 (5210 MHz), 46 (5230 MHz)
- W52 = 36 (5180 MHz), 40 (5200 MHz), 44 (5220 MHz), 48 (5240 MHz)
- W53 = 52 (5260 MHz), 56 (5280 MHz), 60 (5300 MHz), 64 (5320 MHz)

Cisco has organized these frequency sets into the following regulatory domains:

- -J regulatory domain = J52
- -P regulatory domain = W52 + W53
- -U regulatory domain = W52

Regulatory domains are used by Cisco to organize the legal frequencies of the world into logical groups. For example, most of the European countries are included in the -E regulatory domain. Cisco access points are configured for a specific regulatory domain at the factory and, with the exception of this migration process, never change. The regulatory domain is assigned per radio, so an access point's 802.11a and 802.11b/g radios can be assigned to different domains.

Note Controllers and access points might not operate properly if they are not designed for use in your country of operation. For example, an access point with part number AIR-CAP3602I-A-K9 (which is included in the Americas regulatory domain) cannot be used in Australia. Always be sure to purchase controllers and access points that match your country's regulatory domain.

The Japanese regulations allow the regulatory domain that is programmed into an access point's radio to be migrated from the -J domain to the -U domain. New access points for the Japanese market contain radios that are configured for the -P regulatory domain. -J radios are no longer being sold. In order to make sure that your existing -J radios work with the new -P radios in one network, you need to migrate your -J radios to the -U domain.

Country codes define the channels that can be used legally in each country. These country codes are available for Japan:

- JP—Allows only -J radios to join the controller
- J2—Allows only -P radios to join the controller
- J3—Uses the -U frequencies but allows both -U and -P radios to join the controller
- J4—Allows 2.4G PQU and 5G JPQU to join the controller

See the *Channels and Maximum Power Settings for Cisco Aironet Lightweight Access Points* document for the list of channels and power levels supported by access points in the Japanese regulatory domains.

Using the W56 Band in Japan

The Japanese government is formally permitting wireless LAN use of the frequencies in the W56 band for 802.11a radios. The W56 band includes the following channels, frequencies, and power levels (in dBm):

Channel	Frequency (MHz)	Maximum Power for AIR-LAP1132AG-Q-K9	Maximum Power for AIR-LAP1242AG-Q-K9
100	5500	17	15
104	5520	17	15
108	5540	17	15
112	5560	17	15
116	5580	17	15
120	5600	17	15
124	5620	17	15
128	5640	17	15
132	5660	17	15
136	5680	17	15
140	5700	17	15

To set up a network consisting of only -P and -Q access points, configure the country code to J2. To set up a network that consists of -P, -Q, and -U access points, configure the country code to J3.

Dynamic Frequency Selection

The Cisco converged access solution complies with regulations that require radio devices to use DFS to detect radar signals and avoid interfering with them.

When a lightweight access point with a 5-GHz radio operates on one of the 15 channels listed in the table below, the controller to which the access point is associated, automatically uses DFS to set the operating frequency.

When you manually select a channel for DFS-enabled 5-GHz radios, the controller checks for radar activity on the channel for 60 seconds. If there is no radar activity, the access point operates on the channel that you selected. If there is radar activity on the channel that you selected, the controller automatically selects a different channel, and after 30 minutes, the access point retries the channel.

Note After radar has been detected on a DFS-enabled channel, it cannot be used for 30 minutes.

Note The Rogue Location Detection Protocol (RLDP) and rogue containment are not supported on the channels listed in the table below.

Note The maximum legal transmit power is greater for some 5-GHz channels than for others. When the controller randomly selects a 5-GHz channel on which power is restricted, it automatically reduces the transmit power to comply with power limits for that channel.

Table 110: DFS-Enabled 5-GHz Channels

52 (5260 MHz)	104 (5520 MHz)	124 (5620 MHz)
56 (5280 MHz)	108 (5540 MHz)	128 (5640 MHz)
60 (5300 MHz)	112 (5560 MHz)	132 (5660 MHz)
64 (5320 MHz)	116 (5580 MHz)	136 (5680 MHz)
100 (5500 MHz)	120 (5600 MHz)	140 (5700 MHz)

Using DFS, the controller monitors operating frequencies for radar signals. If it detects radar signals on a channel, the controller takes these steps:

- It changes the access point channel to a channel that has not shown radar activity within the last 30 minutes. (The radar event is cleared after 30 minutes.) The controller selects the channel at random.
- If the channel selected is one of the channels in the UNII-2 or UNII-2e that is affected by DFS, it scans the new channel for radar signals for 60 seconds. If there are no radar signals on the new channel, the controller accepts client associations.
- It records the channel that showed radar activity as a radar channel and prevents activity on that channel for 30 minutes.
- It generates a trap to alert the network manager.

How to Configure Country Codes (CLI)

Note The procedure to perform this task using the controller GUI is not currently available.

SUMMARY STEPS

1. **enable**
2. **show wireless country supported**
3. **configure terminal**
4. **ap dot11 24ghz shutdown**
5. **ap dot11 5ghz shutdown**
6. **ap country** *country_code*
7. **end**
8. **show wireless country channels**
9. **configure terminal**
10. **no ap dot11 5ghz shutdown**
11. **no ap dot11 24ghz shutdown**
12. **end**
13. **ap name** *Cisco_AP* **shutdown**
14. **configure terminal**
15. **ap country** *country_code*
16. **end**
17. **ap name** *Cisco_AP* **no shutdown**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Controller# enable`	Enters privileged EXEC mode.
Step 2	**show wireless country supported** **Example:** `Controller# show wireless country supported`	Displays a list of all available country codes.
Step 3	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 4	**ap dot11 24ghz shutdown** **Example:** `Controller(config)# ap dot11 5ghz shutdown`	Disables the 802.11a network.
Step 5	**ap dot11 5ghz shutdown** **Example:** `Controller(config)# ap dot11 24ghz shutdown`	Disables the 802.11b/g network.
Step 6	**ap country** *country_code* **Example:** `Controller(config)# ap country IN`	Assigns access points to a specific country. **Note** Make sure that the country code you choose is compatible with the regulatory domain of at least one of the access point's radios.
Step 7	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.
Step 8	**show wireless country channels** **Example:** `Controller# show wireless country channels`	Displays the list of available channels for the country codes configured on your controller. **Note** Perform Steps 9 through 17 only if you have configured multiple country codes in Step 6.
Step 9	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 10	**no ap dot11 5ghz shutdown** **Example:** `Controller(config)# no ap dot11 5ghz shutdown`	Enables the 802.11a network.
Step 11	**no ap dot11 24ghz shutdown** **Example:** `Controller(config)# no ap dot11 24ghz shutdown`	Enables the 802.11b/g network.
Step 12	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.
Step 13	**ap name** *Cisco_AP* **shutdown** **Example:** `Controller# ap name AP02 shutdown`	Disables the access point. **Note** Ensure that you disable only the access point for which you are configuring country codes.

	Command or Action	Purpose
Step 14	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 15	**ap country** *country_code* **Example:** `Controller# ap country IN`	Assigns an access point to a specific country. **Note** Ensure that the country code that you choose is compatible with the regulatory domain of at least one of the access point's radios. **Note** If you enabled the networks and disabled some access points and then enter the **ap country** *country_code* command, the specified country code is configured on only the disabled access points. All other access points are ignored.
Step 16	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.
Step 17	**ap name** *Cisco_AP* **no shutdown** **Example:** `Controller# ap name AP02 no shutdown`	Enables the access point.

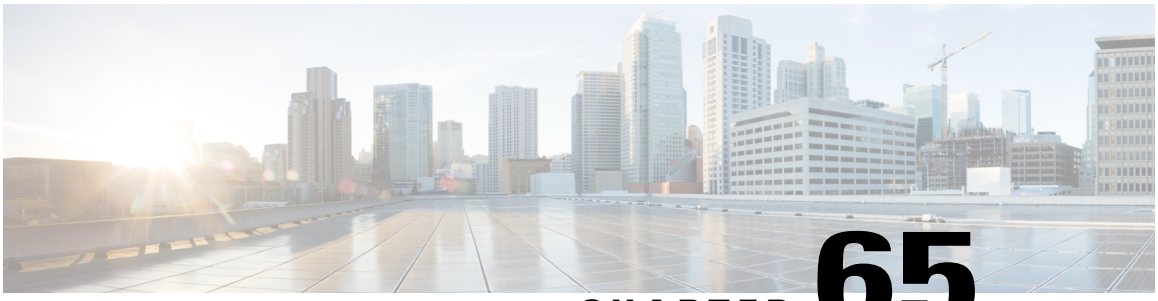
Configuration Examples for Configuring Country Codes

Displaying Channel List for Country Codes: Example

This example shows how to display the list of available channels for the country codes configured on your controller:

```
Controller# show wireless country channels

Configured Country.....................: US - United States
KEY: * = Channel is legal in this country and may be configured manually.
A = Channel is the Auto-RF default in this country.
. = Channel is not legal in this country.
C = Channel has been configured for use by Auto-RF.
x = Channel is available to be configured for use by Auto-RF.
(-,-) = (indoor, outdoor) regulatory domain allowed by this country.
-----------------:+-+-+-+-+-+-+-+-+-+-+-+-+-+-
802.11bg :
Channels : 1 1 1 1 1
: 1 2 3 4 5 6 7 8 9 0 1 2 3 4
-----------------:+-+-+-+-+-+-+-+-+-+-+-+-+-+-
(-A ,-AB ) US : A * * * * A * * * * A . . .
Auto-RF : . . . . . .
-----------------:+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
802.11a : 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
Channels : 3 3 3 4 4 4 4 4 5 5 6 6 0 0 0 1 1 2 2 2 3 3 4 4 5 5 6 6
: 4 6 8 0 2 4 6 8 2 6 0 4 0 4 8 2 6 0 4 8 2 6 0 9 3 7 1 5
```

```
-----------------:+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
(-A ,-AB ) US : . A . A . A . A A A A A * * * * * . . . * * * A A A A
*
Auto-RF : . . . . . .
-----------------:+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
4.9GHz 802.11a :
Channels : 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2
: 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
-----------------:+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
US (-A ,-AB ): * * * * * * * * * * * * * * * * * * * A * * * * * A
Auto-RF : . . . . . .
-----------------:+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
```



CHAPTER 65

Configuring Link Latency

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Link Latency

- The controller displays the current round-trip time as well as a running minimum and maximum round-trip time. The minimum and maximum times continue to run as long as the controller is up or can be cleared and allowed to restart.
- You can configure link latency for a specific access point using the controller GUI or CLI or for all access points joined to the controller using the CLI.

Restrictions for Configuring Link Latency

- Link latency calculates the Control and Provisioning of Wireless Access Points (CAPWAP) response time between the access point and the controller. It does not measure network latency or ping responses.

Information About Configuring Link Latency

You can configure link latency on the controller to measure the link between an access point and the controller. You can use this feature with all access points that are joined to the controller where the link can be a slow or unreliable WAN connection.

TCP MSS

If the client's maximum segment size (MSS) in a Transmission Control Protocol (TCP) three-way handshake is greater than the maximum transmission unit can handle, the client might experience reduced throughput and the fragmentation of packets. To avoid this problem, you can specify the MSS for all access points that are joined to the controller or for a specific access point.

When you enable this feature, the access point selects for TCP packets to and from wireless clients in its data path. If the MSS of these packets is greater than the value that you configured or greater than the default value for the CAPWAP tunnel, the access point changes the MSS to the new configured value.

Link Tests

A link test is used to determine the quality of the radio link between two devices. Two types of link-test packets are transmitted during a link test: request and response. Any radio receiving a link-test request packet fills in the appropriate text boxes and echoes the packet back to the sender with the response type set.

The radio link quality in the client-to-access point direction can differ from that in the access point-to-client direction due to the asymmetrical distribution of the transmit power and receive sensitivity on both sides. Two types of link tests can be performed: a ping test and a CCX link test.

With the *ping link test*, the controller can test link quality only in the client-to-access point direction. The RF parameters of the ping reply packets received by the access point are polled by the controller to determine the client-to-access point link quality.

With the *CCX link test*, the controller can also test the link quality in the access point-to-client direction. The controller issues link-test requests to the client, and the client records the RF parameters (received signal strength indicator [RSSI], signal-to-noise ratio [SNR], and so on) of the received request packet in the response packet. Both the link-test requestor and responder roles are implemented on the access point and controller. Not only can the access point or controller initiate a link test to a CCX v4 or v5 client, but a CCX v4 or v5 client can initiate a link test to the access point or controller.

The controller shows the link-quality metrics for CCX link tests in both directions (out— the access point to the client; in— the client to the access point):

- Signal strength in the form of RSSI (minimum, maximum, and average)
- Signal quality in the form of SNR (minimum, maximum, and average)
- Total number of packets that are retried

- Maximum retry count for a single packet
- Number of lost packets
- Data rate of a successfully transmitted packet

The controller shows this metric regardless of direction:

- Link test request/reply round-trip time (minimum, maximum, and average)

The controller software supports CCX versions 1 through 5. CCX support is enabled automatically for every WLAN on the controller and cannot be disabled. The controller stores the CCX version of the client in its client database and uses it to limit the features for this client. If a client does not support CCXv4 or v5, the controller performs a ping link test on the client. If a client supports CCXv4 or v5, the controller performs a CCX link test on the client. If a client times out during a CCX link test, the controller switches to the ping link test automatically.

How to Configure Link Latency

Configuring Link Latency (CLI)

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ap link-latency**
4. **ap tcp-adjust-mss size** *size*
5. **show ap name** *Cisco_AP* **config general**
6. **ap name** *Cisco_AP* **link-latency** [**reset**]
7. **show ap name** *Cisco_AP* **config general**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Controller# enable`	Enters privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**ap link-latency** **Example:** `Controller(config)# ap link-latency`	Enables link latency for all access points that are currently associated with the controller. **Note** To disable link latency for all the access points that are associated with the controller, use the **no ap link-latency** command. **Note** These commands enable or disable link latency only for access points that are currently joined to the controller. You have to enable or disable link latency for the access points that join in the future. **Note** To enable or disable link latency for specific access points that are associated with the controller, enter the following commands in Priveleged EXEC mode: • **ap name** *Cisco_AP* **link-latency—**Enables link latency. • **ap name** *Cisco_AP* **no link-latency—**Disables link latency.
Step 4	**ap tcp-adjust-mss size** *size* **Example:** `Controller(config)# ap tcp-adjust-mss size 537`	Configures TCP MSS adjust size for all access points. The range is from 536 to 1363.
Step 5	**show ap name** *Cisco_AP* **config general** **Example:** `Controller(config)# show ap name AP02 config general`	Displays the general configuration details of the access point. These configuration details contain the link latency results that correspond to the access point that you specify in the command. The output of this command contains the following link latency results: • Current Delay—The current round-trip time (in milliseconds) of CAPWAP heartbeat packets from the access point to the controller and back. • Maximum Delay—Since the time that link latency has been enabled or reset, the maximum round-trip time (in milliseconds) of CAPWAP heartbeat packets from the access point to the controller and back. • Minimum Delay—Since the time that link latency has been enabled or reset, the minimum round-trip time (in milliseconds) of CAPWAP heartbeat packets from the access point to the controller and back.
Step 6	**ap name** *Cisco_AP* **link-latency** [**reset**] **Example:** `Controller(config)# ap name AP02 link-latency reset`	Clears the current, minimum, and maximum link latency statistics on the controller for a specific access point.
Step 7	**show ap name** *Cisco_AP* **config general** **Example:** `Controller(config)# show ap name AP02 config general`	Displays the general configuration details of the access point. Use this command to see the result of the reset operation.

Configuring Link Latency (GUI)

Step 1 Choose **Configuration > Wireless > AP Summary**.
The **All APs** page appears with a list of access points.

Step 2 Click the access point for which you want to configure link latency.
The **AP > Edit** page appears.

Step 3 Click the **Advanced** tab.

Step 4 In the **Link Latency** area, select the **Enable Link Latency** check box.

Note You can select the **Enable Link Latency** check box to enable link latency for this access point or unselect it to prevent the access point from sending the round-trip time to the controller after every echo response is received. The default is unselected.

Step 5 Click **Apply** to commit your changes.

Step 6 When a message box appears that indicates that AP Parameters are modified successfully, click **Ok**.

Step 7 When the **All APs** page reappears, click the access point that you have modified earlier.
The **AP > Edit** page appears.

Step 8 Click the **Advanced** tab.
In the **Link Latency** area in the **AP > Edit** page, the following link latency and data latency results appear below the **Enable Link Latency** check box:

- **Current(mSec)—**The current round-trip time (in milliseconds) of CAPWAP heartbeat packets or data packets from the access point to the controller and back.
- **Minimum(mSec)—**Since the time that link latency has been enabled or reset, the minimum round-trip time (in milliseconds) of CAPWAP heartbeat packets or data packets from the access point to the controller and back.
- **Maximum(mSec)—**Since the time that link latency has been enabled or reset, the maximum round-trip time (in milliseconds) of CAPWAP heartbeat packets or data packets from the access point to the controller and back.

Step 9 Click **Reset Link Latency** to clear the current, minimum, and maximum link latency and data latency statistics on the controller for this access point.

Note After the page refreshes and the **All APs** page reappears, click the **Advanced** tab. The updated statistics appear in the **Minimum** and **Maximum** text boxes.

How to Configure TCP MSS

Configuring TCP MSS (CLI)

SUMMARY STEPS

1. **configure terminal**
2. **ap tcp-adjust-mss size** *size_value*
3. **reload**
4. **show ap tcp-adjust-mss**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**ap tcp-adjust-mss size** *size_value* **Example:** `Controller(config)# ap tcp-adjust-mss size 537`	Enables the TCP MSS on the particular access point that you specify. **Note** To enable TCP MSS on all the access points that are associated with the controller, enter the **ap tcp-adjust-mss size** *size_value* command, where the size parameter is from 536 to 1363 bytes. The default value varies for different clients.
Step 3	**reload** **Example:** `Controller# reload`	Reboots the controller in order for your change to take effect.
Step 4	**show ap tcp-adjust-mss** **Example:** `Controller# show ap tcp-adjust-mss`	Displays the current TCP MSS setting for all the access points that are associated with the controller. **Note** To display the TCP MSS settings that correspond to a specific access point, enter the **show ap name** *Cisco_AP* **tcp-adjust-mss** command.

Configuring TCP MSS (GUI)

Step 1 Choose **Monitor > AP Summary > Global AP Configuration**.

The **Global Configuration** page appears.

Step 2 In the **TCP MSS** area, select the **Global TCP Adjust MSS** check box and set the MSS for all access points that are associated with the controller. The valid range is from 536 to 1363 bytes.

Performing a Link Test (CLI)

Note The procedure to perform this task using the controller GUI is not currently available.

SUMMARY STEPS

1. **test wireless linktest** *mac_address*
2. **configure terminal**
3. **wireless linktest frame-size** *frame_size*
4. **wireless linktest number-of-frames** *number_of_frames*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**test wireless linktest** *mac_address* **Example:** `Controller# test wireless linktest 00:0d:88:c5:8a:d1`	Runs a link test.
Step 2	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 3	**wireless linktest frame-size** *frame_size* **Example:** `Controller(config)# wireless linktest frame-size 41`	Configures the link test frame size for each packet.
Step 4	**wireless linktest number-of-frames** *number_of_frames* **Example:** `Controller(config)# wireless linktest number-of-frames 50`	Configures the number of frames to send for the link test.

	Command or Action	Purpose
Step 5	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuration Examples for Configuring Link Latency

Running a Link Test: Example

This example shows how to run a link test:

```
Controller# test wireless linktest 00:0d:88:c5:8a:d1

When CCX v4 or later releases is enabled on both the controller and the client being tested,
 information similar to the following appears:
CCX Link Test to 00:0d:88:c5:8a:d1.
Link Test Packets Sent..................................... 20
Link Test Packets Received................................. 10
Link Test Packets Lost (Total/AP to Client/Client to AP).... 10/5/5
Link Test Packets round trip time (min/max/average)........ 5ms/20ms/15ms
RSSI at AP (min/max/average)............................... -60dBm/-50dBm/-55dBm
RSSI at Client (min/max/average)........................... -50dBm/-40dBm/-45dBm
SNR at AP (min/max/average)................................ 40dB/30dB/35dB
SNR at Client (min/max/average)............................ 40dB/30dB/35dB
Transmit Retries at AP (Total/Maximum)..................... 5/3
Transmit Retries at Client (Total/Maximum)................. 4/2
Transmit rate: 1M 2M 5.5M 6M 9M 11M 12M 18M 24M 36M 48M 54M 108M
Packet Count: 0 0 0 0 0 0 0 0 0 2 0 18 0
Transmit rate: 1M 2M 5.5M 6M 9M 11M 12M 18M 24M 36M 48M 54M 108M
Packet Count: 0 0 0 0 0 0 0 0 0 2 0 8 0

When CCX v4 or later releases is not enabled on either the controller or the client being
tested, fewer details appear:
Ping Link Test to 00:0d:88:c5:8a:d1.
Link Test Packets Sent..................... 20
Link Test Packets Received................. 20
Local Signal Strength...................... -49dBm
Local Signal to Noise Ratio................ 39dB
```

Displaying Link Latency Information: Example

This example shows how to display general configuration details of the access point. These configuration details contain the link latency results that correspond to the access point that you specify in the command.

```
Controller# show ap name AP01 config general

Cisco AP Name                                  : AP01
Cisco AP Identifier                            : 55
Country Code                                   : US  - United States
Regulatory Domain Allowed by Country           : 802.11bg:-A     802.11a:-A
AP Country Code                                : US  - United States
AP Regulatory Domain                           : Unconfigured
Switch Port Number                             : Te1/0/1
MAC Address                                    : 0000.2000.03f0
IP Address Configuration                       : Static IP assigned
IP Address                                     : 9.9.9.16
```

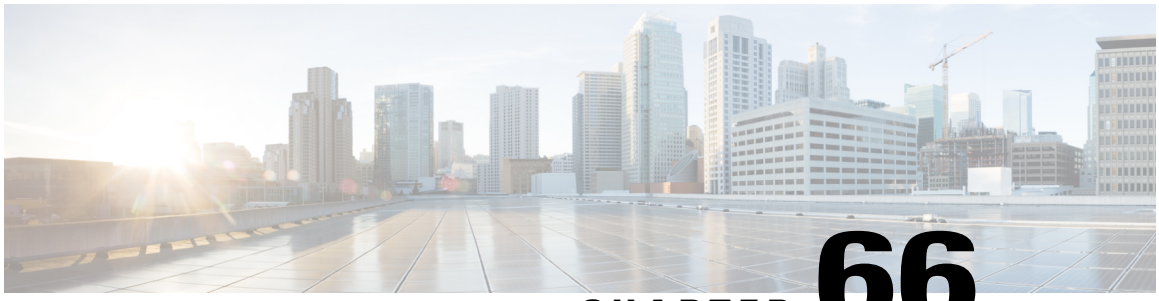
```
IP Netmask                                     : 255.255.0.0
Gateway IP Address                             : 9.9.9.2
Fallback IP Address Being Used                 : 9.9.9.16
Domain                                         : Cisco
Name Server                                    : 0.0.0.0
CAPWAP Path MTU                                : 1485
Telnet State                                   : Enabled
SSH State                                      : Disabled
Cisco AP Location                              : default-location
Cisco AP Group Name                            : default-group
Primary Cisco Controller Name                  : CAPWAP Controller
Primary Cisco Controller IP Address            : 9.9.9.2
Secondary Cisco Controller Name                :
Secondary Cisco Controller IP Address          : Not Configured
Tertiary Cisco Controller Name                 :
Tertiary Cisco Controller IP Address           : Not Configured
Administrative State                           : Enabled
Operation State                                : Registered
AP Mode                                        : Local
AP Submode                                     : Not Configured
Remote AP Debug                                : Disabled
Logging Trap Severity Level                    : informational
Software Version                               : 7.4.0.5
Boot Version                                   : 7.4.0.5
Stats Reporting Period                         : 180
LED State                                      : Enabled
PoE Pre-Standard Switch                        : Disabled
PoE Power Injector MAC Address                 : Disabled
Power Type/Mode                                : Power Injector/Normal Mode
Number of Slots                                : 2
AP Model                                       : 3502E
AP Image                                       : C3500-K9W8-M
IOS Version                                    :
Reset Button                                   :
AP Serial Number                               : SIM1140K002
AP Certificate Type                            : Manufacture Installed
Management Frame Protection Validation         : Disabled
AP User Mode                                   : Customized
AP User Name                                   : Not Configured
AP 802.1X User Mode                            : Not Configured
AP 802.1X User Name                            : Not Configured
Cisco AP System Logging Host                   : 255.255.255.255
AP Up Time                                     : 16 days 3 hours 14 minutes 1 s
econd
AP CAPWAP Up Time                              : 33 minutes 15 seconds
Join Date and Time                             : 01/02/2013 22:41:47
Join Taken Time                                : 16 days 2 hours 40 minutes 45
seconds
Join Priority                                  : 1
Ethernet Port Duplex                           : Auto
Ethernet Port Speed                            : Auto
AP Link Latency                                : Enabled
Current Delay                                  : 0
Maximum Delay                                  : 0
Minimum Delay                                  : 0
Last Updated (based on AP up time)             : 0 seconds
Rogue Detection                                : Disabled
AP TCP MSS Adjust                              : Disabled
AP TCP MSS Size                                : 536
```

Displaying TCP MSS Settings: Example

This example shows how to display the current TCP MSS setting for all the access points that are associated with the controller:

```
Controller# show ap tcp-adjust-mss

AP Name                     TCP State        MSS Size
-------------------------------------------------------
AP01                        Disabled         6146
```

```
AP02                    Disabled        536
AP03                    Disabled        6146
AP04                    Disabled        6146
AP05                    Disabled        6146
```



CHAPTER 66

Configuring Power over Ethernet

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Configuring Power over Ethernet

When an access point that has been converted to lightweight mode (such as an AP1262) access point is powered by a power injector that is connected to a Cisco pre-Intelligent Power Management (pre-IPM) switch, you must configure Power over Ethernet (PoE), which is also known as *inline power*.

How to Configure Power over Ethernet

Configuring Power over Ethernet (CLI)

SUMMARY STEPS

1. **ap name** *Cisco_AP* **power injector installed**
2. **ap name** *Cisco_AP* **power injector override**
3. **ap name** *Cisco_AP* **power injector switch-mac-address** *switch_mac_address*
4. **show ap name** *Cisco_AP* **config general**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**ap name** *Cisco_AP* **power injector installed** **Example:** `Controller# ap name AP02 power injector installed`	Enables the PoE power injector state. The access point remembers that a power injector is connected to this particular switch port. If you relocate the access point, you must reenter this command after the presence of a new power injector is verified. **Note** Enter this command if your network contains any older Cisco 6-W switches that could be accidentally overloaded if connected directly to a 12-W access point. Make sure that the Cisco Discovery Protocol (CDP) is enabled before entering this command. Otherwise, this command will fail.
Step 2	**ap name** *Cisco_AP* **power injector override** **Example:** `Controller# ap name AP02 power injector override`	Removes the safety checks and allows the access point to be connected to any switch port. You can use this command if your network does not contain any older Cisco 6-W switches that could be overloaded if connected directly to a 12-W access point. The access point assumes that a power injector is always connected. If you relocate the access point, it continues to assume that a power injector is present.
Step 3	**ap name** *Cisco_AP* **power injector switch-mac-address** *switch_mac_address* **Example:** `Controller# ap name AP02 power injector switch-mac-address 10a.2d.5c.3d`	Sets the MAC address of the switch port that has a power injector. **Note** Enter this command if you know the MAC address of the connected switch port and do not want to automatically detect it using the installed option.
Step 4	**show ap name** *Cisco_AP* **config general** **Example:** `Controller# show ap name AP02 config general`	Displays common information that includes the PoE settings for a specific access point. **Note** The Power Type/Mode text box shows “degraded mode” if the access point is not operating at full power.

Configuring Power over Ethernet (GUI)

Step 1 Choose **Configuration > Wireless > AP Summary**.
The **All APs** page appears with a list of access points that are joined to the controller.

Step 2 Click the access point that you want to choose.
The **AP > Edit** page appears.

Step 3 Click the **Advanced** tab.

Step 4 In the **Power Over Ethernet Settings** area, select the **Pre-Standard 802.3af Switches** check box.
Select this check box if the access point is being powered by a high-power 802.3af Cisco switch. This switch provides more than the traditional 6 Watts of power but does not support the intelligent power management (IPM) feature.

Note Unselect the **Pre-standard 802.3af switches** check box if power is being provided by a power injector. This is the default value.

Step 5 Select the **Power Injector State** check box.
Select this check box if the attached switch does not support IPM and a power injector is being used. If the attached switch supports IPM, you do not need to select this check box.

The **Power Injector Selection** drop-down list appears. This drop-down list contains parameters that enable you to protect your switch port from an accidental overload if the power injector is inadvertently bypassed.

Step 6 From the **Power Injector Selection** drop-down list, choose an option to specify the desired level of protection.
You can choose any one of the following three options:

- **Disables—**Disables power injector selection. This is the default option.
- **Installed—**Examines and remembers the MAC address of the currently connected switch port and assumes that a power injector is connected. Choose this option if your network contains older Cisco 6-Watt switches and you want to avoid possible overloads by forcing a double-check of any relocated access points.

  If you want to configure the switch MAC address, enter the MAC address in the Injector Switch MAC Address text box. If you want the access point to find the switch MAC address, leave the Injector Switch MAC Address text box blank.

  Note Each time that an access point is relocated, the MAC address of the new switch port fails to match the remembered MAC address, and the access point remains in low-power mode. You must then physically verify the existence of a power injector and reselect this option to cause the new MAC address to be remembered.

- **Override—**Allows the access point to operate in high-power mode without first verifying a matching MAC address. You can use this option if your network does not contain any older Cisco 6-W switches that could be overloaded if connected directly to a 12-W access point. The advantage of this option is that if you relocate the access point, it continues to operate in high-power mode without any further configuration. The disadvantage of this option is that if the access point is connected directly to a 6-W switch, an overload occurs.

Step 7 Click **Apply** to commit your changes.

What to Do Next

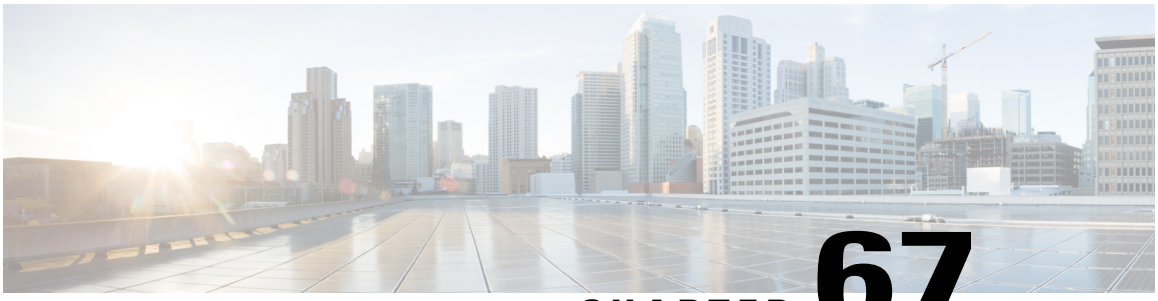
Manually reset the access point in order for the change to take effect.

Configuration Examples for Configuring Power over Ethernet

Displaying Power over Ethernet Information: Example

This example shows how to display common information that includes the PoE settings for a specific access point:

```
Controller# show ap name AP01 config general

Cisco AP Identifier.............. 1
Cisco AP Name.................... AP1
...
PoE Pre-Standard Switch.......... Enabled
PoE Power Injector MAC Addr...... Disabled
Power Type/Mode.................. PoE/Low Power (degraded mode)
...
```



CHAPTER 67

Configuring LED States for Access Points

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 LED States for Access Points

- At least one lightweight access points must be associated to the controller.

Restrictions for Configuring LED States for Access Points

- The LED state configuration at the global level takes precedence over the AP level.

Information About Configuring LED States for Access Points

In a wireless LAN network where there are a large number of access points, it is difficult to locate a specific access point associated with the controller. You can configure the controller to set the LED state of an access point so that it blinks and the access point can be located. This configuration can be done in the wireless network on a global as well as per-AP level.

How to Configure LED State of an Access Point in a Network Globally

Configuring the LED State of an Access Point in a Network Globally (CLI)

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ap led**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Controller# enable`	Enters privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 3	**ap led** **Example:** `Controller# ap led`	Sets the LED state of all access points associated to a controller.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring the LED State of Access Points in a Network Globally (GUI)

Step 1 Choose **Monitor > AP Summary > Global AP Configuration**.
The **Global Configuration** page appears.

Step 2 From the **LED State** drop-down list in the General area, choose **Enabled**.

Configuring the LED State on an Access Point

Note The procedure to perform this task using the controller GUI is not currently available.

SUMMARY STEPS

1. **enable**
2. **show ap name** *Cisco_AP* **config general |include led**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Controller# enable`	Enters privileged EXEC mode.
Step 2	**show ap name** *Cisco_AP* **config general |include led** **Example:** `Controller# show ap name 2602a config general	include led`	Displays the LED state for a specific access point.

Configuration Examples for Configuring LED States for Access Points

Displaying an Access Point Summary: Example

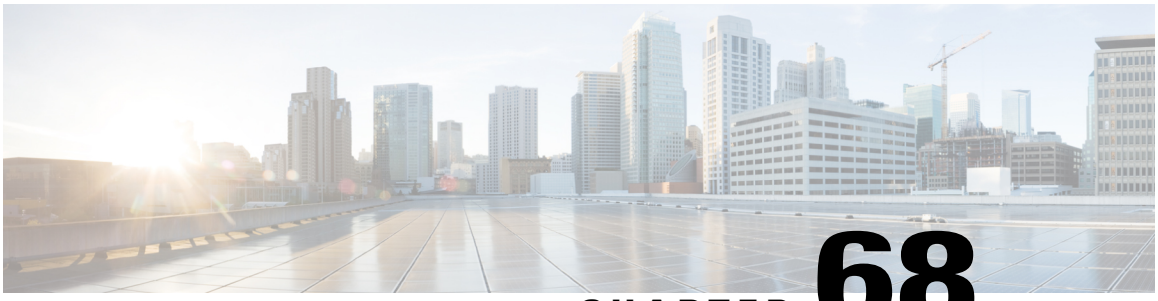
This example shows how to display a summary of all the access points that are associated with the controller:

```
Controller# show ap summary

AP Name AP Model Ethernet MAC   Radio MAC      Status
```

```
------- -------- -------------- -------------- ----------
AP01    1240AG   0000.2000.03f0 0000.2000.0030 Registered
AP02    1142N    6400.f1c5.e04a 1caa.0723.1ca0 Registered
```

PART XII

CleanAir



CHAPTER 68

Configuring Cisco CleanAir

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 CleanAir

Only Cisco CleanAir-enabled access points using the following access point modes can perform Cisco CleanAir spectrum monitoring:

- Local—In this mode, each Cisco CleanAir-enabled access point radio provides air quality and interference detection reports for the current operating channel only.
- FlexConnect—When a FlexConnect access point is connected to the controller, its Cisco CleanAir functionality is identical to local mode.
- Monitor—When Cisco CleanAir is enabled in monitor mode, the access point provides air quality and interference detection reports for all monitored channels.

  The following options are available:

- All— All channels
- DCA—Channel selection governed by the DCA list
- Country—All channel legal within a regulatory domain

Note Suppose you have two APs, one in the FlexConnect mode and the other in the monitor mode. Also suppose that you have created a profile enabling EAP attack against 802.1x auth. The Airmagnet (AM) tool, which can generate different types of attacks, fails to generate any attack even if you have provided valid AP MAC and STA MAC addresses. But if the AP MAC and STA MAC addresses in the AM tool are swapped, that is, the AP MAC address is specified in the STA MAC field and the STA MAC address is specified in the AP MAC field, then the tool is able to generate attacks, which the AP in the Monitor mode is also able to detect.

Note The access point does not participate in AQ HeatMap in Prime Infrastructure.

- SE-Connect—This mode enables a user to connect a Spectrum Expert application running on an external Microsoft Windows XP or Vista PC to a Cisco CleanAir-enabled access point in order to display and analyze detailed spectrum data. The Spectrum Expert application connects directly to the access point, bypassing the controller. An access point in SE-Connect mode does not provide any Wi-Fi, RF, or spectrum data to the controller. In addition to performing spectrum intelligence, an access point can provide other.

Restrictions for CleanAir

Only Cisco CleanAir-enabled access points using the following access point modes can perform Cisco CleanAir spectrum monitoring:

- Local—In this mode, each Cisco CleanAir-enabled access point radio provides air quality and interference detection reports for the current operating channel only.
- FlexConnect—When a FlexConnect access point is connected to the controller, its Cisco CleanAir functionality is identical to local mode.
- Monitor—When Cisco CleanAir is enabled in monitor mode, the access point provides air quality and interference detection reports for all monitored channels.

  The following options are available:

  - All— All channels
  - DCA—Channel selection governed by the DCA list
  - Country—All channel legal within a regulatory domain

Note Suppose you have two APs, one in the FlexConnect mode and the other in the monitor mode. Also suppose that you have created a profile enabling EAP attack against 802.1x auth. The Airmagnet (AM) tool, which can generate different types of attacks, fails to generate any attack even if you have provided valid AP MAC and STA MAC addresses. But if the AP MAC and STA MAC addresses in the AM tool are swapped, that is, the AP MAC address is specified in the STA MAC field and the STA MAC address is specified in the AP MAC field, then the tool is able to generate attacks, which the AP in the Monitor mode is also able to detect.

Note The access point does not participate in AQ HeatMap in Prime Infrastructure.

- SE-Connect—This mode enables a user to connect a Spectrum Expert application running on an external Microsoft Windows XP or Vista PC to a Cisco CleanAir-enabled access point in order to display and analyze detailed spectrum data. The Spectrum Expert application connects directly to the access point, bypassing the controller. An access point in SE-Connect mode does not provide any Wi-Fi, RF, or spectrum data to the controller. In addition to performing spectrum intelligence, an access point can provide other.

Information About CleanAir

Cisco CleanAir is a spectrum intelligence solution designed to proactively manage the challenges of a shared wireless spectrum. All of the users of the shared spectrum can be seen (both native devices and foreign interferers). It also enables the network to act upon this information. For example, the interfering device can be manually removed or the system can automatically change the channel away from the interference.

A Cisco CleanAir system consists of CleanAir-enabled access points, wireless controller modules, mobility controllers, mobility anchors and next generation switches. The access points join the mobility controller directly or through the mobility anchor. They collect information about all devices that operate in the industrial, scientific, and medical (ISM) bands, identify and evaluate the information as a potential interference source, and forward it to the controller. The controller controls the access points, collects spectrum data, and forwards information to WCS or a Cisco mobility services engine (MSE) upon request.

Any networking configurations can be performed only on the mobility controller, configurations cannot be performed in the MA mode. However, any radio level CleanAir configurations can be done using mobility anchor.

For every device operating in the unlicensed band, Cisco CleanAir tells what it is, where it is, how it is impacting the wireless network, and what actions should be taken. It simplifies RF.

Wireless LAN systems operate in unlicensed 2.4-GHz and 5-GHz ISM bands. Many devices like microwave ovens, cordless phones, and Bluetooth devices also operate in these bands and can negatively affect the Wi-Fi operations.

Some of the most advanced WLAN services, such as voice over wireless and IEEE 802.11n radio communications, could be significantly impaired by the interference caused by other legal users of the ISM bands. The integration of Cisco CleanAir functionality addresses this problem of radio frequency (RF) interference.

Role of the Controller in a Cisco CleanAir System

The controller performs the following tasks in a Cisco CleanAir system:

- Configures Cisco CleanAir capabilities on the access point.
- Provides interfaces (GUI, CLI, and SNMP) for configuring Cisco CleanAir features and retrieving data.
- Displays spectrum data.
- Collects and processes air quality reports from the access point and stores them in the air quality database. Air Quality Report (AQR) contains information about the total interference from all identified sources represented by Air Quality Index (AQI) and summary for the most severe interference categories. The CleanAir system can also include unclassified interference information under per interference type reports which enable you to take action in cases where the interference due to unclassified interfering devices is frequent.
- Collects and processes interference device reports (IDRs) from the access point and stores them in the interference device database.
- Forwards spectrum data to WCS and the MSE.

Interference Types that Cisco CleanAir can Detect

Cisco CleanAir can detect interference, report on the location and severity of the interference, and recommend different mitigation strategies. Two such mitigation strategies are persistent device avoidance and spectrum event-driven RRM. New

Wi-Fi chip-based RF management systems share these characteristics:

- Any RF energy that cannot be identified as a Wi-Fi signal is reported as noise.
- Noise measurements that are used to assign a channel plan tend to be averaged over a period of time to avoid instability or rapid changes that can be disruptive to certain client devices.
- Averaging measurements reduces the resolution of the measurement. As such, a signal that disrupts clients might not look like it needs to be mitigated after averaging.
- All RF management systems available today are reactive in nature.

Cisco CleanAir is different and can positively identify not only the source of the noise but also its location and potential impact to a WLAN. Having this information allows you to consider the noise within the context of the network and make intelligent and, where possible, proactive decisions. For CleanAir, two types of interference events are common:

- Persistent interference
- Spontaneous interference

Persistent interference events are created by devices that are stationary in nature and have intermittent but largely repeatable patterns of interference. For example, consider the case of a microwave oven located in a break room. Such a device might be active for only 1 or 2 minutes at a time. When operating, however, it can be disruptive to the performance of the wireless network and associated clients. Using Cisco CleanAir, you can positively identify the device as a microwave oven rather than indiscriminate noise. You can also determine exactly which part of the band is affected by the device, and because you can locate it, you can understand

which access points are most severely affected. You can then use this information to direct RRM in selecting a channel plan that avoids this source of interference for the access points within its range. Because this interference is not active for a large portion of the day, existing RF management applications might attempt to again change the channels of the affected access points. Persistent device avoidance is unique, however, in that it remains in effect as long as the source of interference is periodically detected to refresh the persistent status. The Cisco CleanAir system knows that the microwave oven exists and includes it in all future planning. If you move either the microwave oven or the surrounding access points, the algorithm updates RRM automatically.

Note Spectrum event-driven RRM can be triggered only by Cisco CleanAir-enabled access points in local mode.

Spontaneous interference is interference that appears suddenly on a network, perhaps jamming a channel or a range of channels completely. The Cisco CleanAir spectrum event-driven RRM feature allows you to set a threshold for air quality (AQ) that, if exceeded, triggers an immediate channel change for the affected access point. Most RF management systems can avoid interference, but this information takes time to propagate through the system. Cisco CleanAir relies on AQ measurements to continuously evaluate the spectrum and can trigger a move within 30 seconds. For example, if an access point detects interference from a video camera, it can recover by changing channels within 30 seconds of the camera becoming active. Cisco CleanAir also identifies and locates the source of interference so that more permanent mitigation of the device can be performed at a later time.

In the case of Bluetooth devices, Cisco CleanAir-enabled access points can detect and report interference only if the devices are actively transmitting. Bluetooth devices have extensive power save modes. For example, interference can be detected when data or voice is being streamed between the connected devices.

Interference Device Merging

The Interference Devices (ID) messages are processed on a Mobility Controller (MC). The Mobility Anchor (MA) forwards the ID messages from APs and hence they are processed on the MC. The MC has visibility of the neighbor information across APs connected to different MAs.

ID merging logic requires AP neighbor information. Neighbor information is obtained from the RRM module. This api only gives neighbor information to the APs directly connected to MC. In order to get neighbor information for all APs (connected to MC and connected MA) merging logic needs to use rrm api : XYZ.

Currently the AP neighbor list on MA is synced to MC once every 3 minutes; hence the AP neighbor list obtained by this api could be at most 3 mins old. This delay results in delay in merging of Devices as they are discovered. The subsequent periodic merge will pick up the updated neighbor information and merge is performed

Persistent Devices

Some interference devices such as outdoor bridges and Microwave Ovens only transmit when needed. These devices can cause significant interference to the local WLAN due to short duration and periodic operation remain largely undetected by normal RF management metrics. With CleanAir the RRM DCA algorithm can detect, measure, register and remember the impact and adjust the DCA algorithm. This minimizes the use of channels affected by the persistent devices in the channel plan local to the interference source. Cisco CleanAir detects and stores the persistent device information in the controller and this information is used to mitigate interfering channels.

Persistent Devices Detection

CleanAir-capable Monitor Mode access point collects information about persistent devices on all configured channels and store the information in controller. Local/Bridge mode AP detects interference devices on the serving channels only.

Persistent Device Avoidance

When a Persistent Device (PD) is detected in the CleanAir module, it is reported to the RRM module on the MA. This information is used in the channel selection by the subsequent EDRRM signal sent to the RRM module.

EDRRM and AQR Update Mode

AQ reports are only available on the MC. The mode configuration and timers are held in RCB on MA (for APs connected to MA). There is no change to the current API available for EMS/NMS. No change is required for directly connected APs as RCB (spectrum config and timers) is available locally. For remote APs (APs connected to MA), three new control messages are added. These three messages are for enable, restart timer and disable rapid update mode for a given AP MAC address and slot.

CleanAir High Availability

CleanAir configuration (network and radio) is stateful during the switchover. On the MC, EICORE provides the sync on network configurations across active and standby nodes. The radio configurations are synced using the HA Infrastructre. The CleanAir configurations on MA are pulled from the MC upon joining. The network configuration is not stored in the EICORE on MA, hence it is synced using HA Infrastructure.

CleanAir Data (AQ and IDR) reports are not stateful, that is, the standby and active nodes are not synced. On switchover, the APs send the reports to the current active slot. The RRM Client (HA Infra Client) is used for CleanAir HA sync.

How to Configure CleanAir

Enabling CleanAir for 2.4-GHz Band

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 24ghz cleanair**
3. **end**

DETAILED STEPS

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

Step 2	**ap dot11 24ghz cleanair**
	
	Example:
	`Controller(config)#ap dot11 24ghz cleanair`
	`Controller(config)#no ap dot11 24ghz cleanair`
		Enables the CleanAir feature on 802.11b network. Add **No** in the command to disable CleanAir on the 802.11b network.

Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring a CleanAir Alarm for 2.4-GHz Air-Quality and Devices

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 24ghz cleanair alarm air-quality threshold** *threshold_value*
3. **ap dot11 24ghz cleanair alarm device** {**bt-discovery** | **bt-link** | **canopy** | **cont-tx** | **dect-like** | **fh** | **inv** | **jammer** | **mw-oven** | **nonstd** | **report** | **superag** | **tdd-tx** | **video** | **wimax-fixed** | **wimax-mobile** | **xbox** | **zigbee** }
4. **end**

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	**ap dot11 24ghz cleanair alarm air-quality threshold** *threshold_value* **Example:** `Controller(config)#ap dot11 24ghz cleanair alarm air-quality threshold 50`	Configures the alarm for the threshold value for air-quality for all the 2.4-GHz devices. Add the **No** form of this command to disable the alarm.
Step 3	**ap dot11 24ghz cleanair alarm device** {**bt-discovery** | **bt-link** | **canopy** | **cont-tx** | **dect-like** | **fh** | **inv** | **jammer** | **mw-oven** | **nonstd** | **report** | **superag** | **tdd-tx** | **video** | **wimax-fixed** | **wimax-mobile** | **xbox** | **zigbee** } **Example:** `Controller(config)#ap dot11 24ghz cleanair alarm device canopy`	Configures the alarm for the 2.4-GHz devices. Add the **No** form command to disable the alarm. • **bt-discovery–** Bluetooth Discovery. • **bt-link–** Bluetooth Link. • **canopy–** Canopy devices. • **cont-tx–** Continuous Transmitter. • **dect-like–** Digital Enhanced Cordless Communication (DECT)-like phone. • **fh–** 802.11 frequency hopping devices. • **inv–** Devices using spectrally inverted WiFi signals. • **jammer–** Jammer. • **mw-oven–** Microwave oven. • **nonstd–** Devices using non standard Wi-Fi channels. • **report–** Reports. • **superag–** 802.11 SuperAG devices. • **tdd-tx–** TDD Transmitter. • **video–** Video cameras. • **wimax-fixed–** WiMax Fixed. • **wimax-mobile–** WiMax Mobile. • **xbox–** Xbox. • **zigbee–** 802.15.4 devices.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring Interference Reporting for 2.4-GHz devices

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 24ghz cleanair device**{**bt-discovery | bt-link | canopy | cont-tx | dect-like | fh | inv | jammer | mw-oven | nonstd | report | superag | tdd-tx | video | wimax-fixed | wimax-mobile | xbox | zigbee** }
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 24ghz cleanair device**{**bt-discovery	bt-link	canopy	cont-tx	dect-like	fh	inv	jammer	mw-oven	nonstd	report	superag	tdd-tx	video	wimax-fixed	wimax-mobile	xbox	zigbee** } **Example:** `Controllerconfig#ap dot11 24ghz cleanair device bt-discovery` `Controllerconfig#ap dot11 24ghz cleanair device bt-link` `Controllerconfig#ap dot11 24ghz cleanair device canopy` `Controllerconfig#ap dot11 24ghz cleanair device cont-tx` `Controllerconfig#ap dot11 24ghz cleanair device dect-like` `Controllerconfig#ap dot11 24ghz cleanair device fh` `Controllerconfig#ap dot11 24ghz cleanair device inv` `Controllerconfig#ap dot11 24ghz cleanair device jammer` `Controllerconfig#ap dot11 24ghz cleanair device mw-oven` `Controllerconfig#ap dot11 24ghz cleanair device nonstd` `Controllerconfig#ap dot11 24ghz cleanair device report` `Controllerconfig#ap dot11 24ghz cleanair device superag` `Controllerconfig#ap dot11 24ghz cleanair device tdd-tx` `Controllerconfig#ap dot11 24ghz cleanair device video` `Controllerconfig#ap dot11 24ghz cleanair device wimax-fixed`	Configures the 2.4 GHz interference devices to report to the controller. Add **No** in the command to disable. • **bt-discovery**- Bluetooth Discovery. • **bt-link**- Bluetooth Link. • **canopy**- Canopy devices. • **cont-tx**- Continuous Transmitter. • **dect-like**- Digital Enhanced Cordless Communication (DECT) like phone. • **fh**- 802.11 frequency hopping devices. • **inv**- Devices using spectrally inverted WiFi signals. • **jammer**- Jammer. • **mw-oven**- Microwave Oven. • **nonstd**- Devices using non-standard WiFi channels. • **report**- no description • **superag**- 802.11 SuperAG devices. • **tdd-tx**- TDD Transmitter. • **video**- Video cameras. • **wimax-fixed**- WiMax Fixed. • **wimax-mobile**- WiMax Mobile. • **xbox**- Xbox. • **zigbee**- 802.15.4 devices.

	Command or Action	Purpose
	Controllerconfig#**ap dot11 24ghz cleanair device wimax-mobile** Controllerconfig#**ap dot11 24ghz cleanair device xbox** Controllerconfig#**ap dot11 24ghz cleanair device zigbee**	
Step 3	**end** **Example:** Controller(config)# **end**	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Enabling CleanAir for 5-GHz Band

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 5ghz cleanair**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Controller# **configure terminal**	Enters global configuration mode.
Step 2	**ap dot11 5ghz cleanair** **Example:** Controller(config)#ap dot11 5ghz cleanair Controller(config)#no ap dot11 5ghz cleanair	Enables the CleanAir feature on 802.11a network. Add **No** in the command to disable CleanAir on the 802.11a network.
Step 3	**end** **Example:** Controller(config)# **end**	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring a CleanAir Alarm for 5-GHz Air-Quality and Devices

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 5ghz cleanair alarm air-quality threshold** *threshold_value*
3. **ap dot11 5ghz cleanair alarm device**{**canopy | cont-tx | dect-like | inv | jammer | nonstd | radar | report | superag | tdd-tx | video | wimax-fixed | wimax-mobile**}
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 5ghz cleanair alarm air-quality threshold** *threshold_value* **Example:** `Controller(config)#ap dot11 5ghz cleanair alarm air-quality threshold 50`	Configures the alarm for the threshold value for air-quality for all the 5-GHz devices. Add the **No** form of the command to disable the alarm.
Step 3	**ap dot11 5ghz cleanair alarm device**{**canopy	cont-tx	dect-like	inv	jammer	nonstd	radar	report	superag	tdd-tx	video	wimax-fixed	wimax-mobile**} **Example:** `Controller(config)#ap dot11 5ghz cleanair alarm device`	Configures the alarm for the 5-GHz devices. Add the **No** form of the command to disable the alarm. • **canopy**- Canopy devices. • **cont-tx**- Continuous Transmitter. • **dect-like**- Digital Enhanced Cordless Communication (DECT) like phone. • **fh**- 802.11 frequency hopping devices. • **inv**- Devices using spectrally inverted WiFi signals. • **jammer**- Jammer. • **nonstd**- Devices using non-standard WiFi channels. • **radar**- Radars. • **report**- no description • **superag**- 802.11 SuperAG devices. • **tdd-tx**- TDD Transmitter. • **video**- Video cameras. • **wimax-fixed**- WiMax Fixed.

	Command or Action	Purpose
		• **wimax-mobile**- WiMax Mobile.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring Interference Reporting for 5-GHz devices

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 5ghz cleanair device**{**canopy | cont-tx | dect-like | inv | jammer | nonstd | radar | report | superag | tdd-tx | video | wimax-fixed | wimax-mobile**}
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 5ghz cleanair device**{**canopy	cont-tx	dect-like	inv	jammer	nonstd	radar	report	superag	tdd-tx	video	wimax-fixed	wimax-mobile**} **Example:** `Controllerconfig#ap dot11 5ghz cleanair device canopy` `Controllerconfig#ap dot11 5ghz cleanair device cont-tx` `Controllerconfig#ap dot11 5ghz cleanair device dect-like` `Controllerconfig#ap dot11 5ghz cleanair device inv` `Controllerconfig#ap dot11 5ghz cleanair device jammer` `Controllerconfig#ap dot11 5ghz cleanair device nonstd` `Controllerconfig#ap dot11 5ghz cleanair device radar` `Controllerconfig#ap dot11 5ghz cleanair device report`	Configures the 5-GHz interference devices to report to the controller. Add the **No** form of the command to disable interference device reporting. • **canopy**- Canopy devices. • **cont-tx**- Continuous Transmitter. • **dect-like**- Digital Enhanced Cordless Communication (DECT) like phone. • **fh**- 802.11 frequency hopping devices. • **inv**- Devices using spectrally inverted WiFi signals. • **jammer**- Jammer. • **nonstd**- Devices using non-standard WiFi channels. • **radar**- Radars. • **report**- no description • **superag**- 802.11 SuperAG devices.

	Command or Action	Purpose
	Controllerconfig#**ap dot11 5ghz cleanair device superag** Controllerconfig#**ap dot11 5ghz cleanair device tdd-tx** Controllerconfig#**ap dot11 5ghz cleanair device video** Controllerconfig#**ap dot11 5ghz cleanair device wimax-fixed** Controllerconfig#**ap dot11 5ghz cleanair device wimax-mobile**	• **tdd-tx**- TDD Transmitter. • **video**- Video cameras. • **wimax-fixed**- WiMax Fixed. • **wimax-mobile**- WiMax Mobile.
Step 3	**end** **Example:** Controller(config)# **end**	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring EDRRM for CleanAir-Events

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 24ghz | 5ghz rrm channel cleanair-event**
3. **ap dot11 24ghz | 5ghz rrm channel cleanair-event** [**sensitivity** {**high | low | medium**}]
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** Controller# **configure terminal**	Enters global configuration mode.
Step 2	**ap dot11 24ghz	5ghz rrm channel cleanair-event** **Example:** Controller(config)#**ap dot11 24ghz rrm channel cleanair-event** Controller(config)#**no ap dot11 24ghz rrm channel cleanair-event**	Enables EDRRM cleanair-event. Add the **No** form of the command to disable EDRRM.
Step 3	**ap dot11 24ghz	5ghz rrm channel cleanair-event** [**sensitivity** {**high	low	medium**}]	Configures the EDRRM sensitivity of cleanair-event. • **High–** Specifies the most sensitivity to non Wi–Fi interference as indicated by the air quality (AQ) value.

	Command or Action	Purpose
	Example: `Controller(config)#ap dot11 24ghz rrm channel cleanair-event sensitivity high`	• **Low–** Specifies the least sensitivity to non Wi–Fi interference as indicated by the AQ value. • **Medium–** Specifies medium sensitivity to non Wi–Fi interference as indicated by the AQ value.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring Persistent Device Avoidance

SUMMARY STEPS

1. **configure terminal**
2. **ap dot11 24ghz | 5ghz rrm channel device**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 24ghz | 5ghz rrm channel device** **Example:** `Controller(config)#ap dot11 24ghz rrm channel device`	Enables the persistent non Wi–Fi device avoidance in the 802.11 channel assignment. Add the **No** form of the command to disable the persistent device avoidance.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Monitoring Various CleanAir Parameters

Table 111: Commands for monitoring CleanAir

Commands	Description
show ap dot11 24ghz cleanair air-quality summary	Displays CleanAir Air Quality (AQ) data for 2.4-GHz band
show ap dot11 24ghz cleanair air-quality worst	Displays CleanAir Air Quality (AQ) worst data for 2.4-GHz band
show ap dot11 24ghz cleanair config	Displays CleanAir Configuration for 2.4-GHz band
show ap dot11 24ghz cleanair device type all	Displays all CleanAir Interferers for 2.4-GHz band
show ap dot11 24ghz cleanair device type bt-discovery	Displays CleanAir Interferers of type BT Discovery for 2.4-GHz band
show ap dot11 24ghz cleanair device type bt-link	Displays CleanAir Interferers of type BT Link for 2.4-GHz band
show ap dot11 24ghz cleanair device type canopy	Displays CleanAir Interferers of type Canopy for 2.4-GHz band
show ap dot11 24ghz cleanair device type cont-tx	Displays CleanAir Interferers of type Continuous TX for 2.4-GHz band
show ap dot11 24ghz cleanair device type dect-like	Displays CleanAir Interferers of type DECT Like for 2.4-GHz band
show ap dot11 24ghz cleanair device type fh	Displays CleanAir Interferers of type 802.11FH for 2.4-GHz band
show ap dot11 24ghz cleanair device type inv	Displays CleanAir Interferers of type WiFi Inverted for 2.4-GHz band
show ap dot11 24ghz cleanair device type jammer	Displays CleanAir Interferers of type Jammer for 2.4-GHz band
show ap dot11 24ghz cleanair device type mw-oven	Displays CleanAir Interferers of type MW Oven for 2.4-GHz band
show ap dot11 24ghz cleanair device type nonstd	Displays CleanAir Interferers of type WiFi Inv. Ch for 2.4-GHz band
show ap dot11 24ghz cleanair device type persistent	Displays CleanAir Interferers of type Persistent for 2.4-GHz band

Commands	Description
show ap dot11 24ghz cleanair device type superag	Displays CleanAir Interferers of type SuperAG for 2.4-GHz band
show ap dot11 24ghz cleanair device type tdd-tx	Displays CleanAir Interferers of type TDD Transmit for 2.4-GHz band
show ap dot11 24ghz cleanair device type video	Displays CleanAir Interferers of type Video Camera for 2.4-GHz band
show ap dot11 24ghz cleanair device type wimax-fixed	Displays CleanAir Interferers of type WiMax Fixed for 2.4-GHz band
show ap dot11 24ghz cleanair device type wimax-mobile	Displays CleanAir Interferers of type WiMax Mobile for 2.4-GHz band
show ap dot11 24ghz cleanair device type xbox	Displays CleanAir Interferers of type Xbox for 2.4-GHz band
show ap dot11 24ghz cleanair device type zigbee	Displays CleanAir Interferers of type zigbee for 2.4-GHz band
show ap dot11 5ghz cleanair air-quality summary	Displays CleanAir Air Quality (AQ) data for 5-GHz band
show ap dot11 5ghz cleanair air-quality worst	Displays CleanAir Air Quality (AQ) worst data for 5-GHz band
show ap dot11 5ghz cleanair config	Displays CleanAir Configuration for 5-GHz band
show ap dot11 5ghz cleanair device type all	Displays all CleanAir Interferers for 5-GHz band
show ap dot11 5ghz cleanair device type canopy	Displays CleanAir Interferers of type Canopy for 5-GHz band
show ap dot11 5ghz cleanair device type cont-tx	Displays CleanAir Interferers of type Continuous TX for 5-GHz band
show ap dot11 5ghz cleanair device type dect-like	Displays CleanAir Interferers of type DECT Like for 5-GHz band
show ap dot11 5ghz cleanair device type inv	Displays CleanAir Interferers of type WiFi Inverted for 5-GHz band
show ap dot11 5ghz cleanair device type jammer	Displays CleanAir Interferers of type Jammer for 5-GHz band
show ap dot11 5ghz cleanair device type nonstd	Displays CleanAir Interferers of type WiFi Inv. Ch for 5-GHz band

Commands	Description
show ap dot11 5ghz cleanair device type persistent	Displays CleanAir Interferers of type Persistent for 5-GHz band
show ap dot11 5ghz cleanair device type superag	Displays CleanAir Interferers of type SuperAG for 5-GHz band
show ap dot11 5ghz cleanair device type tdd-tx	Displays CleanAir Interferers of type TDD Transmit for 5-GHz band
show ap dot11 5ghz cleanair device type video	Displays CleanAir Interferers of type Video Camera for 5-GHz band
show ap dot11 5ghz cleanair device type wimax-fixed	Displays CleanAir Interferers of type WiMax Fixed for 5-GHz band
show ap dot11 5ghz cleanair device type wimax-mobile	Displays CleanAir Interferers of type WiMax Mobile for 5-GHz band

Additional References

Related Documents

Related Topic	Document Title
CleanAir commands and their details	*CleanAir Command Reference, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*

MIBs

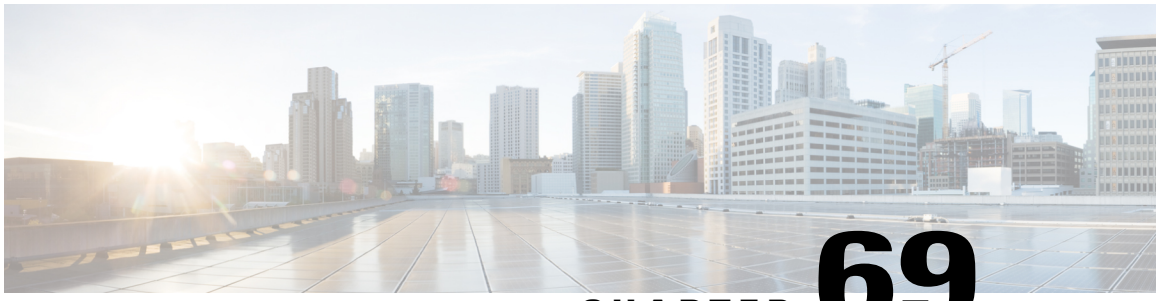
MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

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

PART XIII

Mobility



CHAPTER 69

Information About Mobility

Overview

The CiscoWLC 5700 Series Controller delivers more services at access layer other than merely providing increased speeds and feeds. Wireless services is now integrated with the Cisco Catalyst 3850 Switch, which ensures that the access layer switch terminates the wireless users data plane, thereby delivering on the promise of Cisco's unified architecture. Unification implies that mobility services are provided to both wireless and wired stations.

The Controller provides seamless roaming, which requires transparency of the network configuration and deployment options to the client.

From the end user's perspective, any mobility event must not change its IP address, its default router or DHCP server. This means that as stations roam, they must be able to

- Send an ARP to their default router, or
- Transmit a DHCP request to the server that had previously assigned their address.

From the infrastructure's perspective, as mobility events occur, the station's traffic must follow its current point of attachment, which can either be a mobility agent (MA) or mobility controller (MC). This must be true regardless of whether the station has moved to a network that is configured for a different subnet. The period from which the station is not receiving traffic following its mobility event must be as short as possible, even below 40 ms whenever possible, which includes any authentication procedures that are required.

From the infrastructure's perspective, the mobility management solution must have four main components, and all of these functions must be performed within the constraints of roaming:

- Initial Association–This function is used to identify the user's new point of attachment in the network.
- Context Transfer–This function is used to transfer state information associated with the station. This ensures that the station's static and real-time policies, including security and application ACLs, and services, remain the same across handoffs.
- Handoff–This function is used to signal that the station's point of attachment has changed, and control of the station should be relinquished by the previous access Controller .
- Data Plane –This function is typically tied to the handoff process, and ensures that the station's traffic continues to be delivered and received from the station without any noticeable performance degradation.

Wired and Wireless Mobility

One of the key features of the Converged access solution (applicable to both the Cisco Catalyst 3850 Switch and Cisco WLC 5700 Series Controller) is its ability to provide a device with an IP address and maintain its session persistence, across mobility events from ethernet connections to wireless and vice-versa. This feature allows users to remain on an ethernet network when possible, and make use of the freedom of mobility associated with wireless when necessary.

This feature leverages support from both the client and the infrastructure and uses the two factor authentication-device and user. The device authentication credentials is cached in the mobility controller (MC). When a device transitions across link layers, the device credentials is validated, and if a match is found, the MC ensures that the same IP address is assigned to the new interface.

Features of Mobility

The Controller has the following features:

- Mobility Controller (MC)— The controller provides mobility management services for inter-peer group roaming events. The MC provides a central point of contact for management and policy based control protocols, such as RADIUS. This eliminates the need for the infrastructure servers to maintain a user's location as it transitions throughout the network. The MC sends the configuration to all the mobility agents under its sub-domain of their mobility configuration, peer group membership and list of members. A sub-domain is synonymous to the MC that forms it. Each sub-domain consists of an MC and zero or more access switches that have AP's associated to them.
- Mobility Agents (MA)— A mobility agent is either an access switch that has a wireless module running on it or an MC with an internal MA running on it. A mobility agent is the wireless component that maintains client mobility state machine for a mobile client that is connected via an AP to the device that the MA is running on.
- Mobility Sub Domain— It is an autonomous portion of the mobility domain network. A mobility sub-domain comprises of a single mobility controller and its associated mobility agents (MAs).

  Note Even when more than one mobility controller is present, only one MC can be active at any given time.

  A mobility sub-domain is the set of devices managed by the active mobility controller. A mobility sub-domain comprises of a set of mobility agents and associated access points, across which fast roaming is required.

- Mobility Group— A collection of mobility controllers (MCs) across which fast roaming is supported. The concept of mobility group is the same as a collection of buildings in a campus across which frequent roaming is expected.
- Mobility Domain— A collection of mobility sub-domains across which mobility is supported. The term mobility domain may be the same as a campus network.
- Mobility Oracle (MO)—The mobility oracle acts as the point of contact for mobility events that occur across mobility sub-domains. It also maintains a local database of each station in the entire mobility domain, their home and current sub-domain. A mobility domain includes one or more mobility oracle, though only one would be active at any given time.
- Mobility Tunnel Endpoint (MTE)— The mobility tunnel endpoint (MTE) provides data plane services for mobile devices through the use of tunneling. This minimizes the impact of roaming events on the network by keeping the user's point of presence on the network a constant.
- Point of Attachment— A station's point of attachment is where its data path is initially processed upon entry in the network. This could either be the access switch that is currently providing it service, or the WLAN controller in the case of a legacy deployment.
- Point of Presence— A station's point of presence is the place in the network where the station is being advertised. For instance, if an access switch is advertising reachability to the station via a routing protocol, the interface on which the route is being advertised is considered the station's point of presence.
- Peer Group— A peer group is a statically created list of neighboring access switches between which fast mobility services is provided. A peer group limits the scope of interactions between switches during handoffs to only those that are geographically proximate.
- Station— A user's device that connects to and requests service from the network. The device may have a wired, wireless or both interfaces.
- Known Switch— A peer switch that is part of the local switch's peer group.
- Unknown Switch— A peer access switch that is not part of the local switch's peer group.
- Foreign Mobility Controller— The mobility controller providing mobility management service for the station in a foreign mobility sub-domain. The foreign mobility controller acts as a liaison between access switches in the foreign sub-domain and the mobility controller in the home domain.
- Foreign Mobility Sub-Domain— The mobility sub-domain, controlled by a mobility controller, supporting a station which is anchored in another mobility sub-domain
- Foreign Switch— The access switch in the foreign mobility sub-domain currently providing service to the station.
- Anchor Mobility Controller— The mobility controller providing a single point of control and mobility management service for stations in their home mobility sub-domain.
- Anchor Mobility Sub-Domain— The mobility sub-domain, controlled by a mobility controller, for a station where its IP address was assigned.
- Anchor Switch— The switch in the home mobility sub-domain that last provided service to a station.

Sticky Anchoring for Low Latency Roaming

Sticky Anchoring ensures low roaming latency from the client's point of presence is maintained at the switch where the client initially joins the network. It is expensive to apply client policies at a switch for a roaming client. There can be considerable delay as it involves contacting the AAA server for downloadable ACLs which is not acceptable for restoring time sensitive client traffic.

To manage this delay, when the client roams between APs connected to different switches , irrespective of whether it is an intra sub-domain roam or inter sub-domain roam, the client traffic is always tunneled to the switch where the client first associates. The client is anchored at its first point of attachment for its lifetime in the network.

This behavior is enabled by default. You can also disable this behavior to allow the client anchoring only for inter-subnet roams. This configuration is per WLAN config and is available under the WLAN config mode. The customer can configure different SSIDs for time sensitive and non time sensitive applications.

Bridge Domain ID and L2/L3 Roaming

Bridge domain ID provides the mobility nodes with information to decide on specific roam type, either as L2 or L3 roam. It also allows the network administrators to reuse the VLAN IDs across network distribution. When the VLAN IDs do not have the associated subnet configurations, they may require additional parameter to use in conjunction with VLAN ID. The network administrator ensures that the given VLAN under the same bridge domain ID are associated with the unique subnet. The mobility nodes will first check for the bridge domain ID for the given node and the VLAN ID associated with the client to identify the roam type. The bridge domain ID and the VLAN ID must be same to treat a roam as L2 roam.

The bridge domain ID is configured for each SPG when creating a SPG and later on the MC. The bridge domain ID could be same for more than one SPG and all the MAs under the SPG will share the same bridge domain ID. This information is pushed to the MAs as part of the configuration download when MA comes up initially. If the bridge domain ID is modified when the system is up, it will be pushed to all the MAs in the modified SPG and will take immediate effect for the future roams.

Note The MC can also have a bridge domain ID for it self, as the MC can also be part of a SPG.

Link Down Behavior

This section provides information about data synchronization between MA-MC and MC-MO when MC or MO faces downtime in absence of redundancy manager. When Keepalive is configured between MA-MC or MC-MO the clients database is synchronized between the MO and the MCs and the MC and its MAs respectively.

The database synchronization message contains following information:

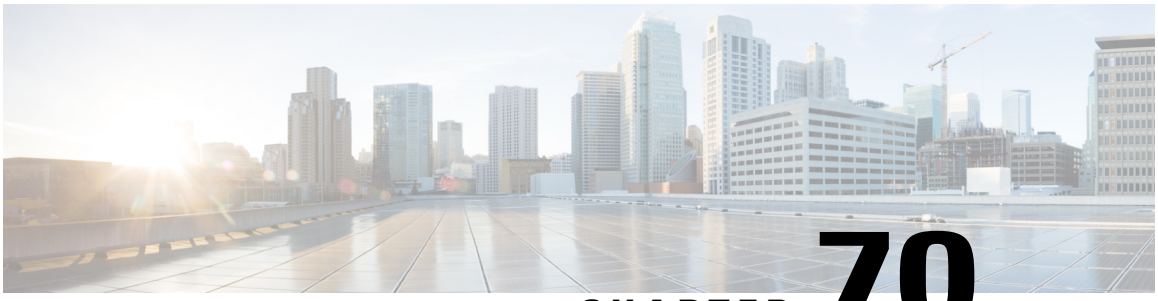
- MORE—This field is used to check if more data, as part of database synchronization message, is present. This field is basically used to split the larger database synchronization message into smaller chunks.
- USE_TIMESTAMP—This field is used to indicate if MA has timestamp embedded in the message. This can be removed when all the mobility deployments are required to be present with NTP server to use timestamp as mechanism to detect the stale messages.

- Client database entry—This is added using TLVs from mobility messages. The TLVs that are used to update the client database entry are used. These TLVs are encoded by MC and sent as part of database synchronization message. TLVs are used to allow backward compatibility so that inter release database synchronization between MC and MO also works.

Platform Specific Scale Requirement for the Mobility Controller

The Mobility Controller (MC) role is supported on a number of different platforms like, the Cisco WLC 5700 Series, CUWN and Catalyst 3850 Switches. The scale requirements on these three platforms are summarized in the table below:

Scalability	Catalyst 3850 as MC	Cisco WLC 5700 as MC	CUWN 5508 as MC	WiSM2 as MC
Max number of MC in Mobility Domain	8	72	72	72
Max number of MC in Mobility Group	8	24	24	24
Max number of MAs in Sub-domain (per MC)	16	350	350	350
Max number of SPGs in Sub-domain (per MC)	8	24	24	24
Max number of MAs in a SPG	16	64	64	64



CHAPTER 70

Mobility Network Elements

Mobility Agent

The Mobility Agent (MA) interacts with the Mobility Controller (MC), which can be a Catalyst 3850 Switch, or a Cisco WLC 5760, or a Cisco Unified Wireless Networking Solution controller. The MA is responsible for:

- Handling the mobility events on the switch
- Configuring the datapath elements on the switch for mobility, and
- Communicating with the mobility controller

As MA, the controller performs the datapath functions by terminating the CAPWAP tunnels that encapsulate 802.11 traffic sourced by wireless stations.

This allows the controller to apply features to wired and wireless traffic in a uniform fashion. As far as controller is concerned, 802.11 is just another access medium.

The MA performs the following functions:

- Support the mobility protocol – The MA is responsible for responding in a timely manner, ensuring the controller is capable of achieving its roaming budget.
- Point of presence – If the wireless subnets are not available at the MC, the MA assumes the point of presence if the wireless client VLAN is not available at the new point of attachment and tunnel the client traffic accordingly.
- ARP Server – When the network is configured in a layer 2 mode, the MA is responsible for advertising reachability for the stations connected to it. If tunneling is employed, the ARP request is transmitted on behalf of the station through the tunnel, which the point of presence (anchor switch) would bridge onto its uplink interface.

- Proxy IGMP – The MA on the controller is responsible for subscribing to multicast groups on behalf of a station after a roaming event has occurred. This information is passed as part of the context to the new controller. This ensures the multicast flows follow the user as it roams.
- Routing – When the controller is connected to a layer 3 access network, the MA is responsible for injecting routes for the stations that are associated with it for which tunneling is not provided.
- 802.1X Authenticator – The authenticator function is included in the MA, and handles both wired and wireless stations.
- Secure PMK Sharing – When a station successfully authenticates to the network, the MA forwards the PMK to the MC. The MC is responsible for flooding the PMK to all the MAs under its sub-domain and to the peer MCs in the mobility group.

The MA also performs the following datapath functions:

- Mobility tunnel – If tunneling is used, the MA encapsulates and decapsulates packets from the mobility tunnel to the MC, and to other MA in the peer group, if the access switches are serving as points of presence. The MA supports the tunneling of client data traffic between the point of attachment and the point of attachment. The packet format used for other switches is CAPWAP with an 802.3 payload. The MA also supports reassembly and fragmentation for mobility tunnels.
- Encryption – The mobility control traffic between the mobility nodes is DTLS encrypted. The MA also encrypts the CAPWAP control and data (optional) at the point of attachment.
- CAPWAP – The controller supports the CAPWAP control and data planes. The controller forwarding logic is responsible for terminating the CAPWAP tunnels with 802.11 as well as 802.3 payloads. Since support for large frames (greater than 1500bytes) is not universally available, the controller supports CAPWAP fragmentation and reassembly.

Mobility Controller

The main function of Mobility Controller is to handle the events in control plane. The other features of the mobility Controller are:

- Station Database - The Mobility Controller maintains a database of all the clients that are connected within the local mobility sub-domain.
- Mobility Protocol – The MC supports the mobility protocol which ensures the target roaming point responds in a timely manner and achieves the 150ms roaming budget
- Interface to Mobility Oracle – The Mobility Controller acts as a gateway between the Controller and the Mobility Oracle. When the Mobility Controller does not find a match in its local database, it suggests a match for a wireless client entry (in its database) and forwards the request to the Mobility Oracle, which manages the Mobility Domain.

Note Mobility Oracle function can be enabled on an MC only if it is supported by the platform.

- ARP Server - When tunneling is employed for a station, its point of presence on the network is the Mobility Tunnel Endpoint (MTE). The Mobility Controller responds to any ARP requests received for the stations it is responsible for.

- Configures MTE - The Mobility Controller is the control point for the Controller for all mobility management related requests. When a change in a station's point of attachment occurs, the Mobility Controller is responsible for configuring the forwarding policy on the MTE.
- NTP Server - The Mobility Controller acts as an NTP server to the Cisco 5700 Wireless LAN Controller and supports all the nodes to have their clocks synchronized with it.

Mobility Oracle

The Mobility Oracle is responsible for the complete mobility domain and consists of the following features:

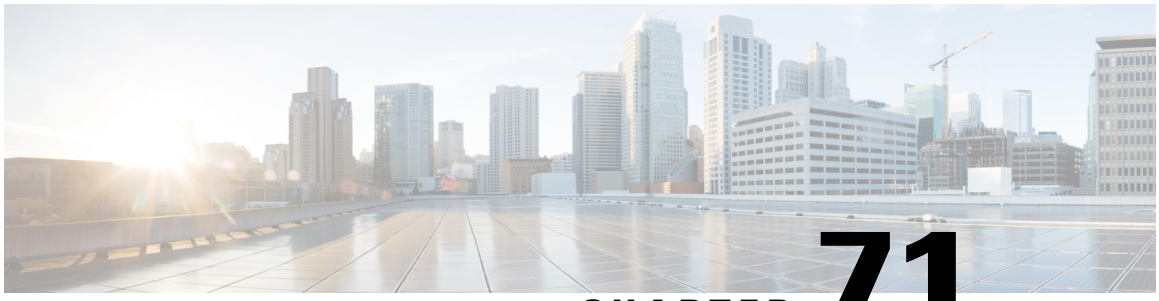
- Station Database – The Mobility Oracle maintains a database of all stations that are serviced within the mobility domain. This database is populated during the Mobility Oracle's interactions with all the Mobility Controllers, in all of the mobility sub-domains it supports.
- Interface to Mobility Controller – When the Mobility Oracle receives a request from a Mobility Controller, it performs a station lookup, and forwards, whenever needed, the request to the proper Mobility Controller.
- NTP Server – The Mobility Oracle acts as an NTP server to the Mobility Controllers and synchronizes all the Controller clocks within the mobility domain.

Guest Controller

The guest access feature provides guest access to wireless clients. The guest tunnels use the same format as the mobility tunnels. Using the guest access feature, there is no need to configure guest VLANs on the access switch. Traffic from the wired and wireless clients terminates on Guest Controller. Since the guest VLAN is not present on the access switch, the traffic is tunneled to the MTE over the existing mobility tunnel, and then via a guest tunnel to the Guest Controller.

The advantage of this approach is that all guest traffic passes through the MTE before it is tunneled to the Guest Controller. The Guest Controller only needs to support tunnels between itself and all the MTEs.

The disadvantage is that the traffic from the guest client is tunneled twice - once to the MTE and then again to the Guest Controller. If encryption is desired on the guest traffic, then both the Switch-to-MTE tunnel and the MTE-to-Guest Controller tunnel need to be encrypted.



CHAPTER 71

Mobility Control Protocols

About Mobility Control Protocols

The mobility control protocol is used regardless of whether tunneled or routed. The mobility control protocol is used for mobility events between the MO, MC and MA.

The mobility architecture uses both,

- Distributed approach, using the direct communication with the switches in their respective SPG, as well as
- Centralized approach, using the MC and MO.

The goal is to reduce the overhead on the centralized MC, while limiting the interactions between switches to help scale the overall system.

Initial Association and Roaming

The following scenarios are applicable to the mobility management protocol:

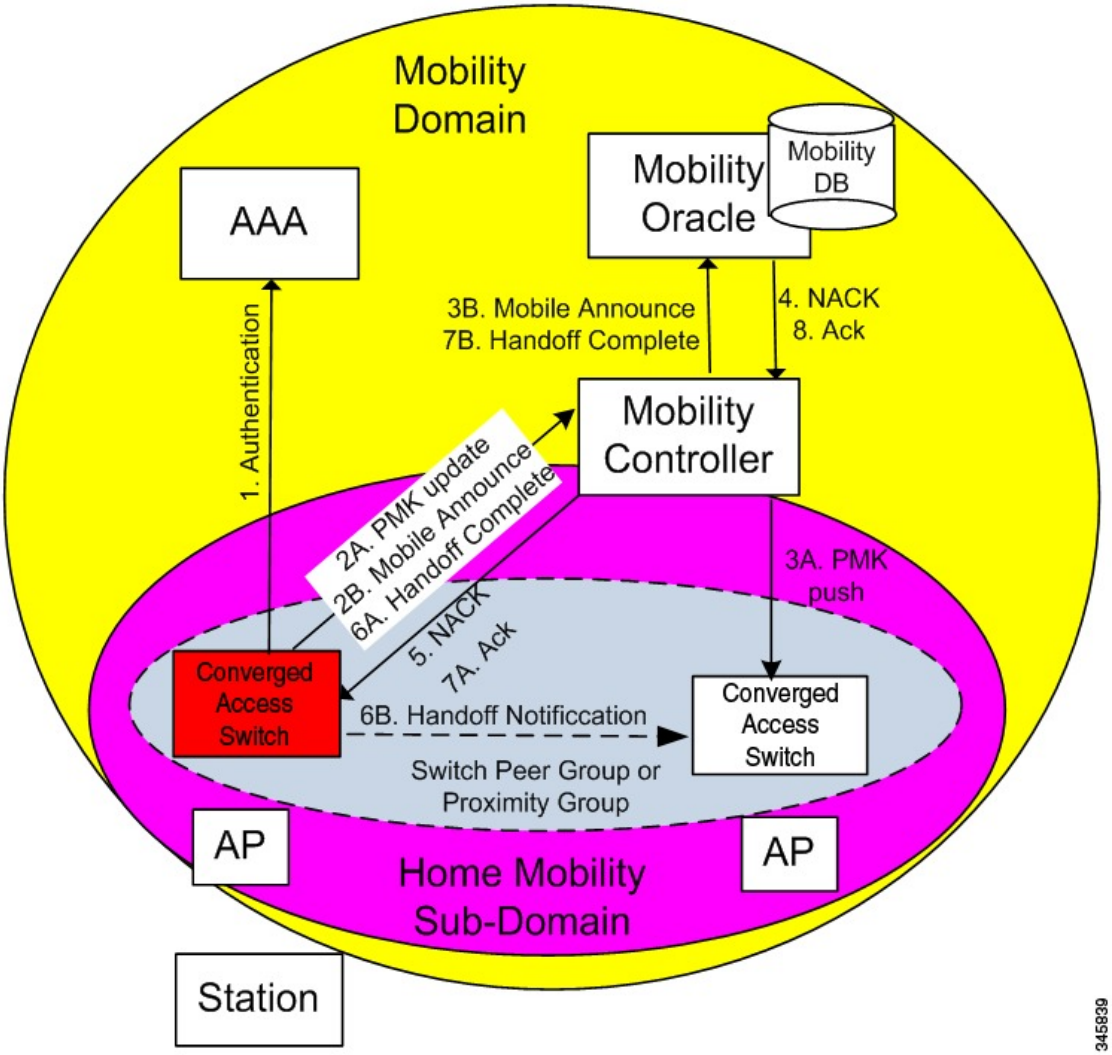
- Initial Association
- Intra Switch Roam
- Intra Switch Peer Group Roam

- Inter Switch Peer Group Roam
- Inter Sub-Domain Roam
- Inter Group Roam

Initial Association

The illustration below explains the initial association process followed by thecontroller:

Figure 47: Initial Association



1 When a station initially associates with a mobility agent, the MA performs a lookup to determine whether keying information for key caching is locally available in the MA. If no keying information is available, which is the case when the station first appears in the network, the controller prompts the device to

authenticate itself to generate the Pairwise Master Key (PMK). The PMK is generated on the client and the RADIUS server side, and the RADIUS sever forwards the PMK to the authenticator, the MA.

2 The MA sends the PMK to the MC.

3 After receiving the PMK from the MA, the MC transmits the PMK to all the MAs in its sub-domain, and to all the other MCs in its mobility group.

4 The mobility group is a single key domain. This ensures that 802.11r compliant stations recognize the key domain, and attempts to utilize the fast transition procedures defined in 802.11r.

Note The 802.11r protocol defines a key domain, which is a collection of access points that share keying information.

5 (Refer to step 2B in the illustration). Since the station is new to the mobility sub-domain, as indicated by the fact that the PMK is not in the MA local key cache, the MA transmits a mobile announce message to the MC.

6 The MC checks if the client exists in its database. As the client cannot be found, the MC in turn forwards it to the MO, if available.

7 (Refer to step 5 in the illustration). As the station is new to the network, the MO returns a negative response (NACK), which is forwarded by the MC to the controller. If the Mobility Oracle is not available then the MC is responsible for not responding to the Mobile Announce.

8 The MA on the controller informs the MC about the station's new point of attachment via the Handoff Complete message.

9 The MA then informs the other MAs in its switch peer group (SPG) about the station's new point of attachment via the Handoff Notification message. It is necessary to transmit this notification to the MAs in its SPG to allow local handoff without interacting with the MC. The Handoff Notification message sent to MAs in SPG need not carry all the information in Handoff Complete message sent to the MC.

10 (Refer to step 7B in the illustration). The MC updates its database and forwards the Handoff Complete message to the Mobility Oracle. This ensures that the Mobility Oracle's database is updated to record the station's current home mobility sub-domain.

To eliminate race conditions that could occur with devices moving quickly across controller, regardless of whether they are within a mobility sub-domain or not, the messages between MA and MC/MO are time synchronized. This would allow the MC and MO to properly process requests, if they are received out of order.

The Handoff Notification sent to MAs in the SPG are not acknowledged.

Intra Switch Handoff

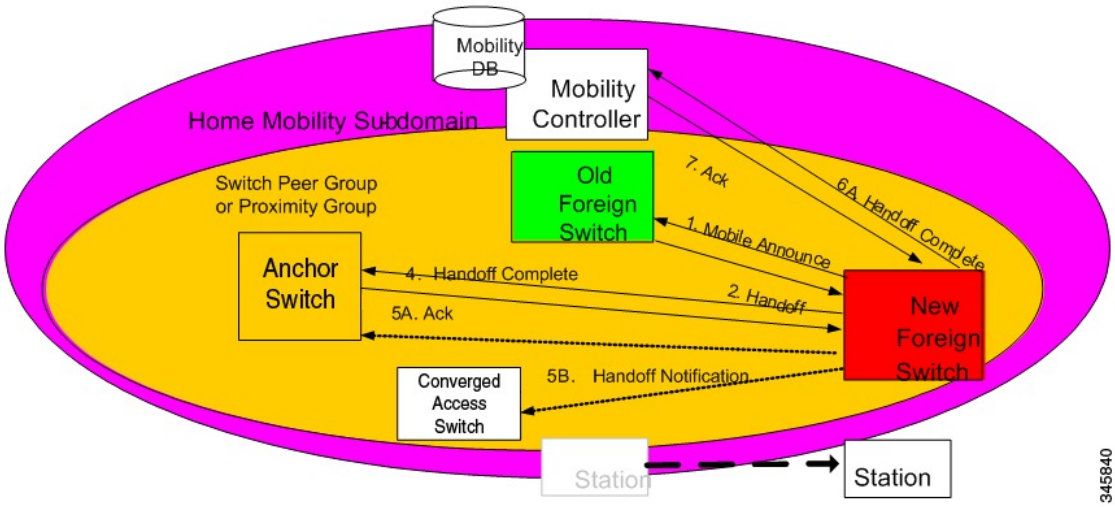
Mobility events within an MA are completely transparent to the SPG and the MC. When a station moves across APs on the same MA and attempts to perform a fast handoff, the PMK is present on the MA. The MA will complete the fast handoff without invoking any additional signal.

Intra Switch Peer Group Handoff

The switch peer group (SPG) is a group of MAs between which users may roam, and expect fast roaming services. Allowing the MA to handoff directly within a SPG reduces the overhead on the MC as it requires fewer messages to be exchanged.

After the initial association is complete the station moves to another MA belonging to its SPG. In an intra switch peer group roam, the initial association, the stations PMK was forwarded to all MAs in the mobility sub-domain.

Figure 48: Intra Switch Peer Group Handoff



The following process explains the intra switch peer group handoff:

1. In the initial association example, the Handoff Notification message is sent to all MAs in its SPG to know the station's current point of attachment.
2. The new MA sends a unicast Mobile Announce message to the previous MA to which the client is associated.
3. After the handoff completion, the new MA transmits a Handoff Complete message to the MC.
4. The new controller sends a Handoff Notification to all MA in its own SPG to inform them about the clients new point of presence.

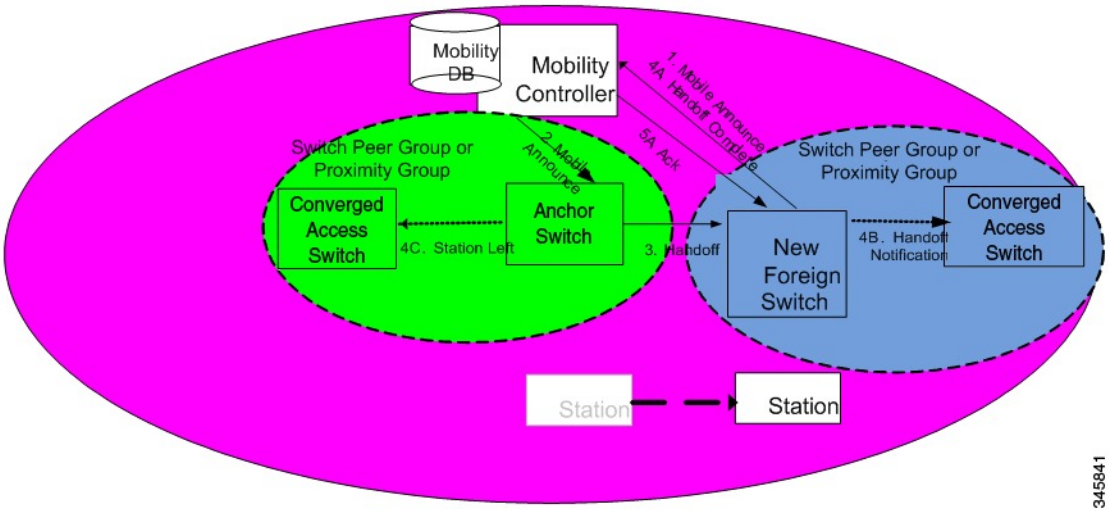
Inter Switch Peer Group Handoff

The Intra SPG roams do not cover all possible scenarios and there can be cases where it is possible for mobility events to occur between two MAs that are not in the same SPG.

When a MA does not have any information about a station's current point of attachment, because of the Handoff Notification message getting lost in the network, or because of the the station roaming to an MA that is not in the new SPG, the MA consults the MC. The MC provides information about the clients point of

presence within the mobility sub-domain. This eliminates the need to consult all other MCs within the mobility sub-domain.

Figure 49: Inter Switch Peer Group Handoff



The image above illustrates an example of a mobility event that occurs across MAs that are not in the same SPG, but within the same mobility sub-domain.

Note The MA color matches the circle representing its SPG.

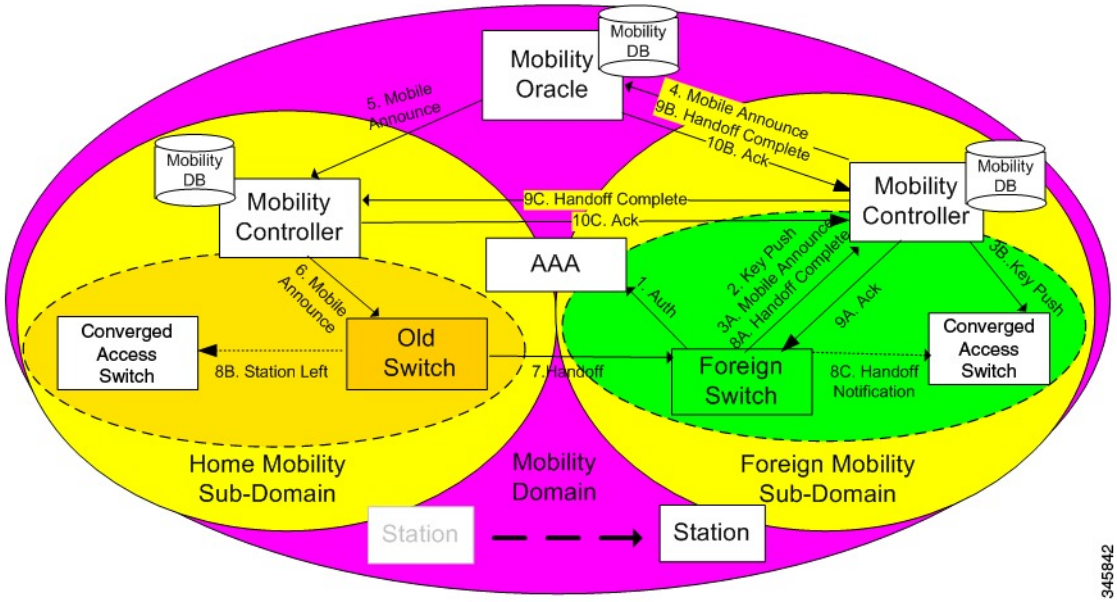
1. The new MA will have the PMK for the station, which was forwarded to each MA in the mobility sub-domain upon client initial authentication.
2. Since the MA had not been previously notified of the station's presence on a neighboring MA inside a different SPG transmits the mobile announce to the sub-domain's MC.
3. (Refer to step 2 in the illustration) On receiving the mobile announce message, the MC performs a lookup in its database, and forwards the request to the MA that was previously providing service to the station. This information is known to the MC through a previously received Handoff Complete message sent in a reliable fashion from the old MA.
4. (Refer to step 3 in the illustration) The old MA, shown in green above, transmits a Handoff message directly to the new MA.
5. The old MA needs to notify other MAs within its SPG of the fact that the station has left the group using a Station Left message. This ensures that if the station were to come back to one of the MA , they would be aware of the fact that the station is no longer being serviced by the old MA.
6. Once the handoff is complete, the new MA transmits the Handoff Complete message in a reliable fashion to the MC.
7. The new MA then transmits the Handoff Notification to the other MAs within its SPG.

Inter Sub Domain Handoff

A sub-domain is an ensemble formed by a mobility controller and the mobility agents it directly manages. An inter sub-domain mobility event implies communication between two mobility controllers. These 2 mobility controllers can be configured with the same mobility group value and recognize each other. They will appear in each other's mobility list, or they can be configured with different mobility group values, and still recognize each other.

When the roaming event occurs across sub-domains between MCs in the same mobility group, the 802.11r key domain advertised by the new APs are the same. Additionally, the client PMK is also transmitted to all MCs upon the client's initial authentication. The new MC does not need to force the client to reauthenticate, and the new MC also knows which previous MC was managing the wireless client mobility.

Figure 50: Inter Sub Domain Handoff



The following steps are involved in the inter sub domain handoff, when mobility controllers belong to the same mobility group:

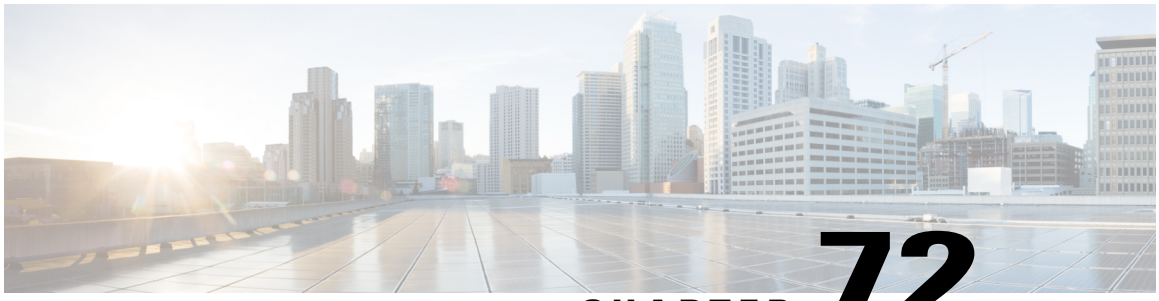
1. When a clients PMK was sent by the initial MA to all the MCs in the mobility group, the new MA already had already received the client PMK from its MC, and re-authentication is not required.
2. The new MA was not notified previously of the station's presence on a neighboring MA inside a different SPG it transmits the mobile announce to the sub-domain's MC.
3. On receiving the mobile announce message, the MC forwards the mobile announce to the MO, which performs a lookup in its database, and forwards the request to the MC that was previously providing service to the station.
4. The previous MC, in turn, forwards the request to the MA that was previously providing service to the station.
5. The old MA, shown in yellow color above, transmits a Handoff message directly to the new MA.

6 The old MA must notify the other MAs within its SPG of the fact that the station has left the SPG using a Station Left message. This ensures that if the station comes back to one of the MA , the MA is aware of the fact that the station is no longer serviced by the old MA.
7 Once the handoff is complete, the new MA transmits the Handoff Complete message in a reliable fashion to the new Mobility Controller.
8 The new MA then transmits the Handoff Notification to all other MAs.
9 The new MC then transmits the Handoff Complete to the old MC.

Inter Mobility Group Handoff

A mobility group is formed by MCs sharing the same mobility group name, and knowing each other.

Since the roaming event occurs across mobility groups, the 802.11r key domain advertised by the new APs differ. This forces the client to re-authenticate. They are propagated only within a mobility group, and roaming across mobility groups requires the stations to re-authenticate when they cross mobility group boundaries. When the authentication is complete, the PMK that is generated is pushed to the MAs and MCs within the same mobility group. The stations cache the PMK from the previous sub-domain because each PMK is associated to a given sub-domain (802.11y key domain). This ensures that you do not have to re-authenticate when the PMK roams back to the previous sub-domain within the pmk cache timeout interval. The remaining procedure follows the inter-sub-domain handoff steps, except that these steps relate to inter mobility group roaming.



CHAPTER 72

Intra Sub Domain Mobility

Overview

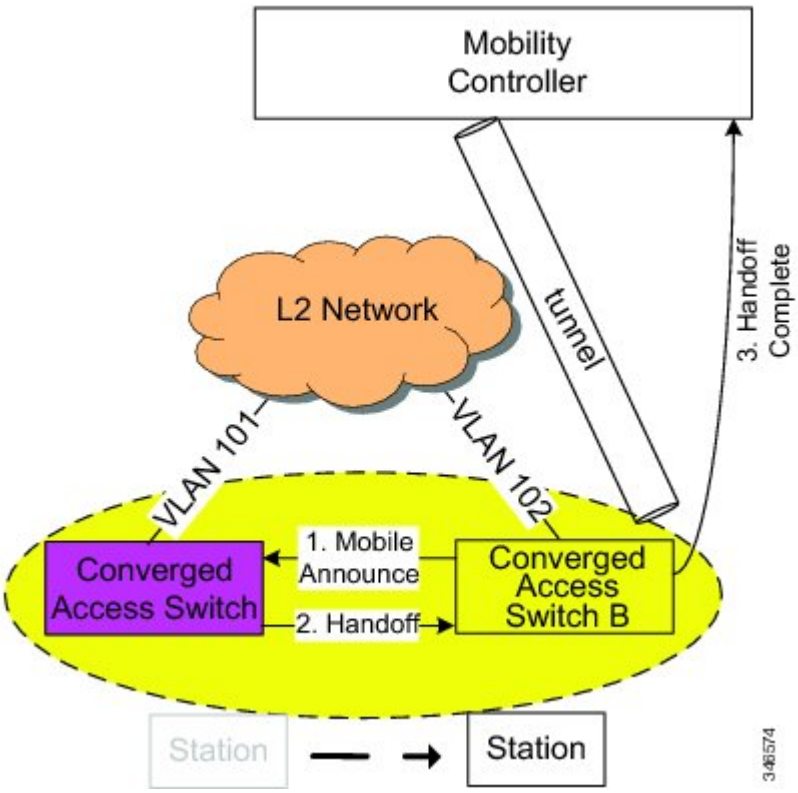
This section explains the mobility events that occur within a mobility sub-domain.

Layer 2 Roaming

This section explains the bridging concept where a station roams across the controller.

When a station roams acrosscontrollers and if its SSID and VLAN mapping matches, it is called a layer 2 roam.

Figure 51: Layer 2 Roaming



In the illustration, when a station roams acrosscontrollers the target switch has access to the station's VLAN or subnet. When the handoff is complete, the MC is informed through the Handoff Complete message. The new controllerbecomes the new point of presence for the station, and is responsible for advertising serviceability for the station directly.

Layer 3 Roaming

Layer 3 roaming happens when a station roams to an controller where the same VLAN or subnet is not available.

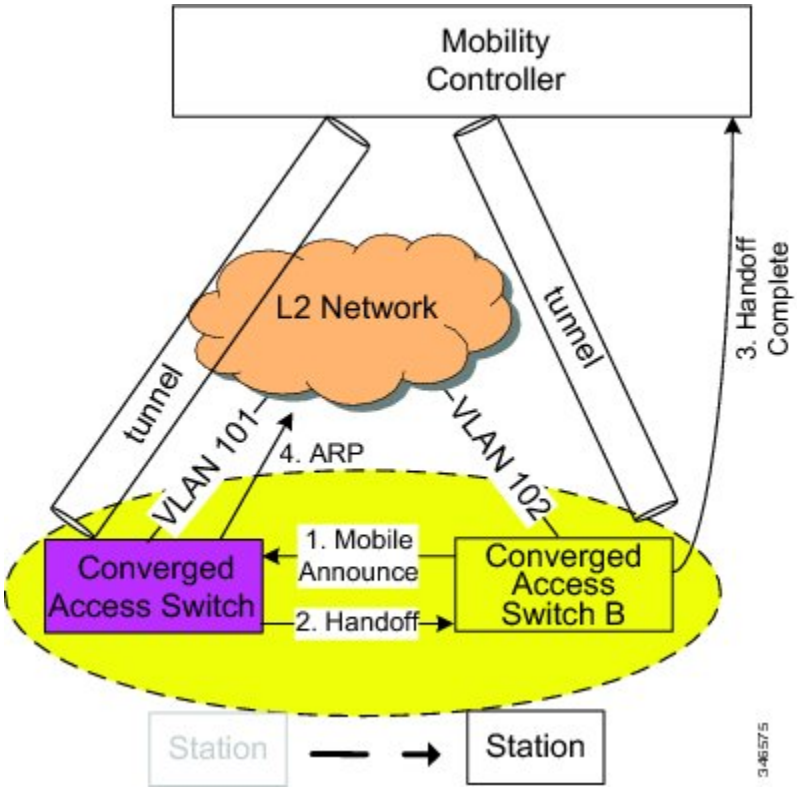
Point of Presence at Access Switch

The original controller becomes the anchor controller of the station. The new controller to which the station roams becomes the foreign controller of the station. In this scenario, the tunneling is direct between the foreign controller and the anchor controller.

In case of intra SPG, the tunneling is direct.

In case of inter SPG, the tunneling is directed through the MC.

Figure 52: Layer 3 Roaming



The illustration above explains the data path for for native stations and roamed stations.

- For native stations, the point of presence and the point of attachment is the same, Switch A.
- For an L3 roamed clients, the point of presence remains at the last switch with which the station was associated and where the station's subnet was available, while point of attachment moves to the client to which it has has roamed.

The following events explains the Layer 3 roam with Anchor Switch as the point of presence:

1. A station joins the network by associating to the AP on Switch A and is provided with an IP address from a subnet available on it.
   Its traffic is natively bridged at the switch and no tunneling is required. Switch A is both the point of presence and point of attachment for the station.
2. When the station roams to Switch B where the same subnet is not available, the information is provided in the handoff process.
3. In inter SPG roaming within the same sub-domain, when the handoff is complete, the MC is informed via the Handoff Complete message, and includes an indicator that traffic arriving on a tunnel from Switch B needs to be transmitted on a tunnel to Switch A.
4. In intra SPG within the same sub-domain, when the handoff is complete, the tunneling does not have to go through MC. The tunneling is direct between Switch B and Switch A.

5 The anchor switch, Switch A, continues to be the station's point of presence and Switch B becomes the point of attachment.

When the roamed station sends traffic to a wired host, the traffic is tunneled to the MTE, and from there to the Switch A. In the same way, since the point of presence is at Switch A, the traffic from the wired host comes to Switch A, and is tunneled first to the MTE, and from there to the foreign Switch B.

CHAPTER 73

Inter Sub Domain Mobility

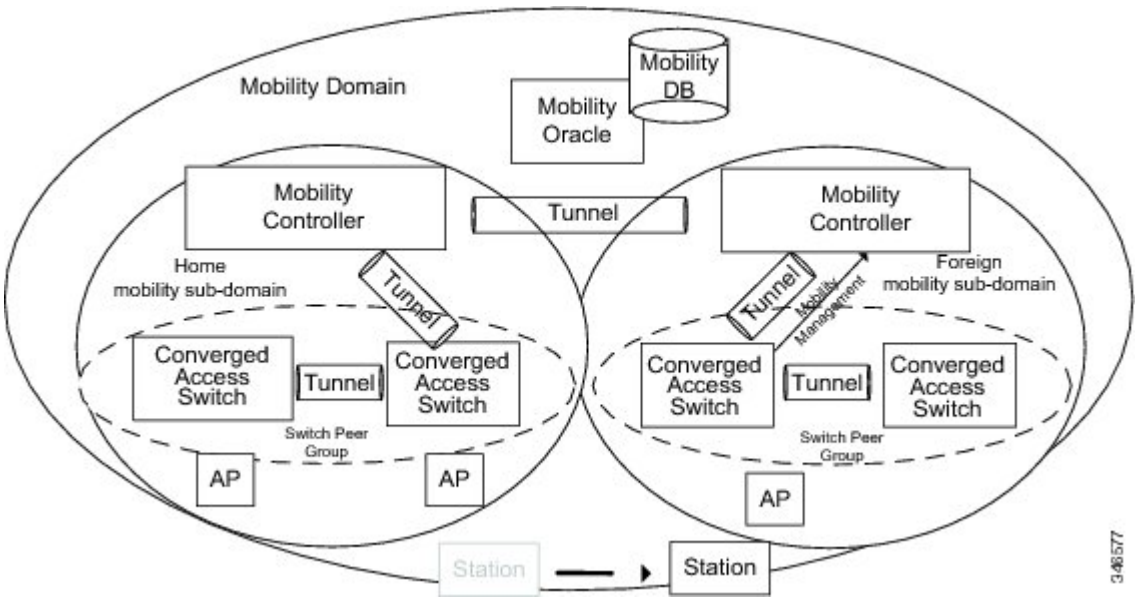
Introduction

This section focuses specifically on mobility events that occur across mobility sub-domains. An inter sub-domain mobility event occurs when a user moves from his home sub-domain to a foreign sub-domain. When the station initially roams to the foreign sub-domain, the foreign Mobility Controller signals the mobility event through the Mobility Oracle. This causes the station's traffic to be tunneled between the MCs in both sub-domains.

Point of Presence at Anchor Controller

When a station moves across sub-domains, the home and foreign MCs ensure that all of the station's traffic is tunneled.

Figure 53:



Tunneling across sub-domains when the client roams from Anchor Switch to Foreign Switch

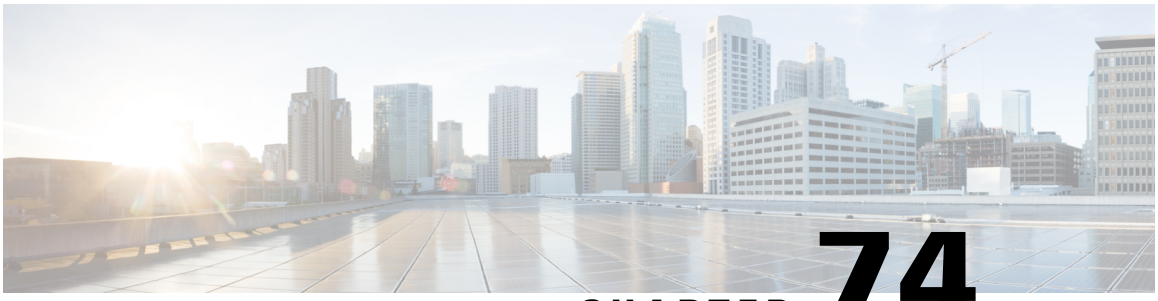
The following events happen when a station roams across sub-domains with the anchor controller in the home sub-domain as its point of presence:

- A station joins the network by associating to the AP on controller A and is provided with an IP address from a subnet available on it.
  Its traffic is natively bridged at the controller and no tunneling is required. controller A is both the point of presence and point of attachment for the station.
- When the station roams to controller B where the same subnet is not available, the information is provided in the handoff process.
- When the handoff is complete, the mobile announce message is sent from controller B to the MC in foreign sub-domain.
- The MC belonging to the foreign sub-domain will then tunnel the mobile announce message to the MO. This cause the stations traffic to be tunneled between the MCs in both the sub-domains.
- The MO will tunnel the mobile announce message to the MC in the home sub-domain.
- The MC in the home sub-domain will then tunnel the traffic to the anchor controller.

Tunneling across sub-domains with the Anchor Switch in the home sub-domain is the point of presence

The following events happen when a station roams across sub-domains with the anchor controller in the home sub-domain as its point of presence:

- A station joins the network by associating to the AP on controller A and is provided with an IP address from a subnet available on it.
  Its traffic is natively bridged at the controller and no tunneling is required. controller A is both the point of presence and point of attachment for the station.
- When the station roams to controller B where the same subnet is not available, the information is provided in the handoff process.
- When the handoff is complete, the anchor switch (Switch A) sends the handoff complete message to Switch B.
- The MC in the foreign sub-domain then forwards the handoff complete message to the MC in the home sub-domain.
- The MC belonging to the foreign sub-domain will then tunnel the traffic to the MO and finally reached the MC in the home sub-domain.
- The MC in the home sub-domain will then tunnel the traffic to the anchor controller.
- The anchor switch continues to be the point of presence for the station.
- The Switch B will contain the point of attachment.



CHAPTER 74

Configuring Mobility

Configuring Mobility Controller

Configuring Converged Access Controllers

Creating Peer Groups, Peer Group Member and Bridge Domain ID

Before You Begin

- On the MA, you can only configure the MCs IP address.
- On the MC, you can define the peer-group and each peer group members IP address.

SUMMARY STEPS

1. **wireless mobility controller peer-group** SPG1
2. **wireless mobility controller peer-group** SPG1 **member ip** 10.10.20.2 **public-ip** 10.10.20.2
3. **wireless mobility controller peer-group** SPG1 **member ip** 10.10.20.6 **public-ip** 10.10.20.6
4. **wireless mobility controller peer-group** SPG2
5. **wireless mobility controller peer-group** SPG2 **member ip** 10.10.10.20 **public-ip** 10.10.10.20
6. **wireless mobility controller peer-group** SPG1 **bridge-domain-id** 54

DETAILED STEPS

	Command or Action	Purpose
Step 1	**wireless mobility controller peer-group** SPG1 **Example:** `Controller(config)# wireless mobility controller peer-group SPG1`	Creates a peer group SPG1.
Step 2	**wireless mobility controller peer-group** SPG1 **member ip** 10.10.20.2 **public-ip** 10.10.20.2 **Example:** `Controller(config)# wireless mobility controller peer-group SPG1 member ip 10.10.20.2 public-ip 10.10.20.2`	Adds a member to peer group. **Note** The **member ip** 10.10.20.2 is used for NATed member and **public-ip** 10.10.20.2 is optional and used only when the member is not NATed.
Step 3	**wireless mobility controller peer-group** SPG1 **member ip** 10.10.20.6 **public-ip** 10.10.20.6 **Example:** `Controller(config)# wireless mobility controller peer-group SPG1 member ip 10.10.20.6 public-ip 10.10.20.6`	Adds another member to the peer group SPG1. **Note** The **member ip** 10.10.20.2 is used for NATed member and **public-ip** 10.10.20.2 is optional and used only when the member is not NATed.
Step 4	**wireless mobility controller peer-group** SPG2 **Example:** `Controller(config)# wireless mobility controller peer-group SPG2`	Creates another peer group SPG2.
Step 5	**wireless mobility controller peer-group** SPG2 **member ip** 10.10.10.20 **public-ip** 10.10.10.20 **Example:** `Controller(config)# wireless mobility controller peer-group SPG2 member ip 10.10.10.20 public-ip 10.10.10.20`	Adds a member to peer group SPG2. **Note** The **member ip** 10.10.20.2 is used for NATed member and **public-ip** 10.10.20.2 is optional and used only when the member is not NATed.
Step 6	**wireless mobility controller peer-group** SPG1 **bridge-domain-id** 54 **Example:** `Controller(config)# wireless mobility controller peer-group SPG1 bridge-domain-id 54`	Adds a bridge domain to SPG1 used for defining the subnet-vlan mapping with other SPGs.

This example shows how to create peer group and add members to it:

```
Controller(config)# wireless mobility controller peer-group SPG1
Controller(config)# wireless mobility controller peer-group SPG1
Controller(config)# wireless mobility controller peer-group SPG1 member ip 10.10.20.2
public-ip 10.10.20.2
Controller(config)# wireless mobility controller peer-group SPG1 member ip 10.10.20.6
public-ip 10.10.20.6
Controller(config)# wireless mobility controller peer-group SPG2
Controller(config)# wireless mobility controller peer-group SPG2 member ip 10.10.10.20
```

```
public-ip 10.10.10.20
Controller(config)# wireless mobility controller peer-group SPG1 bridge-domain-id 54
```

Configuring Local Mobility Group

Configuration for wireless mobility groups and mobility group members where the mobility group is a group of MCs.

Before You Begin

MCs can belong only to one mobility group, and can know MCs in several mobility groups.

SUMMARY STEPS

1. **wireless mobility group name** Mygroup
2. **wireless mobility group member ip** 10.10.34.10
3. **wireless mobility group keepalive interval** 5
4. **wireless mobility group keepalive count** 3

DETAILED STEPS

	Command or Action	Purpose
Step 1	**wireless mobility group name** Mygroup **Example:** `Controller(config)# wireless mobility group name Mygroup`	Creates a mobility group named Mygroup.
Step 2	**wireless mobility group member ip** 10.10.34.10 **Example:** `Controller(config)# wireless mobility group member ip 10.10.34.10`	Adds a mobility controller to the Mygroup mobility group.
Step 3	**wireless mobility group keepalive interval** 5 **Example:** `Controller(config)# wireless mobility group keepalive interval 5`	Configures the interval between two keep alives sent to a mobility member.
Step 4	**wireless mobility group keepalive count** 3 **Example:** `Controller(config)# wireless mobility group keepalive count 3`	Configures the keep alive retries before a member status is termed DOWN.

```
Controller(config)# wireless mobility group name Mygroup
Controller(config)# wireless mobility group member ip 10.10.34.10
Controller(config)# wireless mobility group keepalive interval 5
Controller(config)# wireless mobility group keepalive count 3
```

Adding a Peer Mobility Group

Before You Begin

MCs belong to only one group, and can know MCs in several groups.

SUMMARY STEPS

1. **wireless mobility group member ip** 10.10.10.24 **group** Group2

DETAILED STEPS

	Command or Action	Purpose
Step 1	**wireless mobility group member ip** 10.10.10.24 **group** Group2 **Example:** `Controller(config)# wireless mobility group member ip 10.10.10.24 group Group2`	Adds the member as a peer MC in a different group than the Mygroup.

Configuring Optional Parameters for Mobility Group

Use this configuration to disable the sticky anchor.

SUMMARY STEPS

1. **wlan** open21
2. **no mobility anchor sticky**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**wlan** open21 **Example:** `Controller(config)# wlan open20`	Configures a WLAN.
Step 2	**no mobility anchor sticky** **Example:** `Controller(config-wlan)# no mobility anchor sticky`	Disables the default sticky mobility anchor.

```
Controller(config)# wlan open20
Controller(config-wlan)# no mobility anchor sticky
```

Pointing the Mobility Controller to a Mobility Oracle

Before You Begin

You can configure a mobility oracle on a known mobility controller.

SUMMARY STEPS

1. **wireless mobility group member ip** 10.10.10.10 **group** Group3
2. **wireless mobility oracle ip** 10.10.10.10

DETAILED STEPS

	Command or Action	Purpose
Step 1	**wireless mobility group member ip** 10.10.10.10 **group** Group3 **Example:** `Controller(config)#wireless mobility group member ip 10.10.10.10 group Group3`	Creates and adds a MC to a mobility group.
Step 2	**wireless mobility oracle ip** 10.10.10.10 **Example:** `Controller(config)#wireless mobility oracle ip 10.10.10.10`	Configures the mobility controller as mobility oracle.

```
Controller(config)#wireless mobility group member ip 10.10.10.10 group Group3
Controller(config)#wireless mobility oracle ip 10.10.10.10
```

Configuring Guest Controller

A guest controller is used when the clients traffic is tunneled to an guest anchor controller in the de militarized zone (DMZ). The guest client is authenticated at the anchor controller using web based mechanism. The authentication mechanism is optional and the guest is allowed to pass traffic without authentication also.

Enable the WLAN on the MA on which the guest client connects with the mobility anchor address of the guest controller.

On the guest controller WLAN, which can be CUWN 5500 or 5760, configure the mobility anchors ip address as its own ip address. This allows the traffic to be tunneld to the guest controller from MA.

SUMMARY STEPS

1. **wlan** Mywlan1
2. **mobility anchor** <guest-anchors-ip-address>
3. **client vlan**<vlan-name>
4. **security open**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**wlan** Mywlan1 **Example:** `Controller(config)# wlan Mywlan1`	Creates a wlan for the client.
Step 2	**mobility anchor** <guest-anchors-ip-address> **Example:** `Controller(config-wlan)# mobility anchor 10.10.10.2`	Enables the guest anchors (GA) IP address on the MA.
Step 3	**client vlan**<vlan-name> **Example:** `Controller(config-wlan)# client vlan gc_ga_vlan1`	Assigns a vlan to the clients wlan.
Step 4	**security open** **Example:** `Controller(config-wlan)# security open`	Assigns a security type to the wlan.

```
Controller(config)# wlan Mywlan1
Controller(config-wlan)# mobility anchor 10.10.10.2
Controller(config-wlan)# client vlan gc_ga_vlan1
Controller(config-wlan)# security open
```

Configuring Guest Anchor

SUMMARY STEPS

1. **wlan** Mywlan1
2. **mobility anchor** <guest-anchors-own-ip-address>
3. **client vlan**<vlan-name>
4. **security open**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**wlan** Mywlan1 **Example:** `Controller(config)# wlan Mywlan1`	Creates a wlan for the client.

	Command or Action	Purpose
Step 2	**mobility anchor** <guest-anchors-own-ip-address> **Example:** `Controller(config-wlan)# mobility anchor 10.10.10.2`	Enables the guest anchors IP address on the guest anchor (GA). The GA assigns its own address on itself.
Step 3	**client vlan**<vlan-name> **Example:** `Controller(config-wlan)# client vlan gc_ga_vlan1`	Assigns a vlan to the clients wlan.
Step 4	**security open** **Example:** `Controller(config-wlan)# security open`	Assigns a security type to the wlan.

```
Controller(config)# wlan Mywlan1
Controller(config-wlan)# mobility anchor 10.10.10.2
Controller(config-wlan)# client vlan gc_ga_vlan1
Controller(config-wlan)# security open
```

Configuring Converged Access Controller on 5508 or WiSM 2

Enabling the New Mobility

Before You Begin

You will require Cisco Unified Wireless Network 7.3 MR1, 8.0 or later to configure the new mobility architecture.

SUMMARY STEPS

1. **config mobility new-architecture enable**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**config mobility new-architecture enable** **Example:** `(Cisco Controller) >config mobility new-architecture enable`	Enables and installs the new mobility architecture on the CUWN based controller.

```
(Cisco Controller) >config mobility new-architecture enable
Enabling new-architecture would change mobility architecture from flat to hierarchical !!!
```

```
Configuration changes will be saved and System will be rebooted. !!!
Are you sure you want to continue? (y/n) y
```

Configuring Mobility Controller

This configuration shows how to change the MCs public address, or mobility group name.

SUMMARY STEPS

1.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**Example:**	

Creating Peer Groups, Peer Group Member and Bridge Domain ID

Before You Begin

- On the MA, you can only configure the MCs IP address.
- On the MC, you can define the peer-group and each peer group members IP address.

SUMMARY STEPS

1. **wireless mobility controller peer-group** SPG1
2. **wireless mobility controller peer-group** SPG1 **member ip** 10.10.20.2 **public-ip** 10.10.20.2
3. **wireless mobility controller peer-group** SPG1 **member ip** 10.10.20.6 **public-ip** 10.10.20.6
4. **wireless mobility controller peer-group** SPG2
5. **wireless mobility controller peer-group** SPG2 **member ip** 10.10.10.20 **public-ip** 10.10.10.20
6. **wireless mobility controller peer-group** SPG1 **bridge-domain-id** 54

DETAILED STEPS

	Command or Action	Purpose
Step 1	**wireless mobility controller peer-group** SPG1 **Example:** `Controller(config)# wireless mobility controller peer-group SPG1`	Creates a peer group SPG1.

	Command or Action	Purpose
Step 2	**wireless mobility controller peer-group** SPG1 **member ip** 10.10.20.2 **public-ip** 10.10.20.2 **Example:** `Controller(config)# wireless mobility controller peer-group SPG1 member ip 10.10.20.2 public-ip 10.10.20.2`	Adds a member to peer group. **Note** The **member ip** 10.10.20.2 is used for NATed member and **public-ip** 10.10.20.2 is optional and used only when the member is not NATed.
Step 3	**wireless mobility controller peer-group** SPG1 **member ip** 10.10.20.6 **public-ip** 10.10.20.6 **Example:** `Controller(config)# wireless mobility controller peer-group SPG1 member ip 10.10.20.6 public-ip 10.10.20.6`	Adds another member to the peer group SPG1. **Note** The **member ip** 10.10.20.2 is used for NATed member and **public-ip** 10.10.20.2 is optional and used only when the member is not NATed.
Step 4	**wireless mobility controller peer-group** SPG2 **Example:** `Controller(config)# wireless mobility controller peer-group SPG2`	Creates another peer group SPG2.
Step 5	**wireless mobility controller peer-group** SPG2 **member ip** 10.10.10.20 **public-ip** 10.10.10.20 **Example:** `Controller(config)# wireless mobility controller peer-group SPG2 member ip 10.10.10.20 public-ip 10.10.10.20`	Adds a member to peer group SPG2. **Note** The **member ip** 10.10.20.2 is used for NATed member and **public-ip** 10.10.20.2 is optional and used only when the member is not NATed.
Step 6	**wireless mobility controller peer-group** SPG1 **bridge-domain-id** 54 **Example:** `Controller(config)# wireless mobility controller peer-group SPG1 bridge-domain-id 54`	Adds a bridge domain to SPG1 used for defining the subnet-vlan mapping with other SPGs.

This example shows how to create peer group and add members to it:

```
Controller(config)# wireless mobility controller peer-group SPG1
Controller(config)# wireless mobility controller peer-group SPG1
Controller(config)# wireless mobility controller peer-group SPG1 member ip 10.10.20.2
public-ip 10.10.20.2
Controller(config)# wireless mobility controller peer-group SPG1 member ip 10.10.20.6
public-ip 10.10.20.6
Controller(config)# wireless mobility controller peer-group SPG2
Controller(config)# wireless mobility controller peer-group SPG2 member ip 10.10.10.20
public-ip 10.10.10.20
Controller(config)# wireless mobility controller peer-group SPG1 bridge-domain-id 54
```

Configuring Local Mobility Group

Configuration for wireless mobility groups and mobility group members where the mobility group is a group of MCs.

Before You Begin

MCs can belong only to one mobility group, and can know MCs in several mobility groups.

SUMMARY STEPS

1. **wireless mobility group name** Mygroup
2. **wireless mobility group member ip** 10.10.34.10
3. **wireless mobility group keepalive interval** 5
4. **wireless mobility group keepalive count** 3

DETAILED STEPS

	Command or Action	Purpose
Step 1	**wireless mobility group name** Mygroup **Example:** `Controller(config)# wireless mobility group name Mygroup`	Creates a mobility group named Mygroup.
Step 2	**wireless mobility group member ip** 10.10.34.10 **Example:** `Controller(config)# wireless mobility group member ip 10.10.34.10`	Adds a mobility controller to the Mygroup mobility group.
Step 3	**wireless mobility group keepalive interval** 5 **Example:** `Controller(config)# wireless mobility group keepalive interval 5`	Configures the interval between two keep alives sent to a mobility member.
Step 4	**wireless mobility group keepalive count** 3 **Example:** `Controller(config)# wireless mobility group keepalive count 3`	Configures the keep alive retries before a member status is termed DOWN.

```
Controller(config)# wireless mobility group name Mygroup
Controller(config)# wireless mobility group member ip 10.10.34.10
Controller(config)# wireless mobility group keepalive interval 5
Controller(config)# wireless mobility group keepalive count 3
```

Adding a Peer Mobility Group

Before You Begin

MCs belong to only one group, and can know MCs in several groups.

SUMMARY STEPS

1. **wireless mobility group member ip** 10.10.10.24 **group** Group2

DETAILED STEPS

	Command or Action	Purpose
Step 1	**wireless mobility group member ip** 10.10.10.24 **group** Group2 **Example:** `Controller(config)# wireless mobility group member ip 10.10.10.24 group Group2`	Adds the member as a peer MC in a different group than the Mygroup.

Configuring Optional Parameters for Mobility Group

Use this configuration to disable the sticky anchor.

SUMMARY STEPS

1. **wlan** open21
2. **no mobility anchor sticky**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**wlan** open21 **Example:** `Controller(config)# wlan open20`	Configures a WLAN.
Step 2	**no mobility anchor sticky** **Example:** `Controller(config-wlan)# no mobility anchor sticky`	Disables the default sticky mobility anchor.

```
Controller(config)# wlan open20
Controller(config-wlan)# no mobility anchor sticky
```

Pointing the Mobility Controller to a Mobility Oracle

Before You Begin

You can configure a mobility oracle on a known mobility controller.

SUMMARY STEPS

1. **wireless mobility group member ip** 10.10.10.10 **group** Group3
2. **wireless mobility oracle ip** 10.10.10.10

DETAILED STEPS

	Command or Action	Purpose
Step 1	**wireless mobility group member ip** 10.10.10.10 **group** Group3 **Example:** `Controller(config)#wireless mobility group member ip 10.10.10.10 group Group3`	Creates and adds a MC to a mobility group.
Step 2	**wireless mobility oracle ip** 10.10.10.10 **Example:** `Controller(config)#wireless mobility oracle ip 10.10.10.10`	Configures the mobility controller as mobility oracle.

```
Controller(config)#wireless mobility group member ip 10.10.10.10 group Group3
Controller(config)#wireless mobility oracle ip 10.10.10.10
```

Configuring the Mobility Oracle

Configuring Mobility Oracle on Converged Access Controller

This configuration shows how to configure mobility oracle on a converged access controller only.

Enabling the Mobility Oracle on the Controller

SUMMARY STEPS

1. **wireless mobility oracle**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**wireless mobility oracle** **Example:** `Controller(config)# wireless mobility oracle`	Enables the mobility oracle (MO) on the MC.

This example shows how to enable the mobility oracle on MC:

```
Controller(config)# wireless mobility oracle
```

Configuring Mobility Oracle on CUWN

Enabling Mobility Oracle on CUWN

This configuration shows how to enable the mobility oracle on the 5508 or WiSM2 controllers.

SUMMARY STEPS

1. **config mobility oracleenable**
2. **config mobility oracle ip** 10.10.10.5

DETAILED STEPS

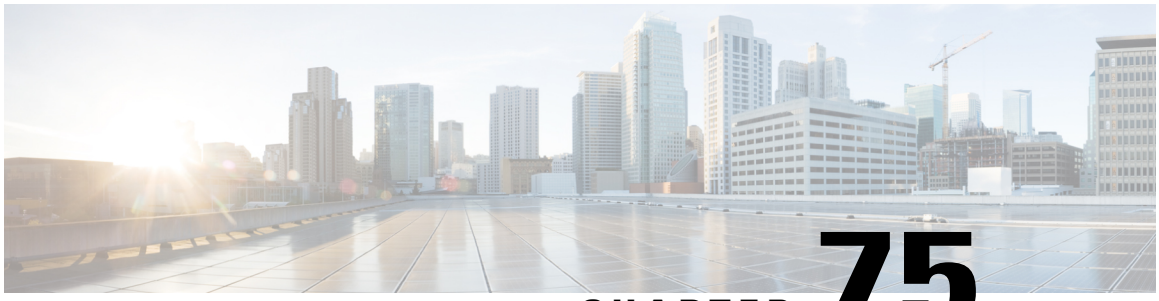
	Command or Action	Purpose
Step 1	**config mobility oracleenable** **Example:** `<cisco-controller> config wireless mobility oracle`	Enables the oracle on CUWN 5500.
Step 2	**config mobility oracle ip** 10.10.10.5 **Example:** `<cisco-controller> config wireless mobility oracle ip 10.10.10.5`	Configures the MC with MO's IP address.

This example shows how to enable oracle on CUWN and make the CUWN also act as MO:

```
<cisco-controller> config wireless mobility oracle
<cisco-controller> config wireless mobility oracle ip 10.10.10.5
```

PART XIV

IPv6



CHAPTER 75

Configuring IPv6 Client IP Address Learning

Prerequisites for IPv6 Client Address Learning

Before configuring IPv6 client address learning, configure the wireless clients to support IPv6.

Information About IPv6 Client Address Learning

Client Address Learning is configured on controller to learn the wireless client's IPv4 and IPv6 address and clients transition state maintained by the controller on an association, re-association, de-authentication and timeout.

There are three ways for IPv6 client to acquire IPv6 addresses:

- Stateless Address Auto-Configuration (SLACC)
- Stateful DHCPv6
- Static Configuration

For all of these methods, the IPv6 client always sends neighbor solicitation DAD (Duplicate Address Detection) request to ensure there is no duplicate IP address on the network. The controller snoops the client's NDP and DHCPv6 packets to learn about its client IP addresses.

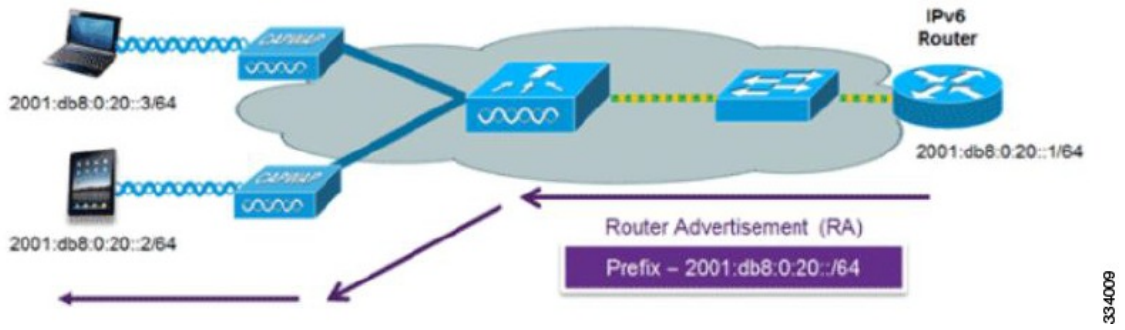
SLAAC Address Assignment

The most common method for IPv6 client address assignment is Stateless Address Auto-Configuration (SLAAC). SLAAC provides simple plug-and-play connectivity where clients self-assign an address based on the IPv6 prefix. This process is achieved when the IPv6 router sends out periodic Router Advertisement (RA) messages, which inform the client of the IPv6 prefix in use (the first 64 bits) and the IPv6 default gateway. From this point, clients can generate the remaining 64 bits of their IPv6 address based on two algorithms:

- EUI-64 which is based on the MAC address of the interface, or
- Private addresses that are randomly generated.

The choice of algorithm is up to the client and is often configurable. Duplicate Address Detection is performed by IPv6 clients in order to ensure that random addresses that are picked do not collide with other clients. The address of the router sending advertisements is used as the default gateway for the client.

Figure 54: SLAAC Address Assignment

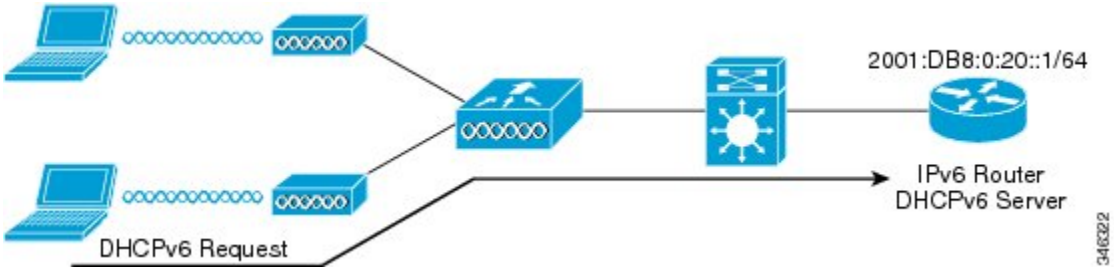


The following Cisco IOS configuration commands from a Cisco-capable IPv6 router are used to enable SLAAC addressing and router advertisements:

```
ipv6 unicast-routing
ipv6 dhcp pool IPV6_DHCPPOOL
interface Vlan20
description IPv6-SLAAC
ip address 192.168.20.1 255.255.255.0
ipv6 address 2001:DB8:0:20::1/64
ipv6 enable
ipv6 nd other-config-flag
ipv6 dhcp server IPV6_DHCPPOOL
end
```

Stateful DHCPv6 Address Assignment

Figure 55: Stateful DHCPv6 Address Assignment



The use of DHCPv6 is not required for IPv6 client connectivity if SLAAC is already deployed. There are two modes of operation for DHCPv6 called Stateless and Stateful.

The DHCPv6 Stateless mode is used to provide clients with additional network information that is not available in the router advertisement, but not an IPv6 address as this is already provided by SLAAC. This information can include the DNS domain name, DNS server(s), and other DHCP vendor-specific options. This interface configuration is for a Cisco IOS IPv6 router implementing stateless DHCPv6 with SLAAC enabled:

```
ipv6 unicast-routing
ipv6 dhcp pool IPV6_DHCPPOOL
address prefix 2001:db8:5:10::/64
domain-name cisco.com
dns-server 2001:db8:6:6::1
interface Vlan20
description IPv6-DHCP-Stateless
ip address 192.168.20.1 255.255.255.0
ipv6 nd other-config-flag
ipv6 dhcp server IPV6_DHCPPOOL
ipv6 address 2001:DB8:0:20::1/64
end
```

The DHCPv6 Stateful option, also known as managed mode, operates similarly to DHCPv4 in that it assigns unique addresses to each client instead of the client generating the last 64 bits of the address as in SLAAC. This interface configuration is for a Cisco IOS IPv6 router implementing stateful DHCPv6 with SLAAC disabled:

```
ipv6 unicast-routing
ipv6 dhcp pool IPV6_DHCPPOOL
address prefix 2001:db8:5:10::/64
domain-name cisco.com
dns-server 2001:db8:6:6::1
interface Vlan20
description IPv6-DHCP-Stateful
ip address 192.168.20.1 255.255.255.0
ipv6 address 2001:DB8:0:20::1/64
ipv6 nd prefix 2001:DB8:0:20::/64 no-advertise
ipv6 nd managed-config-flag
ipv6 nd other-config-flag
ipv6 nd other-config-flag
ipv6 dhcp server IPV6_DHCPPOOL
end
```

Static IP Address Assignment

Statically configured address on a client.

Binding Table Manager

The binding-table manager used by the controller is responsible for managing the L2/L3 binding table. The entry life-cycle is driven by a finite state machine. An entry is created as INCOMPLETE, moves to REACHABLE when binding is known, moves back and forth from REACHABLE to VERIFY if tracking is enabled, at some point moves to STALE when the client stops talking, and finally the entry is deleted.

The important states in a binding table follow:

- Incomplete – An entry is set in this state when it does not have the L3/L2 binding yet
- Reachable – An entry is moved to REACHABLE state when L3 to L2 address mapping is obtained when snooping the RS, NS, NA or DHCP packets from a client
- Verify – An entry is moved into this state, when L3/L2 binding is known but must be verified. This state is reached when tracking is enabled
- Stale – An entry is moved into this state, when the binding table manager does not hear any packets from the client for the configured reachable timer
- Down – An entry is set in this state when the interface, from where the packet is received, goes down

RA Guard

IPv6 clients configure IPv6 addresses and populate their router tables based on IPv6 router advertisement (RA) packets. The RA guard feature is similar to the RA guard feature of wired networks. RA guard increases the security of the IPv6 network by dropping the unwanted or rogue RA packets that come from wireless clients. If this feature is not configured, malicious IPv6 clients could announce themselves as the router for the network often with high priority, which would take higher precedence over legitimate IPv6 routers.

RA-Guard also examines the incoming RA's and decides whether to switch or block them based solely on information found in the message or in the L2-device configuration. The information available in the frames received is useful for RA validation:

- Port on which the frame is received
- IP source address
- Prefix list

The following configuration information created on the L2-device is available to RA-Guard to validate against the information found in the received RA frame:

- Allowed/Disallowed ports for receiving RA-guard messages
- Allowed/Disallowed IP source addresses of RA-sender
- Allowed Prefix list and Prefix ranges
- Control Router Preference

RA guard occurs at the controller. You can configure the controller to drop RA messages at the access point or at the controller. All IPv6 RA messages are dropped, which protects other wireless clients and upstream wired network from malicious IPv6 clients.

RA Throttling

RA throttling allows the controller to enforce limits to RA packets headed toward the wireless network. By enabling RA throttling, routers that send many RA packets can be trimmed to a minimum frequency that will still maintain an IPv6 client connectivity. If a client sends an RS packet, an RA is sent back to the client. This RA is allowed through the controller and unicasted to the client. This process ensures that the new clients or roaming clients are not affected by the RA throttling.

Neighbor Discovery

IPv6 Neighbor Discovery is a set of messages and processes that determine relationships between neighboring nodes. Neighbor Discovery replaces ARP, ICMP Router Discovery, and ICMP Redirect used in IPv4.

IPv6 Neighbor Discovery inspection analyzes neighbor discovery messages in order to build a trusted binding table database, and IPv6 neighbor discovery packets that do not comply are dropped. The neighbor binding table in the controller tracks each IPv6 address and its associated MAC address. Clients are expired from the table according to Neighbor Binding timers.

Neighbor Solicitation

The IPv6 addresses of wireless clients are cached by the controller. If the controller receives an NS multicast looking for an IPv6 address, which belongs to any of the wireless clients of the controller, the controller acts as the proxy and replies with the NA.

Note The controller acts like proxy and respond with NA, only when the **ipv6 nd suppress** command is configured

If the controller does not have the IPv6 address of a wireless client, the controller will not respond with NA and forward the NS packet to the wireless side. To resolve this, an NS Multicast Forwarding knob is provided. If this knob is enabled, the controller gets the NS packet for the IPv6 address that it does not have (cache miss) and forwards it to the wireless side. This packet reaches the intended wireless client and the client replies with NA.

This cache miss scenario occurs rarely, and only very few clients which do not implement complete IPv6 stack may not advertise their IPv6 address during NDP.

How To Configure IPv6 Client Address Learning

Configuring IPv6 on Controller

Use this configuration example to configure IPv6 on an interface.

Before You Begin

Enable IPv6 on the client and IPv6 support on the wired infrastructure.

SUMMARY STEPS

1. **interface vlan** 1
2. **ip address** fe80::1 link-local
3. **ipv6 enable**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**interface vlan** 1 **Example:** `Controller(config)# interface vlan 1`	Creates a interface and enters interface configuration mode.

Step 2	**ip address** fe80::1 link-local
	
	Example:
	`Controller(config-if)# ip address 198.51.100.1 255.255.255.0`
	`Controller(config-if)# ipv6 address fe80::1 link-local`
	
	`Controller(config-if)# ipv6 address 2001:DB8:0:1:FFFF:1234::5/64`
	`Controller(config-if)# ipv6 address 2001:DB8:0:0:E000::F/64`
		Configures IPv6 address on the interface using the link-local option.

Step 3	**ipv6 enable** **Example:** `Controller(config)# ipv6 enable`	(Optional) Enables IPv6 on the interface.
Step 4	**end** **Example:** `Controller(config)# end`	Exits from the interface mode.

Configuring DHCP Pool

SUMMARY STEPS

1. **ipv6 dhcp pool** Vlan21
2. **address prefix** 2001:DB8:0:1:FFFF:1234::/64 **lifetime** 300 10
3. **dns-server** 2001:100:0:1::1
4. **domain-name** example.com
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**ipv6 dhcp pool** Vlan21 **Example:** `Controller(config)# ipv6 dhcp pool vlan1`	Enters the configuration mode and configures the IPv6 DHCP pool on the Vlan.
Step 2	**address prefix** 2001:DB8:0:1:FFFF:1234::/64 **lifetime** 300 10 **Example:** `Controller(config-dhcpv6)# address prefix 2001:DB8:0:1:FFFF:1234::/64 lifetime 300 10`	Enters the configuration-dhcp mode and configures the address pool and its lifetime on a Vlan.
Step 3	**dns-server** 2001:100:0:1::1 **Example:** `Controller(config-dhcpv6)# dns-server 2001:20:21::1`	Configures the DNS servers for the DHCP pool.
Step 4	**domain-name** example.com **Example:** `Controller(config-dhcpv6)# domain-name example.com`	Configures the domain name to complete unqualified host names.
Step 5	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring Stateless Auto Address Configuration (without DHCP)

SUMMARY STEPS

1. **interface vlan** 1
2. **ip address** fe80::1 link-local
3. **ipv6 enable**
4. **no ipv6 nd managed-config-flag**
5. **no ipv6 nd other-config-flag**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**interface vlan** 1 **Example:** `Controller(config)# interface vlan 1`	Creates a interface and enters interface configuration mode.

Step 2	**ip address** fe80::1 link-local
	
	Example:
	`Controller(config-if)# ip address 198.51.100.1 255.255.255.0`
	`Controller(config-if)# ipv6 address fe80::1 link-local`
	`Controller(config-if)# ipv6 address 2001:DB8:0:1:FFFF:1234::5/64`
	`Controller(config-if)# ipv6 address 2001:DB8:0:0:E000::F/64`
		Configures IPv6 address on the interface using the link-local option.

Step 3	**ipv6 enable** **Example:** `Controller(config)# ipv6 enable`	(Optional) Enables IPv6 on the interface.

Step 4	**no ipv6 nd managed-config-flag**
	
	Example:
	`Controller(config)#interface vlan 1`
	`Controller(config-if)# no ipv6 nd managed-config-flag`
		Ensures the attached hosts do not use stateful autoconfiguration to obtain addresses.

Step 5	**no ipv6 nd other-config-flag** **Example:** `Controller(config-if)# no ipv6 nd other-config-flag`	Ensures the attached hosts do not use stateful autoconfiguration to obtain non-address options from DHCP (domain etc).
Step 6	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Configuring Stateless Auto Address Configuration (with DHCP)

SUMMARY STEPS

1. **interface vlan** 1
2. **ip address** fe80::1 link-local
3. **ipv6 enable**
4. **no ipv6 nd managed-config-flag**
5. **ipv6 nd other-config-flag**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**interface vlan** 1 **Example:** `Controller(config)# interface vlan 1`	Creates a interface and enters interface configuration mode.

Step 2	**ip address** fe80::1 link-local
	
	Example:
	`Controller(config-if)# ip address 198.51.100.1 255.255.255.0`
	`Controller(config-if)# ipv6 address fe80::1 link-local`
	`Controller(config-if)# ipv6 address 2001:DB8:0:1:FFFF:1234::5/64`
	`Controller(config-if)# ipv6 address 2001:DB8:0:0:E000::F/64`
		Configures IPv6 address on the interface using the link-local option.

Step 3	**ipv6 enable** **Example:** `Controller(config)# ipv6 enable`	(Optional) Enables IPv6 on the interface.

Step 4	**no ipv6 nd managed-config-flag**
	
	Example:
	`Controller(config)#interface vlan 1`
	`Controller(config-if)# no ipv6 nd managed-config-flag`
		Ensures the attached hosts do not use stateful autoconfiguration to obtain addresses.

Step 5	**ipv6 nd other-config-flag** **Example:** `Controller(config-if)# no ipv6 nd other-config-flag`	Ensures the attached hosts do not use stateful autoconfiguration to obtain non-address options from DHCP (domain etc).
Step 6	**end** **Example:** `Controller(config)# end`	Exits from the interface mode.

Configuring Stateful DHCP

SUMMARY STEPS

1. **configure terminal**
2. **ipv6 dhcp pool** IPv6_DHCPPOOL
3. **address prefix** 2001:DB8:0:1:FFFF:1234::/64
4. **dns-server** 2001:100:0:1::1
5. **domain-name** example.com
6. **exit**
7. **interface** vlan1
8. **ipv6 nd prefix** 2001:db8::/64 **no-advertise**
9. **ipv6 nd managed-config-flag**
10. **ipv6 nd other-config-flag**
11. **ipv6 dhcp server** IPv6_DHCPPOOL

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ipv6 dhcp pool** IPv6_DHCPPOOL **Example:** `Controller (config)# ipv6 dhcp pool IPv6_DHCPPOOL`	Enters the configuration mode and configures the IPv6 DHCP pool on the VLAN.
Step 3	**address prefix** 2001:DB8:0:1:FFFF:1234::/64 **Example:** `Controller (config-dhcpv6)# address prefix 2001:DB8:0:1:FFFF:1234::/64`	Specifies the address range to provide in the pool.
Step 4	**dns-server** 2001:100:0:1::1 **Example:** `Controller (config-dhcpv6)# dns-server 2001:100:0:1::1`	Provides the DNS server option to DHCP clients.
Step 5	**domain-name** example.com **Example:** `Controller (config-dhcpv6)# domain-name example.com`	Provides the domain name option to DHCP clients.

	Command or Action	Purpose
Step 6	**exit** **Example:** `Controller (config-dhcpv6)# exit`	Returns to the previous mode.
Step 7	**interface** vlan1 **Example:** `Controller (config)# interface vlan 1`	Enters the interface mode to configure the stateful DHCP.
Step 8	**ipv6 nd prefix** 2001:db8::/64 **no-advertise** **Example:** `Controller (config-if)# ipv6 nd prefix 2001:db8::/64 no-advertise`	Configures the IPv6 routing prefix advertisement that must not be advertised.
Step 9	**ipv6 nd managed-config-flag** **Example:** `Controller (config-if)# ipv6 nd managed-config-flag`	Configures IPv6 interfaces neighbor discovery to allow the hosts to uses DHCP for address configuration.
Step 10	**ipv6 nd other-config-flag** **Example:** `Controller (config-if)# ipv6 nd other-config-flag`	Configures IPv6 interfaces neighbor discovery to allow the hosts to uses DHCP for non-address configuration.
Step 11	**ipv6 dhcp server** IPv6_DHCPPOOL **Example:** `Controller (config-if)# ipv6 dhcp server IPv6_DHCPPOOL`	Configures the DHCP server on the interface.

Verifying IPv6 Client Address Learning

Verifying IPv6 Address Learning Configuration

This example displays the output of the **show ipv6 dhcp pool** command. This command displays the IPv6 service configuration on the controller. The vlan 21 configured pool detail displays 6 clients that are currently using addresses from the pool.

SUMMARY STEPS

1. **show ipv6 dhcp pool**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**show ipv6 dhcp pool**
	
	Example:
	`Controllershow ipv6 dhcp pool`
	`DHCPv6 pool: vlan21`
	`Address allocation prefix: 2001:DB8:0:1:FFFF:1234::/64 valid 86400 preferred 86400 (6 in use, 0 conflicts)`
	`DNS server: 2001:100:0:1::1`
	`Domain name: example.com`
	`Active clients: 6`
		Displays the IPv6 service configuration on the controller.

Debugging IPv6 Address Learning

Use the following CLI command to debug IPv6 in the controller.

DETAILED STEPS

	Command or Action	Purpose

Step 1	**debug wcdb ipv6**
	
	Example:
	`Controller# debug wcdb ipv6`
	`IPv6 address all debugging is on`
	`.......`
	`.......`
	`.......`
	`*Jan 3 20:06:38.096: %IOSXE-7-PLATFORM: 1 process wcm: 8853.2EDC.68EC apChanged 0 mscb ipAddr 10.10.19.121, apf RadiusOverride 0x0, numIPv6Addr=2001:db8::25:6`
	`.......`
	`.......`
	`.......`
	`*Jan 3 20:06:38.096: %IOSXE-7-PLATFORM: 1 process wcm: 8853.2EDC.68EC Applying site-specific IPv6 override for station 8853.2EDC.68EC - vapId 6, site 'default-group', interface 'VLAN0019'`
	`.......`
	`.......`
	`.......`
		This command is used to debug the controller related IPv6 details, errors, events and packets.

Monitoring Client Address Learning

Viewing Interfaces Configured for IPv6 Address Learning

Use this command to see the interfaces configured for IPv6 address learning.

SUMMARY STEPS

1. **show ipv6 dhcp interface**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**show ipv6 dhcp interface**
	
	Example:
	`Controller (config)# show ipv6 dhcp interface`
	`Vlan21 is in server mode`
	`Using pool: vlan21`
	`Preference value: 0`
	`Hint from client: ignored`
	`Rapid-Commit: disabled`	

Viewing IPv6 Address Learning

Use this command to monitor the IPv6 address learning and neighbor discovery.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show ipv6 dhcp binding** **Example:** `Controller# show ipv6 dhcp binding`	Displays the IPv6 dhcp binding details.
Step 2	**show ipv6 dhcp relay binding** **Example:** `Controller# show ipv6 dhcp relay binding`	Displays the IPv6 dhcp relay binding details.

```
Controller (config)# show ipv6 dhcp binding
Client: FE80::99BC:7B03:D2FB:C301 (Vlan19)
DUID: 000100011864E60700188BBF6407
IA NA: IA ID 0x0F88532E, T1 43200, T2 69120
Address: FC00:19:1:0:48DB:C050:B209:83EC
preferred lifetime 86400, valid lifetime 172800
expires at Jan 06 2013 01:37 PM (172747 seconds)


Controller (config)# show ipv6 dhcp relay binding
Summary:
Total number of Relay bindings = 0
Total number of Relay bindings added by Bulk lease = 0
```

Viewing RA Throttling and NS Suppression

Use this command to see the iRA throttling and NS suppression details.

SUMMARY STEPS

1. **show ipv6 nd raguard** Mypolicy
2. **show ipv6 nd ra-throttler policy** Mythrottle

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show ipv6 nd raguard** Mypolicy **Example:** `Controller (config)# show ipv6 nd raguard Mypolicy`	Shows the IPv6 neighbor discovery RA details.
Step 2	**show ipv6 nd ra-throttler policy** Mythrottle **Example:** `Controller (config)# show ipv6 nd ra-throttler policy Mythrottler`	Shows the IPv6 neighbor RA throttle details.

```
Controller (config)# show ipv6 nd raguard Mypolicy
Policy Mypolicy configuration:
trusted-port
device-role router
Policy Mypolicy is applied on the following targets:
Target               Type  Policy              Feature        Target range
Te1/0/3              PORT  Mypolicy            RA guard       vlan all
vlan 19              VLAN  Mypolicy            RA guard       vlan all
vlan 20              VLAN  Mypolicy            RA guard       vlan all
vlan 21              VLAN  Mypolicy            RA guard       vlan all
vlan 23              VLAN  Mypolicy            RA guard       vlan all


Controller (config)# show ipv6 nd ra-throttler policy Mythrottler
Policy Mythrottle configuration:
The throttle period is set to 20 seconds
Capped at no more than 5 unthrottled RAs per throttle period
The policy allows at least 3 and at most 5 RAs per router
The behaviour upon RAs with an RFC 3775 interval option is inherited and defaults to
passthrough
Policy Mythrottle is applied on the following targets:
Target               Type  Policy              Feature        Target range
vlan 19              VLAN  Mythrottle          RA throttler   vlan all
vlan 20              VLAN  Mythrottle          RA throttler   vlan all
vlan 21              VLAN  Mythrottle          RA throttler   vlan all
vlan 23              VLAN  Mythrottle          RA throttler   vlan all
```

Configuration Example for IPv6 Client Address Learning

Creating a DHCP Scope

This example configures an IPv6 DHCP scope named vlan21. This DHCP scope provides IPv6 addresses in the 2001:DB8:0:1:FFFF:1234::/64 subnet. The validity and preferred validity of each IPv6 address provided is set to one day (86400 seconds, or 24 hours). The DHCP scope also provide a dns-server IPv6 address option (2001:100:0:1::1), and the domain name for the scope.

SUMMARY STEPS

1. **configure terminal**
2. **ipv6 dhcp pool vlan21**
3. **address prefix** 2001:DB8:0:1:FFFF:1234::/64 **lifetime** 86400 86400
4. **dns-server**2001:100:0:1::1
5. **domain-name** example.com
6. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ipv6 dhcp pool vlan21** **Example:** `Controller (config)# ipv6 dhcp pool Vlan21`	Enters the configuration mode and configures the IPv6 DHCP pool on the VLAN.
Step 3	**address prefix** 2001:DB8:0:1:FFFF:1234::/64 **lifetime** 86400 86400 **Example:** `Controller (config-dhcp)# address prefix 2001:DB8:0:1:FFFF:1234::/64 lifetime 86400 86400`	Enters the configuration-dhcp mode and configures the address pool and its lifetime on a VLAN.
Step 4	**dns-server**2001:100:0:1::1 **Example:** `Controller (config-dhcp)# dns-server 2001:100:0:1::1`	Configures the DNS servers for the DHCP pool.
Step 5	**domain-name** example.com **Example:** `Controller (config-dhcp)# domain-name example.com`	Configures the domain name to complete unqualified host names.

	Command or Action	Purpose
Step 6	**exit** **Example:** `Controller (config-dhcp)# exit`	Returns to the previous mode.

Enabling IPv6 on a Interface and Providing IPv6 Addresses to DHCP Clients

This example configures IPv6 on interface VLAN 21. The first address (fe80::1) is a link local address, the second address (2001:db8:0:1:ffff:1234::1/64) is a global unicast address. IPv6 DHCP clients requesting for address in VLAN 21 are directed to the DHCP pool vlan21. The controller informs IPv6 hosts in VLAN 21 that the DHCP scope provided supports stateful auto-configuration (nd managed-config-flag), and can also provide non-address options, such as domain name or dns server (nd other-config-flag).

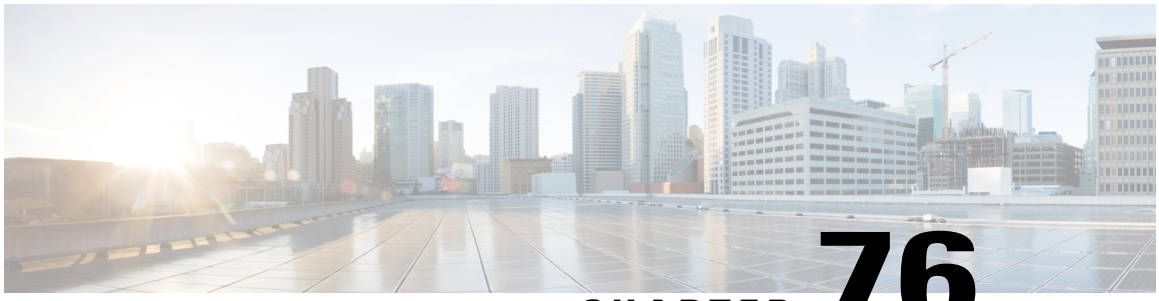
SUMMARY STEPS

1. **interface vlan** 21
2. **ipv6 address** fe80::1 **link-local**
3. **ipv6 address** 2001:db8:0:1:ffff:1234::1/64
4. **ipv6 dhcp server vlan** 21
5. **ipv6 nd managed-config-flag**
6. **ipv6 nd other-config-flag**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**interface vlan** 21 **Example:** `Controller(config)# interface vlan 21`	Enters the configuration mode and configures the IPv6 DHCP pool on the VLAN 21.
Step 2	**ipv6 address** fe80::1 **link-local** **Example:** `Controller (config-if)# ip address fe80::1 link-local`	Configures the IPv6 link local address on VLAN 21.
Step 3	**ipv6 address** 2001:db8:0:1:ffff:1234::1/64 **Example:** `Controller (config-if)# ipv6 address 2001:db8:0:1:ffff:1234::1/64`	Configures the IPv6 global unicast address on VLAN 21.

	Command or Action	Purpose
Step 4	**ipv6 dhcp server vlan** 21 **Example:** `Controller(config-if)# ipv6 dhcp server vlan21`	Directs the IPv6 clients requesting for an address in VLAN 21 to the DHCP pool vlan21.
Step 5	**ipv6 nd managed-config-flag** **Example:** `Controller (config-if)# ipv6 nd managed-config-flag`	Informs the IPv6 hosts in VLAN 21 about the DHCP scope supports stateful auto-configuration.
Step 6	**ipv6 nd other-config-flag** **Example:** `Controller(config-if)# ipv6 nd other-config-flag`	Provides for non-address options, such as domain name or dns server.



CHAPTER 76

Configuring IPv6 WLAN Security

Prerequisites for IPv6 WLAN Security

A client VLAN must be mapped to the WLAN configured on the controller

Restrictions for IPv6 WLAN Security

RADIUS Server Support

- If multiple RADIUS servers are configured for redundancy, the user database must be identical in all the servers for the backup to work properly.

Radius ACS Support

- You must configure RADIUS on both your Cisco Secure Access Control Server (ACS) and your controller
- RADIUS is supported on Cisco Secure ACS version 3.2 and later releases.

Information About IPv6 WLAN Security

Information About RADIUS

Remote Authentication Dial-In User Service (RADIUS) is a client/server protocol that provides centralized security for users attempting to gain management access to a network. It serves as a back-end database similar to Local EAP and provides authentication and accounting services.

- Authentication—The process of verifying users when they attempt to log into the controller

  Users must enter a valid username and password for the controller to authenticate users to the RADIUS server. If multiple databases are configured, then specify the sequence in which the backend database must be tried.

- Accounting— The process of recording user actions and changes.

  Whenever a user successfully executes an action, the RADIUS accounting server logs the changed attributes, the user ID of the person who made the change, the remote host where the user is logged in, the date and time when the command was executed, the authorization level of the user, and a description of the action performed and the values provided. If the RADIUS accounting server is unreachable, the users can continue their sessions uninterrupted.

User Datagram Protocol— RADIUS uses User Datagram Protocol (UDP) for its transport. It maintains a database and listens on UDP port 1812 for incoming authentication requests and UDP port 1813 for incoming accounting requests. The controller, which requires access control, acts as the client and requests AAA services from the server. The traffic between the controller and the server is encrypted by an algorithm defined in the protocol and a shared secret key configured on both devices.

Configures multiple RADIUS accounting and authentication servers. For example, you can have one central RADIUS authentication server but several RADIUS accounting servers in different regions. If you configure multiple servers of the same type and the first one fails or becomes unreachable, the controller automatically tries the second one, then the third one if necessary, and so on.

When RADIUS method is configured for the WLAN, the controller will use the RADIUS method configured for the WLAN. When the WLAN is configured to use local EAP, the RADIUS method configured on the WLAN points to Local. The WLAN must also be configured with the name of the local EAP profile to use.

If no RADIUS method is configured in the WLAN, the controller will use the default RADIUS method defined in global mode.

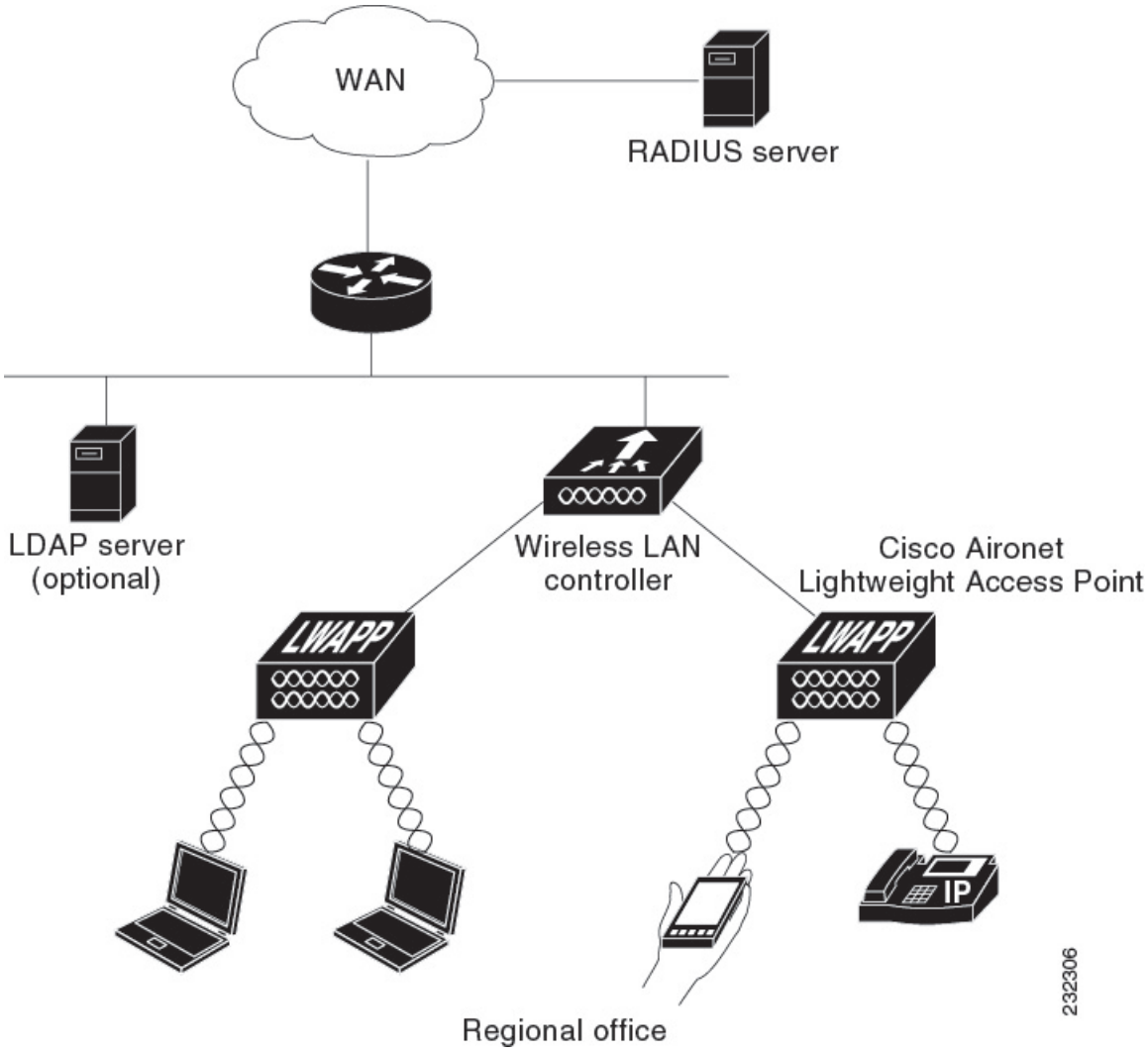
Information About Local EAP

Local EAP is an authentication method that allows users and wireless clients to be authenticated locally. It is designed for use in remote offices that maintain connectivity to wireless clients when the back-end system is disrupted or the external authentication server goes down. When you enable local EAP, the controller serves as the authentication server and the local user database, which removes dependence on an external authentication server. Local EAP retrieves user credentials from the local user database or the LDAP back-end database to authenticate users. Local EAP supports LEAP, EAP-FAST, EAP-TLS, PEAPv0/MSCHAPv2, and PEAPv1/GTC authentication between the controller and wireless clients.

Note The LDAP back-end database supports these local EAP methods: EAP-TLS, EAP-FAST/GTC, and PEAPv1/GTC. LEAP, EAP-FAST/MSCHAPv2, and PEAPv0. MSCHAPv2 is supported only if the LDAP server is set up to return a clear-text password.

Note Controller support Local EAP authentication against external LDAP databases such as Microsoft Active Directory and Novell's eDirectory. For more information about configuring the controller for Local EAP authentication against Novell's eDirectory, see the Configure Unified Wireless Network for Authentication Against Novell's eDirectory Database whitepaper.

Figure 56: Local EAP Example



How to Configure IPv6 WLAN Security

Configuring Local Authentication

Creating a Local User

SUMMARY STEPS

1. **configure terminal**
2. **username** aaa_test
3. **password 0** aaa_test
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global command mode.
Step 2	**username** aaa_test **Example:** `Controller(config)# username aaa_test`	Creates a username.
Step 3	**password 0** aaa_test **Example:** `Controller(config)# username aaa_test password 0 aaa_test`	Assigns a password for the username.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit the global configuration mode.

```
Controller# configure terminal
Controller(config)# username aaa_test password 0 aaa_test
Controller(config)# end
```

Creating an Client VLAN and Interface

SUMMARY STEPS

1. **configure terminal**
2. **vlan**
3. **exit**
4. **interface vlan** vlan_ID
5. **ip address**
6. **ipv6 address**
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global command mode.
Step 2	**vlan** **Example:** `Controller(config)# vlan 137`	Creates a VLAN.
Step 3	**exit** **Example:** `Controller (config-vlan)# exit`	Exits VLAN configuration mode.
Step 4	**interface vlan** vlan_ID **Example:** `Controller (config)# interface vlan 137`	Associates the VLAN to an interface.
Step 5	**ip address** **Example:** `Controller(config-if)# ip address 10.7.137.10 255.255.255.0`	Assigns an IP address to the VLAN interface.
Step 6	**ipv6 address** **Example:** `Controller(config-if)#ipv6 address 2001:db8::20:1/64`	Assigns an IPv6 address to the VLAN interface.
Step 7	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit the global configuration mode.

```
Controller# configure terminal
Controller(config)# vlan 137
Controller(config-vlan)#exit
Controller(config)#interface vlan 137
Controller(config-if)#ip address 10.7.137.10 255.255.255.0
Controller(config-if)#ipv6 address 2001:db8::20:1/64
Controller(config-if)#end
```

Configuring a EAP Profile

SUMMARY STEPS

1. **eap profile** name
2. **method leap**
3. **method tls**
4. **method peap**
5. **method mschapv2**
6. **method md5**
7. **method gtc**
8. **method fast profile** my-fast
9. **description** my_local**eap profile**
10. **exit**
11. **eap method fast profile**myFast
12. **authority-id** [identity|information]
13. **local-key 0** key-name
14. **pac-password 0** password
15. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**eap profile** name **Example:** `Controller(config)# eap profile wcm_eap_prof`	Creates a EAP profile.
Step 2	**method leap** **Example:** `Controller(config-eap-profile)# method leap`	Configures EAP-LEAP method on the profile.
Step 3	**method tls** **Example:** `Controller(config-eap-profile)# method tls`	Configures EAP-TLS method on the profile.

	Command or Action	Purpose
Step 4	**method peap** **Example:** `Controller(config-eap-profile)# method peap`	Configures PEAP method on the profile.
Step 5	**method mschapv2** **Example:** `Controller(config-eap-profile)# method mschapv2`	Configures EAP-MSCHAPV2 method on the profile.
Step 6	**method md5** **Example:** `Controller(config-eap-profile)# method md5`	Configures EAP-MD5 method on the profile.
Step 7	**method gtc** **Example:** `Controller(config-eap-profile)# method gtc`	Configures EAP-GTC method on the profile.

Step 8	**method fast profile** my-fast
	
	Example:
	`Controller(config-eap-profile)# eap method fast profile my-fast`
	`Controller (config-eap-profile)#description my_local eap profile`
		Creates a EAP profile named my-fast.

Step 9	**description** my_local**eap profile** **Example:** `Controller (config-eap-profile)#description my_local eap profile`	Provides a description for the local profile.
Step 10	**exit** **Example:** `Controller (config-eap-profile)# exit`	Exits the eap-profile configuration mode.
Step 11	**eap method fast profile**myFast **Example:** `Controller (config)# eap method fast profile myFast`	Configures the EAP method profile.

Step 12	**authority-id** [identity|information]
	
	Example:
	`Controller(config-eap-method-profile)# authority-id identity my_identity`
	`Controller(config-eap-method-profile)#authority-id information my_information`
		Configure the authority ID and information for the EAP method profile.

	Command or Action	Purpose
Step 13	**local-key 0** key-name **Example:** `Controller(config-eap-method-profile)# local-key 0 test`	Configures the local server key.
Step 14	**pac-password 0** password **Example:** `Controller(config-eap-method-profile)# pac-password 0 test`	Configures the PAC password for manual PAC provisioning.
Step 15	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit the global configuration mode.

```
Controller(config)#eap profile wcm_eap_prof
Controller(config-eap-profile)#method leap
Controller(config-eap-profile)#method tls
Controller(config-eap-profile)#method peap
Controller(config-eap-profile)#method mschapv2
Controller(config-eap-profile)#method md5
Controller(config-eap-profile)#method gtc
Controller(config-eap-profile)#eap method fast profile my-fast
Controller (config-eap-profile)#description my_local eap profile
Controller(config-eap-profile)# exit
Controller (config)# eap method fast profile myFast
Controller(config-eap-method-profile)#authority-id identity my_identity
Controller(config-eap-method-profile)#authority-id information my_information
Controller(config-eap-method-profile)#local-key 0 test
Controller(config-eap-method-profile)#pac-password 0 test
Controller(config-eap-method-profile)# end
```

Creating a Local Authentication Model

SUMMARY STEPS

1. **aaa** new-model
2. **authentication dot1x default local**
3. **dot1x** method_list **local**
4. **aaa authentication dot1x** dot1x_name **local**
5. **aaa authorization credential-download** name **local**
6. **aaa local authentication** auth-name **authorization** authorization-name
7. **session ID**
8. **dot1x system-auth-control**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**aaa** new-model **Example:** `Controller(config)# aaa new-model`	Creates a AAA authentication model.
Step 2	**authentication dot1x default local** **Example:** `Controller(config)# aaa authentication dot1x default local`	Implies that the dot1x must use the default local RADIUS when no other method is found.
Step 3	**dot1x** method_list **local** **Example:** `Controller(config)# aaa authentication dot1x wcm_local local`	Assigns the local authentication for wcm_local method list.
Step 4	**aaa authentication dot1x** dot1x_name **local** **Example:** `Controller(config)# aaa authentication dot1x aaa_auth local`	Configures the local authentication for the dot1x method.
Step 5	**aaa authorization credential-download** name **local** **Example:** `Controller(config)# aaa authorization credential-download wcm_author local`	Configures local database to download EAP credentials from Local/RADIUS/LDAP.
Step 6	**aaa local authentication** auth-name **authorization** authorization-name **Example:** `Controller(config)# aaa local authentication wcm_local authorization wcm_author`	Selects local authentication and authorization.
Step 7	**session ID** **Example:** `Controller(config)# aaa session-id common`	Configures a session ID for AAA.
Step 8	**dot1x system-auth-control** **Example:** `Controller(config)# dot1x system-auth-control`	Enables dot.1x system authentication control.

```
Controller(config)# aaa new-model
Controller(config)# aaa authentication dot1x default local
Controller(config)# aaa authentication dot1x wcm-local local
Controller(config)# aaa authentication dot1x aaa_auth local
Controller(config)# aaa authorization credential-download wcm_author local
Controller(config)# aaa local authentication wcm_local authorization wcm_author
```

```
Controller(config)# aaa session-id common
Controller(config)# dot1x system-auth-control
```

Creating a Client WLAN

Note This example uses 802.1x with dynamic WEP. You can use any other security mechanism supported by the wireless client and configurable on the controller

SUMMARY STEPS

1. **configure terminal**
2. **wlan** wlan name **<identifier>** SSID
3. **broadcast-ssid**
4. **no security wpa**
5. **security dot1x**
6. **security** dot1x authentication-list *wcm-local*
7. **local-auth** wcm_eap_prof
8. **client vlan** *137*
9. **no shutdown**
10. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global command mode.
Step 2	**wlan** wlan name **<identifier>** SSID **Example:** `Controller(config)# wlan wlanProfileName 1 ngwcSSID`	Creates a WLAN.
Step 3	**broadcast-ssid** **Example:** `Controller(config-wlan)# broadcast-ssid`	Configures to broadcast the SSID on a WLAN.
Step 4	**no security wpa** **Example:** `Controller(config-wlan)# no security wpa`	Disables the wpa for WLAN to enable 802.1x.

	Command or Action	Purpose
Step 5	**security dot1x** **Example:** `Controller(config-wlan)# security dot1x`	Configures the 802.1x encryption security for the WLAN.
Step 6	**security** dot1x authentication-list *wcm-local* **Example:** `Controller(config-wlan)# security dot1x authentication-list wcm-local`	Configures the server group mapping to the WLAN for dot1x authentication.
Step 7	**local-auth** wcm_eap_prof **Example:** `Controller (config-wlan)# local-auth wcm_eap_profile`	Configures the eap profile on the WLAN for local authentication.
Step 8	**client vlan** *137* **Example:** `Controller(config-wlan)# client vlan 137`	Associates the VLAN to a WLAN.
Step 9	**no shutdown** **Example:** `Controller(config-wlan)# no shutdown`	Enables the WLAN.
Step 10	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit the global configuration mode.

```
Controller# config terminal
Controller(config)#wlan wlanProfileName 1 ngwcSSID
Controller(config-wlan)#broadcast-ssid
Controller(config-wlan)#no security wpa
Controller(config-wlan)#security dot1x
Controller(config-wlan)#security dot1x authentication-list wcm-local
Controller (config-wlan)# local-auth wcm_eap_prof
Controller(config-wlan)#client vlan 137
Controller(config-wlan)#no shutdown
Controller(config-wlan)#end
Controller#
```

Configuring Local Authentication with WPA2+AES

SUMMARY STEPS

1. **configure terminal**
2. **aaa** new model
3. **dot1x system-auth-control**
4. **aaa authentication dot1x default local**
5. **aaa local authorization credential-download default local**
6. **aaa local authentication default authorization default**
7. **eap profile** wcm_eap_profile
8. **method leap**
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global command mode.
Step 2	**aaa** new model **Example:** `Controller(config)# aaa new-model`	Creates a AAA authentication model.
Step 3	**dot1x system-auth-control** **Example:** `Controller(config)# dot1x system-auth-control`	Enables dot1x system authentication control.
Step 4	**aaa authentication dot1x default local** **Example:** `Controller(config)# aaa authentication dot1x default local`	Configures the local authentication for the default dot1x method.
Step 5	**aaa local authorization credential-download default local** **Example:** `Controller(config)# aaa authorization credential-download default local`	Configures default database to download EAP credentials from local server.
Step 6	**aaa local authentication default authorization default** **Example:** `Controller(config)# aaa local authentication default authorization default`	Selects the default local authentication and authorization.

	Command or Action	Purpose
Step 7	**eap profile** wcm_eap_profile **Example:** `Controller(config)#eap profile wcm_eap_profile`	Creates an EAP profile.
Step 8	**method leap** **Example:** `Controller(config)# method leap`	Configures EAP-LEAP method on the profile.
Step 9	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit the global configuration mode.

```
Controller# configure terminal
Controller(config)# aaa new-model
Controller(config)# dot1x system-auth-control
Controller(config)# aaa authentication dot1x default local
Controller(config)# aaa authorization credential-download default local
Controller(config)# aaa local authentication default authorization default
Controller(config)#eap profile wcm_eap_profile
Controller(config)# method leap
Controller(config)# end
```

Creating Client VLAN for WPA2+AES

Create a VLAN for the WPA2+AES type of local authentication. This VLAN is later mapped to a WLAN.

SUMMARY STEPS

1. **configure terminal**
2. **vlan** vlan_ID
3. **exit**
4. **interface** vlan vlan_ID
5. **ip address**
6. **ipv6 address**
7. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global command mode.

	Command or Action	Purpose
Step 2	**vlan** vlan_ID **Example:** `Controller (config)# vlan 105`	Creates a VLAN.
Step 3	**exit** **Example:** `Controller (config-vlan)# exit`	Exits from the VLAN mode.
Step 4	**interface** vlan vlan_ID **Example:** `Controller(config)# interface vlan 105`	Associates the VLAN to the interface.
Step 5	**ip address** **Example:** `Controller(config-if)# ip address 10.8.105.10 255.255.255.0`	Assigns IP address to the VLAN interface.
Step 6	**ipv6 address** **Example:** `Controller(config-if)#ipv6 address 2001:db8::10:1/64`	Assigns IPv6 address to the VLAN interface.
Step 7	**exit** **Example:** `Controller (config-if)# exit`	Exits from the interface mode.

```
Controller# configure terminal
Controller(config)# vlan105
Controller (config-vlan)# exit
Controller (config)# interface vlan 105
Controller(config-if)#ip address 10.8.105.10 255.255.255.0
Controller(config-if)#ipv6 address 2001:db8::10:1/64
Controller(config-if)#exit
Controller(config)#
```

Creating WLAN for WPA2+AES

Create a WLAN and map it to the client VLAN created for WPA2+AES.

SUMMARY STEPS

1. **configure terminal**
2. **wlan** wpas2-aes-wlan 1 wpas2-aes-wlan
3. **client vlan** 105
4. **local-auth** wcm_eap_profile
5. **security dot1x authentication-list default**
6. **no shutdown**
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global command mode.

Step 2	**wlan** wpas2-aes-wlan 1 wpas2-aes-wlan
	
	Example:
	`Controller(config)#wlan wpa2-aes-wlan 1 wpa2-aes-wlan`
	`Controller(config-wlan)#`
		Creates a WLAN.

Step 3	**client vlan** 105
	
	Example:
	`Controller(config-wlan)#client vlan 105`
	`Controller(config-wlan)#`
		Maps the WLAN to the client VLAN.

Step 4	**local-auth** wcm_eap_profile **Example:** `Controller(config-wlan)#local-auth wcm_eap_profile`	Creates and sets the EAP profile on the WLAN.

Step 5	**security dot1x authentication-list default**
	
	Example:
	`Controller(config-wlan)#security dot1x`
	`authentication-list default`
		Uses the default dot1x authentication list.

Step 6	**no shutdown**
	
	Example:
	`Controller(config-wlan)#no shutdown`
	`Controller(config-wlan)#`
		Enables the WLAN.

Step 7	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

```
Controller# configure terminal
Controller(config)#wlan wpa2-aes-wlan 1 wpa2-aes-wlan
Controller(config-wlan)#client vlan 105
Controller(config-wlan)#local-auth wcm_eap_profile
Controller(config-wlan)#security dot1x authentication-list default
Controller(config-wlan)#no shutdown
Controller(config-wlan)# exit
```

Configuring External RADIUS Server

Configuring RADIUS Authentication Server Host

SUMMARY STEPS

1. **configure terminal**
2. **radius server One**
3. **address ipv4** address **auth-port**auth_port_number **acct-port** acct_port_number
4. **address ipv6** address **auth-port**auth_port_number **acct-port** acct_port_number
5. **key** 0cisco
6.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global command mode.
Step 2	**radius server One** **Example:** `Controller (config)# radius server One`	Creates a radius server.
Step 3	**address ipv4** address **auth-port**auth_port_number **acct-port** acct_port_number **Example:** `Controller (config-radius-server)# address ipv4 10.10.10.10 auth-port 1812 acct-port 1813`	Configures the IPv4 address for the radius server.
Step 4	**address ipv6** address **auth-port**auth_port_number **acct-port** acct_port_number **Example:** `Controller (config-radius-server)# address ipv6 2001:db8::25:2 auth-port 1812 acct-port 1813`	Configures the IPv6 address for the radius server.

	Command or Action	Purpose
Step 5	**key** 0cisco **Example:** `Controller (config-radius-server)# key 0 cisco`	**exit**
Step 6	 **Example:** `Controller (config-radius-server)# exit`	Exits from the radius server mode.

```
Controller# configure terminal
Controller (config)# radius server One
Controller (config-radius-server)# address ipv4 10.10.10.10 auth-port 1812 acct-port 1813
Controller (config-radius-server)# address ipv6 2001:db8::25:2 auth-port 1812 acct-port
1813
Controller (config-radius-server)# key 0 cisco
Controller (config-radius-server)#exit
```

Configuring RADIUS Authentication Server Group

SUMMARY STEPS

1. **configure terminal**
2. **aaa new-model**
3. **aaa group server radius** wcm_rad
4. **server** <ip address>**auth-port**1812**acct-port**1813
5. **aaa authentication dot1x** method_list **group** wcm_rad
6. **dot1x system-auth-control**
7. **aaa session-idcommon**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global command mode.
Step 2	**aaa new-model** **Example:** `Controller(config)#aaa new-model`	Creates a AAA authentication model.

	Command or Action	Purpose

Step 3	**aaa group server radius** wcm_rad
	
	Example:
	`Controller(config)# aaa group server radius wcm_rad`
	`Controller(config-sg-radius)#`
		Creates an radius server-group.

Step 4	**server** <ip address>**auth-port**1812**acct-port**1813
	
	Example:
	`Controller(config-sg-radius)# server One auth-port 1812 acct-port 1813`
	`Controller(config-sg-radius)# server Two auth-port 1812 acct-port 1813`
	`Controller(config-sg-radius)# server Three auth-port 1812 acct-port 1813`
		Adds servers to the radius group created in Step 3. Configures the UDP port for RADIUS accounting server and authentication server.

Step 5	**aaa authentication dot1x** method_list **group** wcm_rad **Example:** `Controller(config)# aaa authentication dot1x method_list group wcm_rad`	Maps the method list to the radius group.
Step 6	**dot1x system-auth-control** **Example:** `Controller(config)# dot1x system-auth-control`	Enables the system authorization control for the radius group.
Step 7	**aaa session-idcommon** **Example:** `Controller(config)# aaa session-id common`	Ensures that all session IDs information sent out, from the radius group, for a given call are identical.

```
Controller# configure terminal
Controller(config)# aaa new-model
Controller(config)# aaa group server radius wcm_rad
Controller(config-sg-radius)# server One auth-port 1812 acct-port 1813
Controller(config-sg-radius)# server Two auth-port 1812 acct-port 1813
Controller(config-sg-radius)# server Three auth-port 1812 acct-port 1813
Controller(config)# aaa authentication dot1x method_list group wcm_rad
Controller(config)# dot1x system-auth-control
Controller(config)# aaa session-id common
Controller(config)#
```

Creating a Client VLAN

SUMMARY STEPS

1. **configure terminal**
2. **vlan** 137
3. **exit**
4. **interface vlan** 137
5. **ip address** 10.7.137.10 255.255.255.0
6. **ipv6 address** 2001:db8::30:1/64
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global command mode.
Step 2	**vlan** 137 **Example:** `Controller(config)# vlan 137`	Creates a VLAN and associate it to the interface.
Step 3	**exit** **Example:** `Controller (config-vlan)# exit`	Exits from the VLAN mode.
Step 4	**interface vlan** 137 **Example:** `Controller (config)# interface vlan 137`	Assigns a VLAN to an interface.
Step 5	**ip address** 10.7.137.10 255.255.255.0 **Example:** `Controller(config-if)# ip address 10.7.137.10 255.255.255.0`	Assigns an IPv4 address to the VLAN interface.
Step 6	**ipv6 address** 2001:db8::30:1/64 **Example:** `Controller(config-if)# ipv6 address 2001:db8::30:1/64`	Assigns an IPv6 address to the VLAN interface.
Step 7	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

```
Controller# configure terminal
Controller(config)# vlan137
Controller(config-vlan)# exit
Controller(config)# interface vlan137
Controller(config-if)# ip address 10.7.137.10 255.255.255.0
Controller(config-if)# ipv6 address 2001:db8::30:1/64
Controller(config-if)# end
```

Creating 802.1x WLAN Using an External RADIUS Server

SUMMARY STEPS

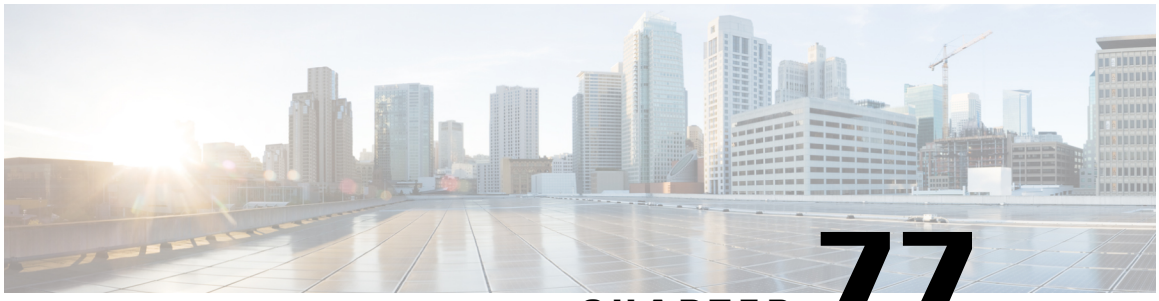
1. **configure terminal**
2. **wlan** ngwc-1x**<ssid>**ngwc-1x
3. **broadcast-ssid**
4. **no security wpa**
5. **security dot1x**
6. **security dot1x authentication-list** wcm-rad
7. **client vlan** 137
8. **no shutdown**
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global command mode.
Step 2	**wlan** ngwc-1x**<ssid>**ngwc-1x **Example:** `Controller(config)# wlan ngwc_8021x 2 ngwc_8021x`	Creates a new WLAN for 802.1x authentication.
Step 3	**broadcast-ssid** **Example:** `Controller(config-wlan)# broadcast-ssid`	Configures to broadcast the SSID on WLAN.
Step 4	**no security wpa** **Example:** `Controller(config-wlan)# no security wpa`	Disables the WPA for WLAN to enable 802.1x.

	Command or Action	Purpose
Step 5	**security dot1x** **Example:** `Controller(config-wlan)# security dot1x`	Configures the 802.1x encryption security for the WLAN.
Step 6	**security dot1x authentication-list** wcm-rad **Example:** `Controller(config-wlan)# security dot1x authentication-list wcm-rad`	Configures the server group mapping to the WLAN for dot1x authentication.
Step 7	**client vlan** 137 **Example:** `Controller(config-wlan)# client vlan 137`	Associates the VLAN to a WLAN.
Step 8	**no shutdown** **Example:** `Controller(config-wlan)# no shutdown`	Enables the WLAN.
Step 9	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit the global configuration mode.

```
Controller# configure terminal
Controller(config)#wlan ngwc_8021x 2 ngwc_8021x
Controller(config-wlan)# broadcast-ssid
Controller(config-wlan)# no security wpa
Controller(config-wlan)# security dot1x
Controller(config-wlan)# security dot1x authentication-list wcm-rad
Controller(config-wlan)# client vlan 137
Controller(config-wlan)# no shutdown
Controller(config-wlan)# end
```



CHAPTER 77

Configuring IPv6 ACL

Prerequisites for IPv6 ACL

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 feature set.

Restrictions for IPv6 ACL

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

The controller supports most of the Cisco IOS-supported IPv6 ACLs with some exceptions:

- The controller does not support routing and only inbound ACLs are supported for wireless clients.
- The controller does not support matching on these keywords: **flowlabel**, **routing header**, and **undetermined-transport**.
- The controller does not support reflexive ACLs (the **reflect** keyword).
- The controller does not apply MAC-based ACLs on IPv6 frames.
- 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 controller 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 controller does not allow the ACE to be added to the ACL that is currently attached to the interface

Information About IPv6 ACL

An access control list (ACL) is a set of rules used to limit access to a particular interface (for example, if you want to restrict a wireless client from pinging the management interface of the controller). ACLs are configured on the controllernd applied to the management interface, the AP-manager interface, any of the dynamic interfaces, or a WLAN to control data traffic to and from wireless clients or to the controller central processing unit (CPU) to control all traffic destined for the CPU.

You can also create a preauthentication ACL for web authentication. Such an ACL is used to allow certain types of traffic before authentication is complete.

IPv6 ACLs support the same options as IPv4 ACLs including source, destination, source and destination ports.

Note You can enable only IPv4 traffic in your network by blocking IPv6 traffic. That is, you can configure an IPv6 ACL to deny all IPv6 traffic and apply it on specific or all WLANs.

Understanding IPv6 ACLs

Controller considers the local ACL configured on the WLAN as its client ACL. It supports:

- Client ACL
- port ACL
- vlan ACL
- router ACL

The ACL's are evaluated in the above order. By default, when there is no ACL configured on the WLAN, an implicit permit-all is configured.

Note Port ACL is not applicable to the WLAN.

Types of ACL

Per User IPv6 ACL

For the per-user ACL, the full access control entries (ACE) as the text strings are configured on the ACS.

The ACE is not configured on the Controller. The ACE is sent to the controller in the `ACCESS-Accept` attribute and applies it directly for the client. When a wireless client roams into an foreign controller, the ACEs are sent to the foreign controller as an AAA attribute in the mobility Handoff message.

Filter ID IPv6 ACL

For the filter-Id ACL, the full ACEs and the `acl name(filter-id)` is configured on the controller and only the `filter-id` is configured on the ACS. The `filter-id` is sent to the controller in the ACCESS-Accept attribute, and the controller looks up the filter-id for the ACEs, and then applies the ACEs to the client. When the client L2 roams to the foreign controller, only the filter-id is sent to the foreign controller in the mobility Handoff message. The foreign controller has to configure the filter-id and ACEs beforehand.

Downloadable IPv6 ACL

For the downloadable ACL(dACL), the full ACEs and the `dacl` name are all configured on the ACS only.

Note The controller does not configure any ACL.

The ACS sends the `dacl` name to the controller in its `ACCESS-Accept` attribute, which takes the `dacl` name and sends the `dACL` name back to the ACS, for the ACEs, using the `access-request` attribute.

The ACS responds to the corresponding ACEs of the controller in the `access-accept` attribute. When the wireless client roams to an foreign controller, only the `dacl` name is sent to the foreign controller in the mobility Handoff message. The foreign controller contacts the ACS server with the `dacl` name to retrieve the ACEs.

Configuring IPv6 ACLs

To filter IPv6 traffic, you perform these steps:

SUMMARY STEPS

1. Create an IPv6 ACL, and enter IPv6 access list configuration mode.
2. Configure the IPv6 ACL to block (deny) or pass (permit) traffic.
3. Apply the IPv6 ACL to the interface where the traffic needs to be filtered.

DETAILED STEPS

	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 the interface where the traffic needs to be filtered.	

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 forwarded to the CPU, and the ACLs are applied in software.

How To Configure an IPv6 ACL

Creating IPv6 ACL

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

SUMMARY STEPS

1. **configure terminal**
2. **ipv6access-list***acl_name*
3. **{deny|permit} protocol**
4. **{deny|permit} tcp**
5. **{deny|permit} udp**
6. **{deny|permit} icmp**
7. **end**
8. **show ipv6 access-list**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**ipv6access-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 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.

	Command or Action	Purpose
		• (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} [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 3, 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 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]`	(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.

	Command or Action	Purpose
	`[neq {port |protocol}] [range {port |protocol}] [routing][sequence value][time-range name]`	
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.
Step 7	**end** **Example:** `Controller(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 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.

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

SUMMARY STEPS

1. **configure terminal**
2. **interface** interface_id
3. **no switchport**
4. **ipv6 address** *ipv6_address*
5. **ipv6 traffic-filter** *acl_name*
6. **end**
7. **show running-config interface tenGigabitEthernet 1/0/3**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**interface** interface_id **Example:** `Controller# 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:** `Controller# 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:** `Controller# 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:** `Controller# ipv6 traffic-filter access-list-name {in | out}`	Applies the access list to incoming or outgoing traffic on the interface.
Step 6	**end** **Example:** `Controller(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**	Shows the configuration summary.

	Command or Action	Purpose

	Example:
	`Controller# 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`	

Step 8	**copy running-config startup-config** **Example:** `copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Creating WLAN IPv6 ACL

SUMMARY STEPS

1. **ipv6 traffic-filter acl***acl_name*
2. **ipv6 traffic-filter acl web**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**ipv6 traffic-filter acl***acl_name* **Example:** `Controller(config-wlan)# ipv6 traffic-filter acl <acl_name>`	Creates a named WLAN ACL.
Step 2	**ipv6 traffic-filter acl web** **Example:** `Controller(config-wlan)# ipv6 traffic-filter acl web <acl_name-preauth>`	Creates a pre-authentication for WLAN ACL.

```
Controller(config-wlan)# ipv6 traffic-filter acl <acl_name>
Controller(config-wlan)#ipv6 traffic-filter acl web <acl_name-preauth>
```

Verifying IPv6 ACL

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.

SUMMARY STEPS

1. **show access-list**
2. **show ipv6 access-list** *acl_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show access-list** **Example:** `Controller# show access-lists`	Displays all access lists configured on the controller
Step 2	**show ipv6 access-list** *acl_name* **Example:** `Controller# 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.

```
Controller(config)# ipv6 access-list CISCO
Controller(config-ipv6-acl)# deny tcp any any gt 5000
Controller (config-ipv6-acl)# deny ::/0 lt 5000 ::/0 log
Controller(config-ipv6-acl)# permit icmp any any
Controller(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.

```
Controller(config)# interface TenGigabitEthernet 1/0/3
Controller(config-if)# no switchport
Controller(config-if)# ipv6 address 2001::/64 eui-64
Controller(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.

```
Controller #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.

```
Controller# 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
```

Example: Configuring RA Throttling and NS Suppression

This task describes how to create an RA throttle policy in order to help the power-saving wireless clients from being disturbed by frequent unsolicited periodic RA's. The unsolicited multicast RA is throttled by the controller.

Before You Begin

Enable IPv6 on the client machine.

SUMMARY STEPS

1. **configure terminal**
2. **ipv6 nd ra-throttler policy** Mythrottle
3. **throttle-period** 20
4. **max-through** 5
5. **allow at-least** 3 **at-most** 5
6. **switch (config)# vlan configuration 100**
7. **ipv6 nd suppress**
8. **ipv6 nd ra-th attach-policy** attach-policy_name
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**ipv6 nd ra-throttler policy** Mythrottle **Example:** `Controller (config)# ipv6 nd ra-throttler policy Mythrottle`	Creates a RA throttler policy called Mythrottle.
Step 3	**throttle-period** 20 **Example:** `Controller (config-nd-ra-throttle)# throttle-period 20`	Determines the time interval segment during which throttling applies.
Step 4	**max-through** 5 **Example:** `Controller (config-nd-ra-throttle)# max-through 5`	Determines how many initial RA's are allowed.
Step 5	**allow at-least** 3 **at-most** 5 **Example:** `Controller (config-nd-ra-throttle)# allow at-least 3 at-most 5`	Determines how many RA's are allowed after the initial RAs have been transmitted, until the end of the interval segment.
Step 6	**switch (config)# vlan configuration 100** **Example:** `Controller (config)# vlan configuration 100`	Creates a per vlan configuration.
Step 7	**ipv6 nd suppress** **Example:** `Controller (config)# ipv6 nd suppress`	Disables the neighbor discovery on the Vlan.
Step 8	**ipv6 nd ra-th attach-policy** attach-policy_name **Example:** `Controller (config)# ipv6 nd ra-throttle attach-policy attach-policy_name`	Enables the router advertisement throttling.
Step 9	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Example: Configuring RA Guard Policy

SUMMARY STEPS

1. **ipv6 nd raguard policy** MyPloicy
2. **trusted-port**
3. **device-role router**
4. **interface tenGigabitEthernet** 1/0/1
5. **ipv6 nd raguard attach-policy**MyPolicy
6. **vlan configuration** 19-21,23
7. **ipv6 nd suppress**
8. **ipv6 snooping**
9. **ipv6 nd raguard attach-policy** MyPolicy
10. **ipv6 nd ra-throttler attach-policy** Mythrottle

DETAILED STEPS

	Command or Action	Purpose
Step 1	**ipv6 nd raguard policy** MyPloicy **Example:** `Controller (config)# ipv6 nd raguard policy MyPolicy`	
Step 2	**trusted-port** **Example:** `Controller (config-nd-raguard)# trusted-port`	Configures the trusted port for the policy created above.

Step 3	**device-role router**
	
	Example:
	`Controller (config-nd-raguard)# device-role [host|monitor|router|switch]`
	`Controller (config-nd-raguard)# device-role router`
		Defines the trusted device that can send RAs to the trusted port created above.

Step 4	**interface tenGigabitEthernet** 1/0/1 **Example:** `Controller (config)# interface tenGigabitEthernet 1/0/1`	Configures the interface to the trusted device.
Step 5	**ipv6 nd raguard attach-policy**MyPolicy **Example:** `Controller (config-if)# ipv6 nd raguard attach-policy Mypolicy`	Configures and attaches the policy to trust the RA's received from the port.

	Command or Action	Purpose
Step 6	**vlan configuration** 19-21,23 **Example:** `Controller (config)# vlan configuration 19-21,23`	Configures the wireless client vlans.
Step 7	**ipv6 nd suppress** **Example:** `Controller (config-vlan-config)# ipv6 nd suppress`	Suppresses the ND messages over wireless.
Step 8	**ipv6 snooping** **Example:** `Controller (config-vlan-config)# ipv6 snooping`	Captures IPv6 traffic.
Step 9	**ipv6 nd raguard attach-policy** MyPolicy **Example:** `Controller (config-vlan-config)# ipv6 nd raguard attach-policy Mypolicy`	Attaches the RA Guard policy to the wireless client vlans.
Step 10	**ipv6 nd ra-throttler attach-policy** Mythrottle **Example:** `Controller (config-vlan-config)#ipv6 nd ra-throttler attach-policy Mythrottle`	Attaches the RA throttling policy to the wireless client vlans.

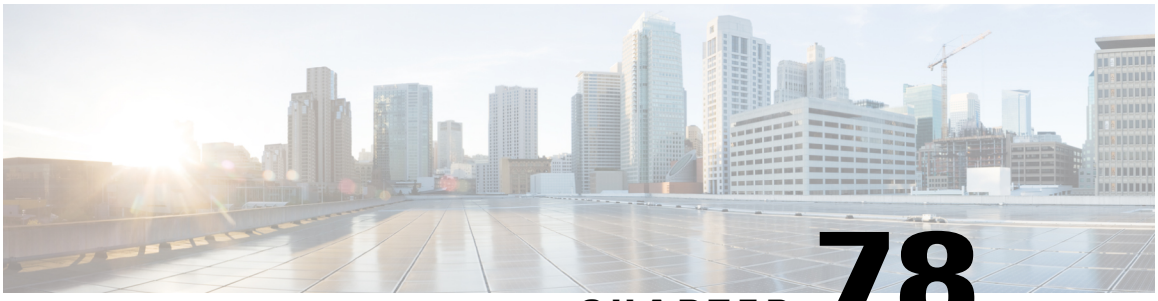
Example: Configuring IPv6 Neighbor Binding

SUMMARY STEPS

1. **ipv6 neighbor binding** [vlan]19 2001:db8::25:4 **interface tenGigabitEthernet** 1/0/3 aaa.bbb.ccc

DETAILED STEPS

	Command or Action	Purpose
Step 1	**ipv6 neighbor binding** [vlan]19 2001:db8::25:4 **interface tenGigabitEthernet** 1/0/3 aaa.bbb.ccc **Example:** `Controller (config)# ipv6 neighbor binding vlan 19 2001:db8::25:4 interface tenGigabitEthernet 1/0/3 aaa.bbb.ccc`	Sets and validates the neighbor 2001:db8::25: 4 only valid when transmitting on VLAN 19 through interface te1/0/3 with the source mac-address as aaa.bbb.ccc.



CHAPTER 78

Configuring IPv6 Web Authentication

Prerequisites for IPv6 Web Authentication

The following configurations must be in place before you start with IPv6 Web Authentication:

- IPv6 Device Tracking.
- IPv6 DHCP Snooping.
- Disable security of type 802.1x on the wlan.
- Each WLAN must have a vlan associated to it.
- Change the default wlan setting from **shutdown** to **no shutdown**.

Restrictions for IPv6 Web Authentication

The following restrictions are implied when using IPv6 web authentication:

Information About IPv6 Web Authentication

Web authentication is a Layer 3 security feature and the controller disallows IP traffic (except DHCP and DNS -related packets) from a particular client until it supplies a valid username and password. It is a simple authentication method without the need for a supplicant or client utility. Web authentication is typically used by customers who deploy a guest-access network. Traffic from both, HTTP and HTTPS, page is allowed to display the login page.

Note Web authentication does not provide data encryption and is typically used as simple guest access for either a hot spot or campus atmosphere, where connectivity is always a factor.

A WLAN is configured as **security webauth** for web based authentication. The controller supports the following types of web based authentication:

- Web Authentication – The client enters the credentials in a web page which is then validated by the Wlan controller.
- Web Consent – The Wlan controller presents a policy page with Accept/Deny buttons. Click Accept button to access the network.

A Wlan is typically configured for open authentication, that is without Layer 2 authentication, when web-based authentication mechanism is used.

Web Authentication Process

The following events occur when a WLAN is configured for web authentication:

- The user opens a web browser and enters a URL address, for example, *http://www.example.com*. The client sends out a DNS request for this URL to get the IP address for the destination. The controller bypasses the DNS request to the DNS server, which in turn responds with a DNS reply that contains the IP address of the destination *www.example.com*. This, in turn, is forwarded to the wireless clients.
- The client then tries to open a TCP connection with the destination IP address. It sends out a TCP SYN packet destined to the IP address of *www.example.com*.
- The controller has rules configured for the client and cannot act as a proxy for *www.example.com*. It sends back a TCP SYN-ACK packet to the client with source as the IP address of *www.example.com*. The client sends back a TCP ACK packet in order to complete the three-way TCP handshake and the TCP connection is fully established.
- The client sends an HTTP GET packet destined to *www.example.com*. The controller intercepts this packet and sends it for redirection handling. The HTTP application gateway prepares an HTML body and sends it back as the reply to the HTTP GET requested by the client. This HTML makes the client go to the default web-page of the controller, for example, *http://<Virtual-Server-IP>/login.html*.
- The client closes the TCP connection with the IP address, for example, *www.example.com*.
- If the client wants to go to virtual IP, the client tries to open a TCP connection with the virtual IP address of the controller. It sends a TCP SYN packet for virtual IP to the controller.
- The controller responds back with a TCP SYN-ACK and the client sends back a TCP ACK to the controller in order to complete the handshake.
- The client sends an HTTP GET for */login.html* destined to virtual IP in order to request for the login page.
- This request is allowed to the web server of the controller, and the server responds with the default login page. The client receives the login page in the browser window where the user can log in.

How to Configure IPv6 Web Authentication

Disabling WPA

Before You Begin

Disable 802.1x. A typical web authentication does not use Layer 2 security. Use this configuration to remove Layer 2 security.

SUMMARY STEPS

1. **configure terminal**
2. **wlan** test1 2 test1
3. **no security** wpa

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**wlan** test1 2 test1 **Example:** `Controller(config)# wlan test1 2 test1`	Creates a WLAN and assign an SSID to it.
Step 3	**no security** wpa **Example:** `Controller(config-wlan)# no security wpa`	Disables the WPA support for Wlan.

What to Do Next

Enable the following:

- Security Web Authentication.
- Parameter Local.
- Authentication List.

Enabling Security on the WLAN

SUMMARY STEPS

1. **parameter-map type web-auth global**
2. **virtual-ip ipv4** 192.0.2.1
3. **virtual-ip ipv6** 2001:db8::24:2

DETAILED STEPS

	Command or Action	Purpose
Step 1	**parameter-map type web-auth global** **Example:** `Controller(config)# parameter-map type web-auth global`	Applies the parameter map to all the web-auth wlans.
Step 2	**virtual-ip ipv4** 192.0.2.1 **Example:** `Controller(config-params-parameter-map)# virtual-ip ipv4 192.0.2.1`	Defines the virtual gateway IPv4 address.
Step 3	**virtual-ip ipv6** 2001:db8::24:2 **Example:** `Controller(config-params-parameter-map)# virtual-ip ipv6 2001:db8::24:2`	Defines the virtual gateway IPv6 address.

Enabling a Parameter Map on the WLAN

SUMMARY STEPS

1. **security web-auth parameter-map** <mapname>

DETAILED STEPS

	Command or Action	Purpose
Step 1	**security web-auth parameter-map** <mapname> **Example:** `Controller(config-wlan)# security web-auth parameter-map webparalocal`	Enables web authentication for the wlan and creates a parameter map.

Enabling Authentication List on WLAN

SUMMARY STEPS

1. **security web-auth authentication-list** webauthlistlocal

DETAILED STEPS

	Command or Action	Purpose
Step 1	**security web-auth authentication-list** webauthlistlocal **Example:** `Controller(config-wlan)# security web-auth`	Enables web authentication for the wlan and creates a local web authentication list.

Configuring a Global WebAuth WLAN Parameter Map

Use this example to configure a global web auth WLAN and add a parameter map to it.

SUMMARY STEPS

1. **parameter-map type webauth global**
2. **virtual-ip ipv6** 2001:db8:4::1
3. **ratelimit init-state-sessions** 120
4. **max-https-conns** 70

DETAILED STEPS

	Command or Action	Purpose
Step 1	**parameter-map type webauth global** **Example:** `Controller (config)# parameter-map type webauth global`	Configures a global webauth and adds a parameter map to it.
Step 2	**virtual-ip ipv6** 2001:db8:4::1 **Example:** `Controller (config-params-parameter-map)# virtual-ip ipv6 2001:db8:4::1`	Defines a virtual gateway IP address that appears to the wireless clients for authentication.
Step 3	**ratelimit init-state-sessions** 120 **Example:** `Controller (config-params-parameter-map)# ratelimit init-state-sessions 120`	Sets the global ratelimit to limit the bandwidth that the web clients can use on the controller to avoid over-flooding attacks.

	Command or Action	Purpose
Step 4	**max-https-conns** 70 **Example:** `Controller (config-params-parameter-map)# max-http-conns 70`	Sets the maximum number of attempted http connections on the controller to avoid over-flooding atatcks.

Configuring the WLAN

Before You Begin

- The WLAN must have a Vlan associated with it. By default, a new Wlan is always associated with Vlan 1, which can be changed as per the configuration requirements.
- Configure and enable the WLAN to *no shutdown*. By default, the Wlan is configured with the *shutdown* parameter and is disabled.

SUMMARY STEPS

1. **wlan** *1*
2. **client vlan** *interface ID*
3. **security web-auth authentication list** webauthlistlocal
4. **security web-auth parameter-map** global
5. **no security wpa**
6. **no shutdown**
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**wlan** *1* **Example:** `Controller(config-wlan)# wlan 1 name vicweb ssid vicweb`	Creates a wlan and assign an SSID to it.
Step 2	**client vlan** *interface ID* **Example:** `Controller(config-wlan)# client vlan VLAN0136`	Assigns the client to vlan interface.

	Command or Action	Purpose
Step 3	**security web-auth authentication list** webauthlistlocal **Example:** `Controller(config-wlan)# security web-auth authentication-list webauthlistlocal`	Configures web authentication for the wlan.
Step 4	**security web-auth parameter-map** global **Example:** `Controller(config-wlan)# security web-auth parameter-map global`	Configures the parameter map on the wlan.
Step 5	**no security wpa** **Example:** `Controller(config-wlan)# no security wpa`	Configures the security policy for a wlan. This enables the wlan.
Step 6	**no shutdown** **Example:** `Controller(config-wlan)# no shutdown`	Configures and enables the Wlan.
Step 7	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

Enabling IPv6 in Global Configuration Mode

Enable IPv6 in global configuration for web authentication.

SUMMARY STEPS

1. **configure terminal**
2. **web-auth global**
3. **virtual IPv6**

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	**web-auth global** **Example:** `Controller(config)# parameter-map type webauth global`	Globally configures the parameter map type as web authentication.
Step 3	**virtual IPv6** **Example:** `Controller(config-params-parameter-map)# virtual-ip ipv6`	Selects IPv6 as the virtaul IP for web authentication. **Note** You can also select IPv4 as the preferred IP for web authentication.

Verifying IPv6 Web Authentication

Verifying the Parameter Map

Use the **show running configuration** command to verify the parameter map configured for Wlan.

SUMMARY STEPS

1. **show running config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show running config** **Example:** `Controllershow running config`	Displays the entire running configuration for the controller. Grep for parameter map to view the result.

```
wlan alpha 2 alpha
 no security wpa
 no security wpa akm dot1x
 no security wpa wpa2
 no security wpa wpa2 ciphers aes
 security web-auth
 security web-auth authentication-list webauthlistlocal
 security web-auth parameter-map webparalocal
```

Verifying Authentication List

Use the **show running configuration** command to verify the authentication list configured for the Wlan.

SUMMARY STEPS

1. **show running configuration**
2. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show running configuration** **Example:** `Controller#show running-config`	Displays the Wlan configuration. `Controller# show running-config`
Step 2	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

```
Controller#show running-config
 ..................................
 ..................................
 ..................................
 wlan alpha 2 alpha
 no security wpa
 no security wpa akm dot1x
 no security wpa wpa2
 no security wpa wpa2 ciphers aes
 security web-auth
 security web-auth authentication-list webauthlistlocal
 security web-auth parameter-map webparalocal
 ..................................
 ..................................
 ..................................
```

CHAPTER 79

Configuring IPv6 Client Mobility

Prerequisites for IPv6 Client Mobility

To enable wireless IPv6 client connectivity, the underlying wired network must support IPv6 routing and an address assignment mechanism such as SLAAC or DHCPv6. The controller must have L2 adjacency to the IPv6 router, and the VLAN needs to be tagged when the packets enter the controller. APs do not require connectivity on an IPv6 network, as all traffic is encapsulated inside the IPv4 CAPWAP tunnel between the AP and controller.

Restrictions For IPv6 Client Mobility

- When using the IPv6 Client Mobility, clients must support IPv6 with either static stateless auto configuration (such as Windows XP clients) or stateful DHCPv6 IP addressing (such as Windows 7 clients).
- To allow smooth operation of stateful DHCPv6 IP addressing, you must have a switch or router that supports the DHCP for IPv6 feature (such as the controller) that is configured to act like a DHCPv6 server, or you need a dedicated server such as a Windows 2008 server with a built-in DHCPv6 server. Cisco Catalyst 3850 switch and Cisco Catalyst 5700 switch can act as (internal) a DHCPv6 server.

Note To load the SDM IPv6 template in the Cisco Catalyst 3850 switch, enter the **sdm prefer dual-ipv4** and **v6** default command and then reset the switch.

Information About IPv6 Client Mobility

The Controller supports IPv6 mobility for IPv6-only or dual-stack nodes. The IPv6 Client Mobility is divided into:

- Link Layer and
- Network Layer

The link layer is handled by the 802.11 protocol which enables the client to roam to any AP in the same BSS (basic service set) identified by the same SSID without losing the link layer connectivity.

However, link layer mobility is not enough to make wireless client Layer 3 applications continue to work seamlessly while roaming. Cisco IOSd's wireless mobility module uses mobility tunneling to retain seamless connectivity for the client's Layer 3 PoP (point of presence) when the client roams across different subnets on different switches.

IPv6 is the next-generation network layer Internet protocol intended to replace IPv4 in the TCP/IP suite of protocols. This new version increases the internet global address space to accommodate users and applications that require unique global IP addresses. IPv6 incorporates 128-bit source and destination addresses, which provide significantly more addresses than the 32-bit IPv4 addresses.

To support IPv6 clients across controllers, ICMPv6 messages must be dealt with specially to ensure the IPv6 client remains on the same Layer 3 network. The controller keep track of IPv6 clients by intercepting the ICMPv6 messages to provide seamless mobility and protect the network from network attacks. The NDP (neighbor discovery packets) packets are converted from multicast to unicast and delivered individually per client. This unique solution ensures that Neighbor Discovery and Router Advertisement packets are not leaked across Vlans. Clients can receive specific Neighbor Discovery and Router Advertisement packets ensuring correct IPv6 addressing and avoids unnecessary multicast traffic.

The configuration for IPv6 mobility is the same as IPv4 mobility and requires no separate software on the client side to achieve seamless roaming. The controller must be part of the same mobility group. Both IPv4 and IPv6 client mobility are enabled by default.

IPv6 client mobility is used for the following:

- Retaining the client IPv6 multiple addresses in Layer-2 and Layer-3 roaming.
- IPv6 Neighbor Discovery Prootcol (NDP) packet management.
- Client IPv6 addresses learning.

Using Router Advertisment

The Neighbor Discovery Protocol(NDP) operates in the link-layer and is responsible for the discovery of other nodes on the link. It determines the link-layer addresses of other nodes, finds the available routers, and maintains reachability information about the paths to other active neighbor nodes.

Router Advertisement (RA) is one of the IPv6 Neighbor Discovery Protocol (NDP) packets that is used by the hosts to discover available routers, acquire the network prefix to generate the IPv6 addresses, link MTU, and so on. The routers send RA on a regular basis, or in response to hosts Router Solicitation messages.

IPv6 wireless client mobility manages the IPv6 RA packet . The converged access controller forwards the link-local all-nodes multicast RA packets to the local and roaming wireless nodes mapped on same VLAN the RA was received on.

Figure 1 illustrates the link-local all-nodes mcast RA forwarding issue in the wireless node mobility.

Figure 57: Roaming Client Receiving Invalid RA from Router 2

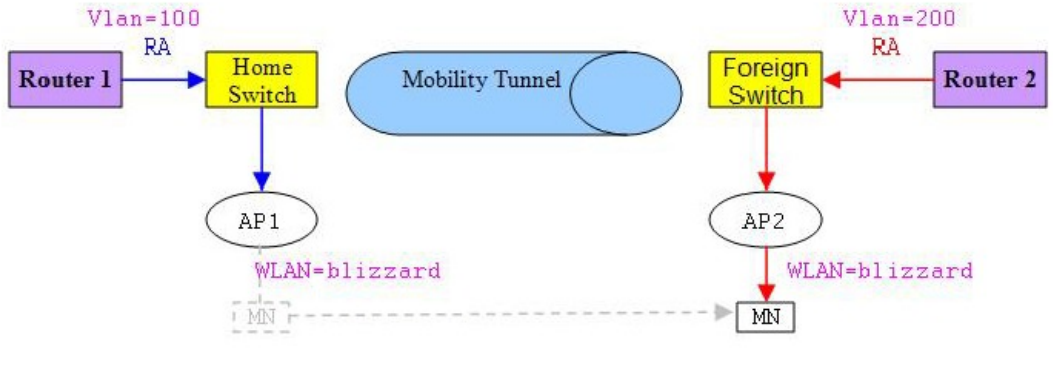
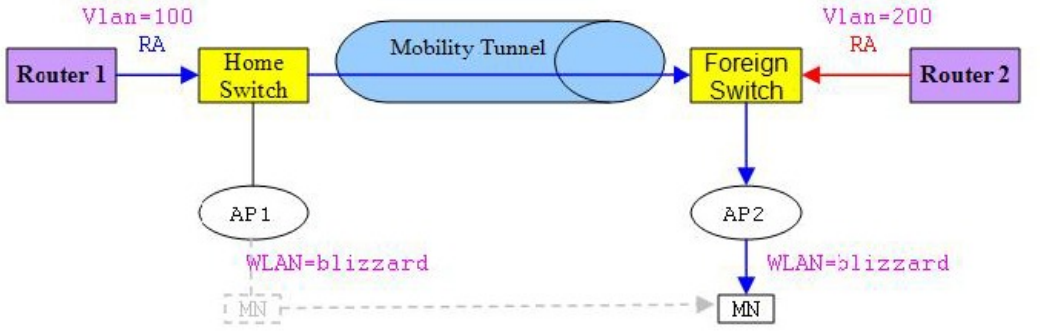


Figure 2 illustrates how a roaming client “MN” receives RA from VLAN 200 in a foreign switch and how it acquires an new IP address and breaks into L3 mobility's point of presence.

Figure 58: Roaming Client Receives Valid RA from Router 1



RA Throttling and NS suppression

To safeguard the power-saving wireless clients form being disturbed by frequent unsolicited periodic RAs, the controller can throttle the unsolicited multicast RA.

IPv6 Address Learning

There are three ways for IPv6 client to acquire IPv6 addresses:

- Stateless Address Auto-Configuration (SLAAC)
- Stateful DHCPv6
- Static configuration

For these methods, the IPv6 client always sends NS DAD (duplicate address detection) to ensure that there is no duplicated IP address on the network. The controller snoops the clients NDP and DHCPv6 packets to learn about its client IP addresses and then updates the controllers database. The database then informs the controller for the clients new IP address.

Handling Multiple IP Addresses

In the case when the new IP address is received after RUN state, whether an addition or removal, the controller updates the new IP addresses on its local database for display purposes. Essentially, the IPv6 uses the existing or same PEM state machine code flow as in IPv4. When the IP addresses are requested by external entities, for example, from Prime Infrastructure, the controller will include all the available IP addresses, IPv4 and IPv6, in the API/SPI interface to the external entities.

An IPv6 client can acquire multiple IP addresses from stack for different purposes. For example, a link-local address for link local traffic, and a routable unique local or global address.

When the client is in the DHCP request state and the controller receives the first IP address notification from the database for either an IPv4 or IPv6 address, the PEM moves the client into the RUN state.

When a new IP address is received after the RUN state, either for addition or removal, the controller updates the new IP addresses on its local database for display purposes.

When the IP addresses are requested by external entities, for example, from Prime Infrastructure, the controller provides the available IP addresses, both IPv4 and IPv6, to the external entities.

IPv6 Configuration

The controller supports IPv6 client as seamlessly as the IPv4 clients. The administrator must manually configure the Vlans to enable the IPV6, IPv6's snooping and throttling functionality. This will enable the NDP packets to throttle between the controller and its various clients

Verifying IPv6 Client Mobility

The commands listed in the Table 1 applies to the IPv6 client mobility.

Table 112: Commands for Verifying IPv6 Client Mobility on Cisco 5760 WLC

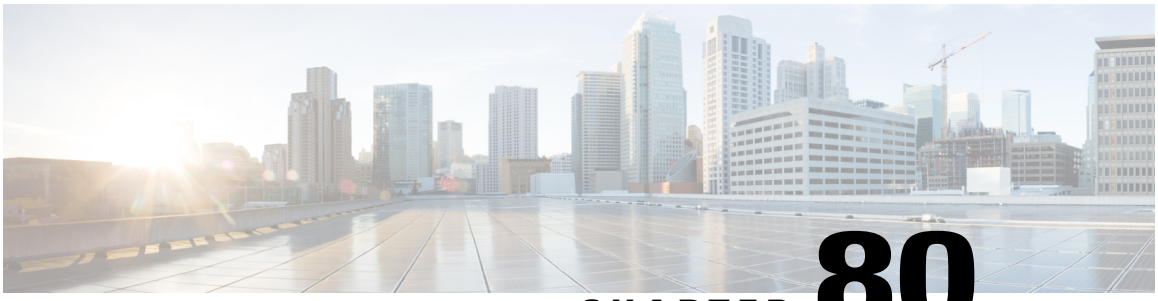
Command	Description
debug mobility ipv6	Enables all the wireless client IPv6 mobility debugs.
debug **client** **mac-address** (mac-addr)	Displays wireless client debugging. Enter a MAC address for debugging information.

Monitoring IPv6 Client Mobility

The commands in Table 2 are used to monitor IPv6 Client mobility on the controller.

Table 113: Monitoring IPv6 Client Mobility Commands

Commands	Description
show wireless client summary	Displays the wireless specific configuration of active clients.
show wireless client mac-address (mac-addr)	Displays the wireless specific configuration of active clients based on their MAC address.



CHAPTER 80

Configuring IPv6 Mobility

Pre-requisites for IPv6 Mobility

The mobility and its related infrastructure must be configured and ready for use.

Information About IPv6 Mobility

Mobility, or roaming, is a wireless LAN client's ability to maintain its association seamlessly from one access point to another securely and with as little latency as possible. This section explains how mobility works whencontroller are included in a wireless network.

When a wireless client associates and authenticates to an access point, the access point's controller places an entry for that client in its client database. This entry includes the client's MAC and IP addresses, security context and associations, quality of service (QoS) contexts, the WLAN, and the associated access point. The controller uses this information to forward frames and manage traffic to and from the wireless client.

When the wireless client moves its association from one access point to another, the controller simply updates the client database with the newly associated access point. If necessary, new security context and associations are established as well. The process becomes more complicated, however, when a client roams from an access point joined to one controller to an access point joined to a different controller. It also varies based on whether thecontroller are operating on the same subnet.

Inter Controller Roaming

When the client associates to an access point joined to a new controller, the new controller exchanges mobility messages with the original controller, and the client database entry is moved to the new controller if sticky anchoring is disabled.

Intra Subnet Roaming with Sticky Anchoring, and Inter Subnet Roaming

Inter-subnet roaming is similar to inter-controller roaming in that the controller exchange mobility messages on the client roam. However, instead of moving the client database entry to the new controller, the original controller marks the client with an "Anchor" entry in its own client database. The database entry is copied to the new controller client database and marked with a "Foreign" entry in the new controller. The roam remains transparent to the wireless client, and the client maintains its original IP address.

In inter-subnet roaming, WLANs on both anchor and foreign controller need to have the same network access privileges and no source-based routing or source-based firewalls in place. Otherwise, the clients may have network connectivity issues after the handoff.

For more information on configuring mobility see, the Cisco 5700 Wireless LAN Controller Mobility Configuration Guide, Cisco IOS XE, Release 3.2SE.

How to Configure IPv6 Mobility

Monitoring IPv6 Mobility

This chapter displays the mobility related IPv6 configuration. To see the mobility related configurations refer to the Cisco 5700 Wireless LAN Controller Mobility Configuration Guide, Cisco IOS XE 3.2SE.

SUMMARY STEPS

1. **show ipv6 neighbors binding mac** C0C1.C06B.C4E2

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show ipv6 neighbors binding mac** C0C1.C06B.C4E2 **Example:** `Controller# show ipv6 neighbors binding mac C0C1.C06B.C4E2`	Displays the IPv6 related mobility configurations.

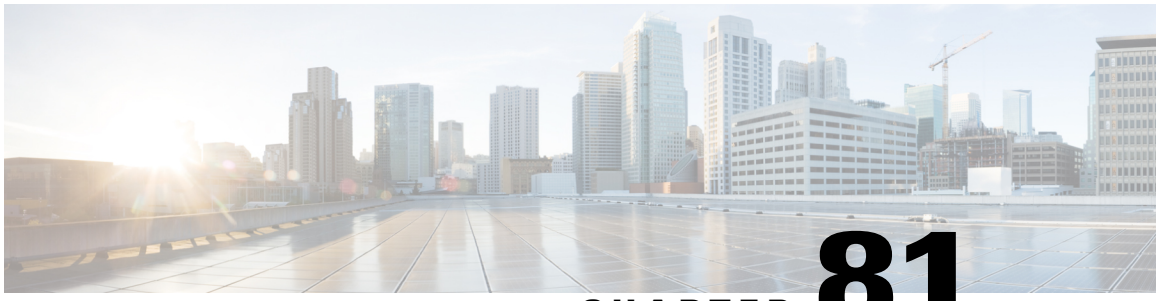
```
Controller# show ipv6 neighbors binding mac C0C1.C06B.C4E2
Binding Table has 45 entries, 37 dynamic (limit 100)
Codes: L - Local, S - Static, ND - Neighbor Discovery, DH - DHCP, PKT - Other Packet, API
- API created
Preflevel flags (prlvl):
0001:MAC and LLA match     0002:Orig trunk           0004:Orig access
0008:Orig trusted trunk    0010:Orig trusted access  0020:DHCP assigned
0040:Cga authenticated     0080:Cert authenticated   0100:Statically assigned


    IPv6 address                            Link-Layer addr Interface vlan prlvl  age
state    Time left
L   FE80:20:25::16                          2037.064C.BA71  Vl25        25  0100 3137mn
REACHABLE
L   FE80:20:24::16                          2037.064C.BA41  Vl24        24  0100 3137mn
REACHABLE
L   FE80:20:23::16                          2037.064C.BA44  Vl23        23  0100 3137mn
```

```
REACHABLE
ND  FE80:20:23::13                          2037.0653.6BC4  Te1/0/1     23  0005   85s
REACHABLE  223 s try 0
ND  FE80:20:22::17                          2037.064D.06F6  Te1/0/1     22  0005    3mn
REACHABLE  92 s try 0
L   FE80:20:22::16                          2037.064C.BA76  Vl22        22  0100 3137mn
REACHABLE
ND  FE80:20:22::13                          2037.0653.6BF6  Te1/0/1     22  0005  165s
REACHABLE  136 s try 0
ND  FE80:20:22::12                          2037.064C.94F6  Te1/0/1     22  0005   23s
REACHABLE  281 s try 0
ND  FE80:20:22::2                           0022.550E.8FC3  Te1/0/1     22  0005   18s
REACHABLE  295 s try 0
ND  FE80:20:21::17                          2037.064D.06E8  Te1/0/1     21  0005    4mn
REACHABLE  60 s try 0
L   FE80:20:21::16                          2037.064C.BA68  Vl21        21  0100 3137mn
REACHABLE
ND  FE80:20:21::13                          2037.0653.6BE8  Te1/0/1     21  0005   57s
REACHABLE  252 s try 0
ND  FE80:20:21::12                          2037.064C.94E8  Te1/0/1     21  0005    4s
REACHABLE  297 s
ND  FE80:20:21::2                           0022.550E.8FC2  Te1/0/1     21  0005    2s
REACHABLE  307 s try 0
ND  FE80::F866:8BE0:12E4:39CF               C0C1.C06B.C4E2  Ca4         21  0005    3mn
REACHABLE  89 s try 0
ND  FE80::6D0A:DB33:D69E:91C7               0050.B606.A6CE  Te1/0/1     22  0005  135s
REACHABLE  171 s try 0
ND  FE80::985:8189:9937:BB05                8CA9.8295.09CC  Ca0         21  0005   15s
REACHABLE  287 s
ND  FE80::20:24:13                          2037.0653.6BC1  Te1/0/1     24  0005  155s
REACHABLE  145 s try 0
L   2001:20:23::16                          2037.064C.BA44  Vl23        23  0100 3137mn
REACHABLE
DH  2001:20:22:0:C96C:AF29:5DDC:2689        0050.B606.A6CE  Te1/0/1     22  0024   19s
REACHABLE  286 s try 0(16574
DH  2001:20:22:0:A46B:90B2:F0DB:F952        0050.B606.A6CE  Te1/0/1     22  0024 2339mn
STALE      32401 s
DH  2001:20:22:0:7DFD:14EC:B1E4:1172        0050.B606.A6CE  Te1/0/1     22  0024 2339mn
STALE      24394 s
DH  2001:20:22:0:7CB3:D6DD:FD6A:50F         0050.B606.A6CE  Te1/0/1     22  0024 2333mn
STALE      29195 s
DH  2001:20:22:0:6D32:AF24:FDE1:2504        0050.B606.A6CE  Te1/0/1     22  0024  509mn
STALE      118821 s
DH  2001:20:22:0:5106:5AD:FE98:A2F0         0050.B606.A6CE  Te1/0/1     22  0024 2328mn
STALE      31362 s
ND  2001:20:22::201:13                      0050.B606.A6CE  Te1/0/1     22  0005   49s
REACHABLE  264 s try 0
L   2001:20:22::16                          2037.064C.BA76  Vl22        22  0100 3137mn
REACHABLE
ND  2001:20:22::13                          2037.0653.6BF6  Te1/0/1     22  0005  175s
REACHABLE  131 s try 0
ND  2001:20:22::2                           0022.550E.8FC3  Te1/0/1     22  0005   28s
REACHABLE  274 s try 0
ND  2001:20:21:0:F866:8BE0:12E4:39CF        C0C1.C06B.C4E2  Ca4         21  0005    4mn
REACHABLE  21 s try 0
ND  2001:20:21:0:C085:9D4C:4521:B777        0021.CC73.AA17  Te1/0/1     21  0005   11s
REACHABLE  290 s try 0
ND  2001:20:21:0:6233:4BFF:FE1A:744C        6033.4B1A.744C  Ca4         21  0005    3mn
REACHABLE  108 s try 0
ND  2001:20:21:0:447E:745D:2F48:1C68        8CA9.8295.09CC  Ca0         21  0005   34s
REACHABLE  276 s
ND  2001:20:21:0:3920:DDE8:B29:AD51         C0C1.C06B.C4E2  Ca4         21  0005    3mn
REACHABLE  87 s try 0
ND  2001:20:21:0:1016:A333:FAD5:6E66        0021.CC73.AA17  Te1/0/1     21  0005    4mn
REACHABLE  18 s try 0
ND  2001:20:21:0:C42:E317:BA9B:EB17         6033.4B1A.744C  Ca4         21  0005    4mn
REACHABLE  61 s try 0
ND  2001:20:21:0:985:8189:9937:BB05         8CA9.8295.09CC  Ca0         21  0005  135s
REACHABLE  173 s try 0
ND  2001:20:21::201:20                      0021.CC73.AA17  Te1/0/1     21  0005    4mn
REACHABLE  43 s try 0
ND  2001:20:21::17                          2037.064D.06E8  Te1/0/1     21  0005    4mn
REACHABLE  50 s try 0
```

```
L   2001:20:21::16                          2037.064C.BA68  Vl21        21  0100 3137mn
REACHABLE
ND  2001:20:21::13                          2037.0653.6BE8  Te1/0/1     21  0005   67s
REACHABLE  237 s try 0
ND  2001:20:21::12                          2037.064C.94E8  Te1/0/1     21  0005    5mn
REACHABLE  512 ms try 0
ND  2001:20:21::2                           0022.550E.8FC2  Te1/0/1     21  0005   12s
REACHABLE  294 s try 0
```

Additional References

For more information on configuring mobility see, the Cisco 5700 Wireless LAN Controller Mobility Configuration Guide, Cisco IOS XE, Release 3.2SE.



CHAPTER 81

Configuring IPv6 NetFlow

Prerequisites For IPv6 Netflow

The networking device must be running a Cisco IOSd release that supports Cisco IOS Flexible NetFlow.

IPv6 Traffic

- One of the following must be enabled on your router and on any interfaces on which you want to enable Flexible NetFlow:
  - Cisco Express Forwarding IPv6 or
  - Distributed Cisco Express Forwarding IPv6.

Restrictions For IPv6 Netflow

The following restrictions apply to IPv6 Netflow configurations:

- Locally generated traffic (traffic that is generated by the router, Cisco WLC 5760, on which the Flexible NetFlow Output Accounting feature is configured) is not counted as flow traffic for the Output Flexible NetFlow Accounting feature.
- The Flexible NetFlow Output Accounting feature counts CEF-switched packets only. Process switched transit packets are not counted.

Information About IPv6 Netflow

NetFlow is a monitoring feature used on customer applications for network monitoring, user monitoring and profiling, network planning, security analysis, billing and accounting, and data warehousing and mining. You can use Flexible NetFlow on uplink ports to monitor user-defined flows, collect flow statistics, and perform per-flow policing. It collects and exports flow statistics to a collector device.

Note Not all of the Flexible NetFlow commands in the command reference are available on the controller. Unsupported commands are either not visible or generate an error message if entered.

Understanding Flexible Netflow

With Flexible NetFlow, traffic is processed and packets are classified into flows. New flows are inserted in the NetFlow table, and statistics are automatically updated. You must configure both ingress and egress NetFlow monitoring. The network services module supports one monitor per interface per direction.

Flexible NetFlow consists of the following components:

- Records— These are combinations of key and non-key fields assigned to monitor Flexible NetFlow monitors to define the cache used to store data.
- Flow monitors— These are applied to interfaces to perform network traffic monitoring. A flow monitor includes a user-defined record, an optional flow exporter, and a cache that is automatically created when the monitor is applied to the first interface. The switch supports normal caches that age out according to settings.
- Flow exporters— These export the data in the flow monitor cache to a remote system, such as a server running NetFlow collector.
- Flow samplers— These reduce the load that Flexible NetFlow puts on the networking device to monitor traffic by limiting the number of packets that are analyzed.

You can configure unidirectional flow (destination or source-address based flows), and flow aging. The following features are supported on the network services module:

- Configuring collection statistics for Layer 2-switched (non-routing) traffic, Layer 3 (CAPWAP) IPv4 and IPv6 traffic, and Layer 4 TCP, IGMP, and ICMP traffic.
- NetFlow counting, maintenance, troubleshooting (debugging commands).
- NetFlow analysis is performed on traffic crossing the physical interfaces on the network services module. The controller processes egress (outbound) traffic after forwarding decisions are performed. Locally switched or routed traffic is forced through service module ports by configuring private VLANs or protected ports.

The following NetFlow characteristics are not supported:

- Netflow-5 protocol
- Predefined flow records
- ISL

- Policy-based NetFlow
- Cisco TrustSec monitoring

Though other modules that can be installed in the controller have 1-Gigabit and 10-Gigabit uplink interfaces, NetFlow is supported only on the network services module.

IPv6 Netflow

Flexible Netflow (FNF) allows the user to define a flow record (a particular set of key, non-key, counter and time-stamp fields of interest) that is optimal for a particular application by selecting the fields from a big collection of pre-defined fields, using CLI configuration commands.

The collection of the pre-defined fields includes the following fields:

- Data-link layer (L2) header fields
- IPv4 header fields
- IPv6 header fields
- Transport layer (L4) header fields
- Application layer (L5) header fields
- Routing attributes (generic, IPv4, IPv6)
- Interface fields
- Counter fields
- Timestamp fields

How To Configure IPv6 Netflow

Configuring a Customized Flow Record

You can match the following fields for the flow record:

- IPv4 or IPv6 destination address
- Datalink fields, to identify Layer 2 source and destination address and VLAN for traffic entering or leaving the interfaces, providing the MAC address of the directly connected host. Class of Service (CoS) and Ethertype datalink header fields are also available.
- Transport field source and destination ports, to identify the type of application: ICMP, IGMP, or TCP traffic.

You can collect the following fields for the flow record:

- The total number of bytes, flows or packets sent by the exporter (exporter) or the number of bytes or packets in a 64-bit counter (long). The timestamp based on system uptime from the time the first packet was sent or from the time the most recent (last) packet was seen.

- The SNMP index of the input or output interface. The interface for traffic entering or leaving the service module is based on the switch forwarding cache. This field is typically used in conjunction with datalink, IPv4, and IPv6 addresses, and provides the actual first-hop interface for directly connected hosts.
  - A value of 0 means that interface information is not available in the cache.
  - Some NetFlow collectors require this information in the flow record.

The following steps configure the customized flow record:

SUMMARY STEPS

1. **configure terminal**
2. **flow record** recordname
3. **description** description
4. **match** {ipv4 | ipv6} {destination | hop-limit | protocol | source | traffic-class| version} **address**
5. **match datalink** [dot1q | ethertype | mac | vlan]
6. **match transport** [destination-port | icmp |igmp | source-port]
7. **match interface** [input |output]
8. **match flow direction**
9. **collect counter** {bytes [layer2 | long] | packets [long]}
10. **collect timestamp absolute** [first | last]
11. **collect interface** [input | output]
12. **collect transport tcp flags** {ack | cwr | ece | fin | psh | rst | syn | urg}
13. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**flow record** recordname **Example:** `Controller(config)# flow record TestRecordName`	Creates a flow record and enters Flexible NetFlow flow record configuration mode. This command can also modify an existing flow record.
Step 3	**description** description **Example:** `Controller(config-flow-record)# description SampleNetflowDescription`	(Optional) Creates a description for the flow record.

	Command or Action	Purpose
Step 4	**match**{ipv4 | ipv6}{destination | hop-limit | protocol | source | traffic-class| version} **address** **Example:** `Controller(config-flow-record)# match ipv6 destination address`	Configures key ipv4 and ipv6 fields for the flow record.
Step 5	**match datalink** [dot1q | ethertype | mac | vlan] **Example:** `Controller(config-flow-record)# match datalink [dot1q | ethertype | mac | vlan]`	Configures key datalink (layer 2) fields for the flow record.
Step 6	**match transport** [destination-port | icmp |igmp | source-port] **Example:** `Controller(config-flow-record)# match transport [destination-port | icmp |igmp | source-port]`	Configures key transport layer fields for the flow record.
Step 7	**match interface** [input |output] **Example:** `Controller(config-flow-record)# match interface input`	Configures key interface fields for the flow record.
Step 8	**match flow direction** **Example:** `Controller(config-flow-record)# match flow direction`	Configures key flow identity fields for the flow record.
Step 9	**collect counter** {bytes [layer2 | long] | packets [long]} **Example:** `Controller(config-flow-record)#collect counter bytes layer2 long`	Configures the counter key field for the flow record.
Step 10	**collect timestamp absolute** [first | last] **Example:** `Controller(config-flow-record)# collect timestamp absolute [first | last ]`	Configures the timestamp key field for the flow record.
Step 11	**collect interface** [input | output] **Example:** `Controller(config-flow-record)# collect interface [input | output]`	Configures the interface key field for the flow record.
Step 12	**collect transport tcp flags** {ack | cwr | ece | fin | psh | rst | syn | urg} **Example:** `Controller(config-flow-record)# collect transport tcp flags ack`	Configures transports tcp flag fields for the flow record.

	Command or Action	Purpose
Step 13	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

```
Controller(config)# flow record
Controller(config-flow-record)# description record to monitor network traffic
Controller(config-flow-record)# match ipv6 destination address
Controller(config-flow-record)# match datalink [dot1q | ethertype | mac | vlan]
Controller(config-flow-record)# match transport [destination-port | icmp |igmp | source-port]
Controller(config-flow-record)# match interface input
Controller(config-flow-record)# match flow direction
Controller(config-flow-record)#collect counter bytes layer2 long
Controller(config-flow-record)# collect timestamp absolute first
Controller(config-flow-record)# collect interface [input | output]
Controller(config-flow-record)# collect transport tcp flags ack
Controller(config-flow-record)# end
```

Configuring the Flow Exporters

The following steps are used to configure the NetFlow exporter.

Note The optional export-protocol flow exporter configuration command specifies the NetFlow export protocol used by the exporter. The switch supports only netflow-v9. Though visible in the CLI help, netflow-5 is not supported.

SUMMARY STEPS

1. **configure terminal**
2. **flow exporter** exporter-name
3. **description** description
4. **destination** {hostname | ip-address} **vrf** vrf-name
5. **dscp** <0-63>
6. **source** interface-id
7. **option** {exporter-stats | interface-table | sampler-table} **timeout** seconds]
8. **export-protocolnetflow-v9**
9. **template data timeout** seconds
10. **transport udp** udp-port
11. **ttl** seconds
12. **end**

DETAILED STEPS

	Command or Action
Step 1	**configure terminal** **Example:** `Controller# configure terminal`
Step 2	**flow exporter** exporter-name **Example:** `Controller(config)# flow exporter TestNetFlowExporterName`
Step 3	**description** description **Example:** `Controller(config-flow-exporter)# description SampleNetFlowExporterDescription`
Step 4	**destination** {hostname | ip-address} **vrf** vrf-name **Example:** `Controller(config-flow-exporter)# destination 198.51.100.120 vrf SampleVrfName`
Step 5	**dscp** <0-63> **Example:** `Controller(config-flow-exporter)# dscp 23`

	Command or Action

Step 6	**source** interface-id
	
	Example:
	`Controller(config-flow-exporter)# source {`
	`Auto-Template|Capwap|GigabitEthernet|GroupVI|InternalInterface|Loopback|Null|Port-channel|TenGigabitEthernet|Tunne`

Step 7	**option** {exporter-stats | interface-table | sampler-table} **timeout** seconds] **Example:** `Controller(config-flow-exporter)# option exporter-stats timeout 600`
Step 8	**export-protocolnetflow-v9** **Example:** `Controller(config-flow-exporter)# export-protocol netflow-v9`

Step 9	**template data timeout** seconds
	
	Example:
	`Controller(config-flow-exporter)# template data timeout 600`
	`Controller(config-flow-exporter)#`

Step 10	**transport udp** udp-port **Example:** `Controller(config-flow-exporter)# transport udp 67`

	Command or Action
Step 11	**ttl** seconds **Example:** `Controller(config-flow-exporter)# ttl 100`
Step 12	**end** **Example:** `Controller(config)# end`

```
Controller(config)# flow exporter QoS-Collector
Controller(config-flow-exporter)# description QoS Collector Bldg 19
Controller(config-flow-exporter)# destination 172.20.244.28
Controller(config-flow-exporter)# source vlan 1
Controller(config-flow-exporter)# dscp 3
Controller(config-flow-exporter)# transport udp 2055
Controller(config-flow-exporter)# end
```

What to Do Next

Configuring a Customized Flow Monitor.

Configuring a Customized Flow Monitor

The following steps are used to configure a NetFlow monitor.

SUMMARY STEPS

1. **configure terminal**
2. **flow monitor** monitor -name
3. **description** description
4. **record** {TestNetflowRecordName|TestRecord}
5. **cache** {timeout [active| inactive|update] (seconds) | type (normal)}
6. **cache** {timeout [active| inactive|update] (seconds) | type (normal)}
7. **exporter** TestNetFlowExporterName
8. **cache** {timeout [active| inactive|update] (seconds) | type (normal)}
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**flow monitor** monitor -name **Example:** `Controller(config)# flow monitor SampleMonitorName`	Creates a flow monitor and enters Flexible NetFlow flow monitor configuration mode. You can also use this command to modify an existing flow monitor.
Step 3	**description** description **Example:** `Controller(config-flow-monitor)# Description SampleNetFlowMonitorName`	(Optional) Configures a description for the flow monitor.
Step 4	**record** {TestNetflowRecordName|TestRecord} **Example:** `Controller(config-flow-monitor)#record TestNetflowRecordName`	Specifies the record for the flow monitor.
Step 5	**cache** {timeout [active| inactive|update] (seconds)	type (normal)} **Example:** `Controller(config-flow-monitor)# cache type normal`	(Optional) Modifies the flow monitor cache parameters such as timeout values, number of cache entries, and the cache type. • **timeout active** *seconds*—Configures the active flow timeout. This defines the granularity of the traffic analysis. The range is from 1 to 604800 seconds. The default is 1800. Typical values are 60 or 300 seconds. See the Configuring Data Export for Cisco IOS Flexible NetFlow with Flow Exporters document for recommended values. • **type normal**—Configures normal flow removal from the flow cache.

	Command or Action	Purpose
		Note Although visible in the command line help, the entries keyword and inactive and update timeouts are not supported.
Step 6	**cache** {timeout [active| inactive|update] (seconds) | type (normal)} **Example:** `Controller(config-flow-monitor)# cache type normal`	Repeat step 5 to configure additional cache parameters for the flow monitor.
Step 7	**exporter** TestNetFlowExporterName **Example:** `Controller(config-flow-monitor)# exporter TestNetFlowExporterName`	(Optional) Specifies the name of an exporter that was created previously.
Step 8	**cache** {timeout [active| inactive|update] (seconds) | type (normal)} **Example:** `Controller(config-flow-monitor)# cache type normal`	Repeat step 5 to configure additional cache parameters for the flow monitor.
Step 9	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

```
Controller(config)# flow monitor FLOW-MONITOR-1
Controller(config-flow-monitor)# Used for ipv4 traffic analysis
Controller(config-flow-monitor)# record FLOW-RECORD-1
Controller(config-flow-monitor)# cache timeout active 300
Controller(config-flow-monitor)# cache type normal
Controller(config-flow-monitor)# exporter EXPORTER-1
Controller(config-flow-monitor)# exit
```

What to Do Next

Apply a flow monitor to an interface

Applying a Flow Monitor to an Interface

The following are used to configure a NetFlow monitor to an interface.

SUMMARY STEPS

1. **configure terminal**
2. **interface** interface-id
3. **wlan** ssid
4. [ip | ipv6 | datalink] **flow monitor** monitor -name **sampler** [sampler | input | output]
5. **exit**
6. Repeat steps 2 and 3
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**interface** interface-id **Example:** `Controller(config)# interface tengigabitEthernet 1/0/1`	Identifies an interface and enters interface configuration mode. Flexible Net Flow is supported only on the service module 1-Gigabit or 10-Gigabit Ethernet interfaces. **Note** You cannot attach a NetFlow monitor to a port channel interface. If both service module interfaces are part of an EtherChannel, you should attach the monitor to both physical interfaces.
Step 3	**wlan** ssid **Example:** `Controller (config)# wlan test 1 test`	Configures the flow monitor on WLAN.
Step 4	[ip | ipv6 | datalink] **flow monitor** monitor -name **sampler** [sampler | input | output] **Example:** `Controller(config-if)# ipv6 flow monitor SampleMonitorName input`	Activates a previously created flow monitor by assigning it to the interface to analyze incoming or outgoing traffic. • ip—Enters record matching IPv4 IP addresses. • ipv6—Enters record matching IPv6 IP addresses. **Note** This keyword is visible only when the dual IPv4 and IPv6 Switch Database Management (SDM) template is configured on the switch. • input—Applies the flow monitor on input traffic. • output—Applies the flow monitor on output traffic. • sampler—(Optional) Applies the flow monitor sampler.

Step 5	**exit**
	
	Example:
	`Controller(config-if)# exit`
	`Controller(config)#`
		Returns to global configuration mode.

	Command or Action	Purpose
Step 6	Repeat steps 2 and 3 **Example:**	Configures additional cache parameters for the flow monitor.
Step 7	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

```
Controller(config)# interface tengigabitethernet 1/0/1
Controller(config-if)# ip flow monitor FLOW-MONITOR-1 input
Controller(config-if)# ip flow monitor FLOW-MONITOR-2 output
Controller(config-if)# end
```

Configuring and Enabling Flow Sampling

The following steps are used to configure and enable flow sampling.

SUMMARY STEPS

1. **configure terminal**
2. **sampler** sampler -name
3. **description** description
4. **mode** {deterministic|random} (<1-1>)**out-of** <2-1024>
5. **end**
6. **interface** interface-id
7. **wlan** ssid
8. {ip | ipv6 | datalink] **flow monitor** monitor-name **sampler** sampler-name {input | output}
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**sampler** sampler -name **Example:** `Controller(config)# sampler SampleNameForSAMPLER`	Creates a flow monitor and enters Flexible NetFlow sampler configuration mode. You can also use this command to modify an existing sampler.

	Command or Action	Purpose
Step 3	**description** description **Example:** `Controller(config-sampler)#description SamplerName_1`	(Optional) Configures a description for the sampler.
Step 4	**mode** {deterministic|random} (<1-1>)**out-of** <2-1024> **Example:** `Controller(config-sampler)#mode random 1 out-of 2`	Specifies the mode and window size from which to select packets. The window size range is from 2 to 1024. **Note** Although visible in the CLI help, the mode deterministic keyword is not supported.
Step 5	**end** **Example:** `Controller(config-sampler)# end`	Returns to global configuration mode.
Step 6	**interface** interface-id **Example:** `Controller(config)# interface tengigabitethernet 1/0/1`	Identifies an interface and enters interface configuration mode.
Step 7	**wlan** ssid **Example:** `Controller(config)# wlan test 1 test`	Configures to apply flow sampler on WLAN.
Step 8	{ip | ipv6 | datalink] **flow monitor** monitor-name **sampler** sampler-name {input | output} **Example:** `Controller(config-if)# ip flow monitor FLOW-MONITOR-1 sampler SAMPLE-1 input`	Activates a previously created IPv4 or IPv6 flow monitor by assigning it to the interface to analyze traffic.
Step 9	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode.

```
Controller(config)# sampler SAMPLER-1
Controller(config-sampler)# description Sample at 50
Controller(config-sampler)# mode random 1 out-of 2
Controller(config-sampler)# exit
Controller(config)# interface tengigabitethernet 1/0/1
Controller(config)# wlan test 1 test
Controller(config-if)# ip flow monitor FLOW-MONITOR-1 sampler SAMPLE-1 input
```

What to Do Next

How to configure netflow v9 for IPv6.

Verifying IPv6 Netflow

This section describes the Netflow related **show** commands for IPv6. The following commands can be used to verify Netflow on the controller.

Command	Purpose
show flow record	Displays the status of the flow records.
show flow ssid <ssid_name>	Displays SSID interface information.
show flow monitor {monitor name} {cache|provisioning|statistics}	Displays the flow monitor information.
show flow exporter exporter-name	Displays the status of a flow exporter.
show flow monitor monitor -name	Displays the current status of a flow monitor.
show flow interface interface-id	Verifies that the Flexible NetFlow is configured on the interface.
show flow monitor monitor -name cache format [csv | record | table}	Displays data in the flow monitor cache.
show sampler sampler -name	Displays the current status of a flow sampler.

Monitoring IPv6 Netflow

This section describes the Netflow commands for IPv6.The following commands can be used to monitor Netflow on the controller.

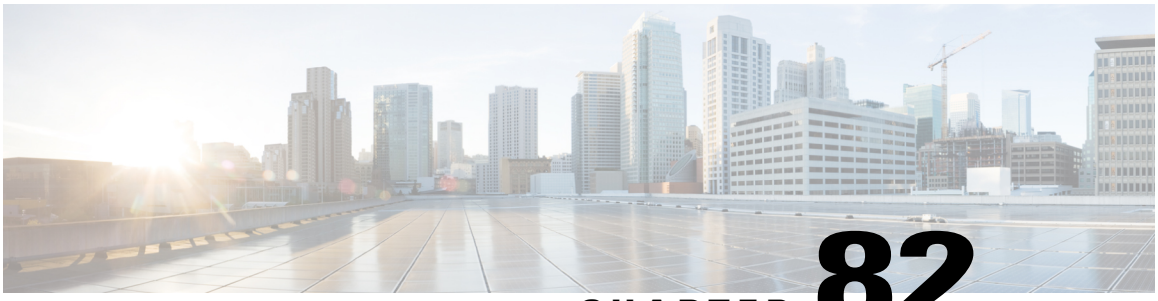
Command	Purpose
show running-config flow record	Displays the configured flow records.
show running-config flow exporter exporter-name	Verifies the configured flow exporter.
show running-config flow monitor monitor -name	Verifies the flow monitor configuration.

Additional References

- For more detailed information about Flexible NetFlow, see the NetFlow Configuration Guide.
- For information about the commands, see the Cisco IOS Flexible NetFlow Command Reference.
- For more information about configuring Flexible NetFlow flow exporters, see: Configuring Data Export for Cisco IOS Flexible NetFlow with Flow Exporters.

PART XV

Flexible Netflow



CHAPTER 82

Configuring Flexible NetFlow

This module contains the following topics:

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Flexible NetFlow

The following are prerequisites for your Flexible NetFlow configuration:

- You must configure a source interface. If you do not configure a source interface, the exporter will remain in a disabled state.
- You must configure a valid record name for every flow monitor.

Prerequisites for Wireless Flexible NetFlow

The following are the prerequisites for wireless Flexible NetFlow:

- Ensure that the networking device is running a Cisco release that supports wireless Flexible NetFlow.
- Ensure that the target is connected to a WLAN.
- The networking device must be configured to support protocol types such as IP, IPv6, and datalink.
- Valid flow record and monitor are required before generating the flow.

Restrictions for Flexible NetFlow

The following are restrictions for Flexible NetFlow:

- Traditional NetFlow (TNF) accounting is not supported.
- Flexible NetFlow v5 export format is not supported, only NetFlow v9 export format is supported.
- Both ingress and egress NetFlow accounting is supported.
- Microflow policing feature shares the NetFlow hardware resource with FNF.
- Only one flow monitor per interface and per direction is supported.
- Layer 2, IPv4, and IPv6 traffic types are supported; however, the controller can apply a flow monitor to only one of these types at a time for a given direction and interface.
- Layer 2, VLAN, and Layer 3 interfaces are supported, but the controller does not support SVI and tunnels.
- The following NetFlow table sizes are supported:

Trim Level	Ingress NetFlow Table	Egress NetFlow Table
LAN Base	Not supported	Not supported
IP Base	8 K	16 K
IP Services	8 K	16 K

- The controller supports three ASICs and each ASIC has 8K and 16K entries.
- The NetFlow tables are on separate compartments and cannot be combined. Depending on which ASIC processed the packet, the flows will be created in the table in the corresponding ASIC.
- Both full flow accounting and sampled NetFlow accounting are supported.
- NetFlow hardware implementation supports four hardware samplers. You can select a sampler rate from 1 out of 2 to 1 out of 1024. Only random sampling mode is supported.
- With the microflow policing feature (which is enabled only for wireless implementation), NetFlow can and should be used only in full flow mode i.e. NetFlow policing cannot be used. For wireless traffic, applying a sampler is not permitted, as it hinders microflow QoS.

- Only full flow accounting is supported for wireless traffic.
- NetFlow hardware uses hash tables internally. Hash collisions can occur in the hardware. Therefore, in spite of the internal overflow Content Addressable Memory (CAM), the actual NetFlow table utilization could be about 80 percent.
- Depending on what fields are used for the flow, a single flow could take two consecutive entries. IPv6 flows also take two entries. In these situations, the effective usage of NetFlow entries is half the table size, which is separate from the above hash collision limitation.
- The controller supports up to 16 flow monitors.
- Microflow policing uses a separate set of flow monitors (limit 3).
- SSID-based NetFlow accounting is supported. SSID is treated in a manner similar to an interface. However, certain fields are not supported (such as AP MAC address and user ID).
- NetFlow v9 format NetFlow export is supported.
- Ingress flows are present in the ASIC that first received the packets for the flow. Egress flows are present in the ASIC from which the packets actually left the controller set up.
- The reported value for the bytes count field (called “bytes long”) is Layer-2-packet-size—18 bytes. For classic Ethernet traffic (802.3), this will be accurate. For all other Ethernet types, this field will not be accurate. Use the "bytes layer2” field, which always reports the accurate Layer 2 packet size. For information about supported Flexible NetFlow fields, see Supported Flexible NetFlow Fields, on page 1282.

Information About Flexible NetFlow

NetFlow is a Cisco technology that provides statistics on packets flowing through the controller. NetFlow is the standard for acquiring IP operational data from IP networks. NetFlow provides data to enable network and security monitoring, network planning, traffic analysis, and IP accounting. Flexible NetFlow improves on original NetFlow by adding the capability to customize the traffic analysis parameters for your specific requirements. Flexible NetFlow facilitates the creation of more complex configurations for traffic analysis and data export through the use of reusable configuration components.

Flexible NetFlow Overview

Flexible NetFlow uses flows to provide statistics for accounting, network monitoring, and network planning.

A flow is a unidirectional stream of packets that arrives on a source interface and has the same values for the keys. A key is an identified value for a field within the packet. You create a flow using a flow record to define the unique keys for your flow.

The controller supports the Flexible NetFlow feature that enables enhanced network anomalies and security detection. Flexible NetFlow allows you to define an optimal flow record for a particular application by selecting the keys from a large collection of predefined fields.

All key values must match for the packet to count in a given flow. A flow might gather other fields of interest, depending on the export record version that you configure. Flows are stored in the Flexible NetFlow cache.

You can export the data that Flexible NetFlow gathers for your flow by using an exporter and export this data to a remote Flexible NetFlow collector.

You define the size of the data that you want to collect for a flow using a monitor. The monitor combines the flow record and exporter with the Flexible NetFlow cache information.

Wireless Flexible NetFlow Overview

The wireless Flexible NetFlow infrastructure supports the following:

- Flexible NetFlow Version 9.0
- User-based rate limiting
- Microflow policing
- Voice and video flow monitoring
- Reflexive access control list (ACL)

Microflow Policing and User-Based Rate Limiting

Microflow policing associates a 2-color 1-rate policer and related drop statistics to each flow present in the NetFlow table. When the flow mask comprises all packet fields, this functionality is known as microflow policing. When the flow mask comprises either source or destination only, this functionality is known as user-based rate limiting.

Voice and Video Flow Monitoring

Voice and video flows are full flow mask-based entries. The ASIC provides the flexibility to program the policer parameters, share policers across multiple flows and rewrite the IP address and Layer 4 port numbers of these flows.

Note For dynamic entries, the NetFlow engine will use the policer parameters that are derived for the flow based on the policy (ACL/QoS-based policies). Dynamic entries cannot share policer across multiple flows.

Reflexive ACL

Reflexive ACLs allow IP packets to be filtered based on upper-layer session information. The ACLs allow outbound traffic and limit inbound traffic in response to the sessions that originate inside the trusted network. The reflexive ACLs are transparent to the filtering mechanism until a data packet that matches the reflexive entry activates it. At this time, a temporary ACL entry is created and added to the IP-named access lists. The information obtained from the data packet to generate the reflexive ACL entry is permit/deny bit, the source IP address and port, the destination IP address, port, and the protocol type. During reflexive ACL entry evaluation, if the protocol type is either TCP or UDP, then the port information must match exactly. For other protocols, there is no port information to match. After this ACL is installed, the firewall is then opened for the reply packets to pass through. At this time, a potential hacker could have access to the network behind the firewall. To narrow this window, an idle timeout period can be defined. However, in the case of TCP, if two FIN bits or an RST is detected, the ACL entry can be removed.

Related Topics

Configuring WLAN to Apply Flow Monitor in IPV4 and IPv6 Input/Output Direction, on page 1300
Example: Configuring IPv4 Flexible NetFlow in WLAN (Ingress Direction), on page 1302
Example: Configuring IPv6 and Transport Flag Flexible NetFlow in WLAN (Egress Direction), on page 1302

Example: Configuring IPv6 Flexible NetFlow in WLAN (Both Ingress and Egress Directions), on page 1303

Flow Records

A flow record defines the keys that Flexible NetFlow uses to identify packets in the flow, as well as other fields of interest that Flexible NetFlow gathers for the flow. You can define a flow record with any combination of keys and fields of interest. The controller supports a rich set of keys. A flow record also defines the types of counters gathered per flow. You can configure 64-bit packet or byte counters. The controller enables the following match fields as the defaults when you create a flow record:

- match datalink—Layer 2 attributes
- match flow—Flow identifying attributes
- match interface—Interface attributes
- match ip4—IPv4 attributes
- match ipv6—IPv6 attributes
- match transport—Transport layer fields

Related Topics

Creating a Flow Record, on page 1287

Flexible NetFlow Match Parameters

The following table describes Flexible NetFlow match parameters. You must configure at least one of the following match parameters for the flow records.

Table 114: Match Parameters

Command	Purpose
match datalink {**dot1q** | **ethertype** | **mac** | **vlan** }	Specifies a match to datalink or Layer 2 fields. The following command options are available: • **dot1q**—Matches to the dot1q field. • **ethertype**—Matches to the ethertype of the packet. • **mac**—Matches the source or destination MAC fields. • **vlan**—Matches to the VLAN that the packet is located on (input or output).
match flow direction	Specifies a match to the flow identifying fields.

Command	Purpose
match interface {**input** | **output**}	Specifies a match to the interface fields. The following command options are available: • **input—**Matches to the input interface. • **output—**Matches to the output interface.
match ipv4 {**destination** | **protocol** | **source** | **tos** | **ttl** | **version**}	Specifies a match to the IPv4 fields. The following command options are available: • **destination—**Matches to the IPv4 destination address-based fields. • **protocol—**Matches to the IPv4 protocols. • **source—**Matches to the IPv4 source address based fields. • **tos—**Matches to the IPv4 Type of Service fields. • **ttl—**Matches to the IPv4 Time To Live fields. • **version—**Matches to the IP version from the IPv4 header.
match ipv6 {**destination** | **hop-limit** | **protocol** | **source** | **traffic-class** | **version** }	Specifies a match to the IPv6 fields. The following command options are available: • **destination—**Matches to the IPv6 destination address-based fields. • **hop-limit—**Matches to the IPv6 hop limit fields. • **protocol—**Matches to the IPv6 payload protocol fields. • **source—**Matches to the IPv6 source address based fields. • **traffic-class—**Matches to the IPv6 traffic class. • **version—**Matches to the IP version from the IPv6 header.

Command	Purpose
match transport {**destination-port** | **igmp** | **icmp** | **source-port**}	Specifies a match to the Transport Layer fields. The following command options are available: • **destination-port—**Matches to the transport destination port. • **icmp—**Matches to ICMP fields, including ICMP IPv4 and IPv6 fields. • **igmp—**Matches to IGMP fields. • **source-port—**Matches to the transport source port.

Flexible NetFlow Collect Parameters

The following table describes the Flexible NetFlow collect parameters.

Table 115: Collect Parameters

Command	Purpose
collect counter { **bytes** { **layer2** { **long** } | **long** } | **packets** { **long** } }	Collects the counter fields total bytes and total packets.
collect interface {**input** | **output**}	Collects the fields from the input or output interface.
collect timestamp absolute {**first** | **last**}	Collects the fields for the absolute time the first packet was seen or the absolute time the most recent packet was last seen (in milliseconds).

Command	Purpose
collect transport tcp flags	Collects the following transport TCP flags: • **ack—**TCP acknowledgement flag • **cwr—**TCP congestion window reduced flag • **ece—**TCP ECN echo flag • **fin—**TCP finish flag • **psh—**TCP push flag • **rst—**TCP reset flag • **syn—**TCP synchronize flag • **urg—**TCP urgent flag **Note** On the controller, you cannot specify which TCP flag to collect. You can only specify to collect transport TCP flags. All TCP flags will be collected with this command.

Exporters

An exporter contains network layer and transport layer details for the Flexible NetFlow export packet. The following table lists the configuration options for an exporter.

Table 116: Flexible NetFlow Exporter Configuration Options

Exporter Configuration	Description
default	Sets a command to its default values.
description	Provides a description for the flow exporter.
destination	Export destination.
dscp	Optional DSCP value.
exit	Exits from the flow exporter configuration mode.
export-protocol	Export protocol version.
no	Negates the command or its default.
option	Selects option for exporting.
source	Originating interface for the net flow.
template	Flow exporter template configuration.

Exporter Configuration	Description
transport	Transport protocol.
ttl	Optional TTL or hop limit.

The controller exports data to the collector whenever a timeout occurs or when the flow is terminated (TCP Fin or Rst received, for example). You can configure the following timers to force a flow export:

- Active timeout—The flow continues to have the packets for the past *m* seconds since the flow was created.
- Inactive timeout—The flow does not have any packets for the past *n* seconds.

Related Topics

Creating a Flow Exporter, on page 1289

Export Formats

The controller supports only NetFlow Version 9 export formats. NetFlow Version 9 export format provides the following features and functionality:

- Variable field specification format
- Support for IPv6, Layer 2, and MPLS fields
- More efficient network utilization

Note For information about the Version 9 export format, see RFC 3954.

Monitors

A monitor references the flow record and flow exporter. You apply a monitor to an interface on the controller.

Note the following when applying a flow monitor to an interface:

- If you apply a flow monitor in the input direction:
  - Use the **match** keyword and use the input interface as a key field.
  - Use the **collect** keyword and use the output interface as a collect field. This field will be present in the exported records but with a value of 0.
- If you apply a flow monitor in the output direction:
  - Use the **match** keyword and use the output interface as a key field.
  - Use the **collect** keyword and use the input interface as a collect field. This field will be present in the exported records but with a value of 0.

Related Topics

Creating a Flow Monitor, on page 1291

Samplers

If you are using sampled mode, you use the sampler to specify the rate at which packets are sampled.

Related Topics

Creating a Sampler, on page 1293

Supported Flexible NetFlow Fields

The following tables provide a consolidated list of supported fields in Flexible NetFlow (FNF) for various traffic types and traffic direction.

Note If the packet has a VLAN field, then that length is not accounted for.

Field	Layer 2 In	Layer 2 Out	IPv4 In	IP v4 Out	IPv6 In	IPv6 Out	Notes
Key or Collect Fields							
Interface input	Yes	—	Yes	—	Yes	—	If you apply a flow monitor in the input direction: • Use the **match** keyword and use the input interface as a key field. • Use the **collect** keyword and use the output interface as a collect field. This field will be present in the exported records but with a value of 0.

Field	Layer 2 In	Layer 2 Out	IPv4 In	IP v4 Out	IPv6 In	IPv6 Out	Notes
Interface output	—	Yes	—	Yes	—	Yes	If you apply a flow monitor in the output direction: • Use the **match** keyword and use the output interface as a key field. • Use the **collect** keyword and use the input interface as a collect field. This field will be present in the exported records but with a value of 0.

Field	Layer 2 In	Layer 2 Out	IPv4 In	IP v4 Out	IPv6 In	IPv6 Out	Notes
Key Fields							
Flow direction	Yes	Yes	Yes	Yes	Yes	Yes	
Ethertype	Yes	Yes	—	—	—	—	
VLAN input	Yes	—	Yes	—	Yes	—	Supported only for a switch port.
VLAN output	—	Yes	—	Yes	—	Yes	Supported only for a switch port.
dot1q VLAN input	Yes	—	Yes	—	Yes	—	Supported only for a switch port.
dot1q VLAN output	—	Yes	—	Yes	—	Yes	Supported only for a switch port.
dot1q priority	Yes	Yes	Yes	Yes	Yes	Yes	Supported only for a switch port.

Field	Layer 2 In	Layer 2 Out	IPv4 In	IP v4 Out	IPv6 In	IPv6 Out	Notes
MAC source address input	Yes	Yes	Yes	Yes	Yes	Yes	
MAC source address output	—	—	—	—	—	—	
MAC destination address input	Yes	—	Yes	—	Yes	—	
MAC destination address output	—	Yes	—	Yes	—	Yes	
IPv4 version	—	—	Yes	Yes	Yes	Yes	
IPv4 TOS	—	—	Yes	Yes	Yes	Yes	
IPv4 protocol	—	—	Yes	Yes	Yes	Yes	Must use if any of src/dest port, ICMP code/type, IGMP type or TCP flags are used.
IPv4 TTL	—	—	Yes	Yes	Yes	Yes	
IPv4 source address	—	—	Yes	Yes	—	—	
IPv4 destination address	—	—	Yes	Yes	—	—	
ICMP IPv4 type	—	—	Yes	Yes	—	—	
ICMP IPv4 code	—	—	Yes	Yes	—	—	

Field	Layer 2 In	Layer 2 Out	IPv4 In	IP v4 Out	IPv6 In	IPv6 Out	Notes
IGMP type	—	—	Yes	Yes	—	—	

Field	Layer 2 In	Layer 2 Out	IPv4 In	IP v4 Out	IPv6 In	IPv6 Out	Notes
Key Fields continued							
IPv6 version	—	—	Yes	Yes	Yes	Yes	Same as IP version.
IPv6 protocol	—	—	Yes	Yes	Yes	Yes	Same as IP protocol. Must use if any of src/dest port, ICMP code/type, IGMP type or TCP flags are used.
IPv6 source address	—	—	—	—	Yes	Yes	
IPv6 destination address	—	—	—	—	Yes	Yes	
IPv6 traffic-class	—	—	Yes	Yes	Yes	Yes	Same as IP TOS.
IPv6 hop-limit	—	—	Yes	Yes	Yes	Yes	Same as IP TTL.
ICMP IPv6 type	—	—	—	—	Yes	Yes	
ICMP IPv6 code	—	—	—	—	Yes	Yes	
source-port	—	—	Yes	Yes	Yes	Yes	
dest-port	—	—	Yes	Yes	Yes	Yes	

Field	Layer 2 In	Layer 2 Out	IPv4 In	IP v4 Out	IPv6 In	IPv6 Out	Notes
Collect Fields							
Bytes long	Yes	Yes	Yes	Yes	Yes	Yes	Packet size = (Ethernet frame size including FCS - 18 bytes) **Recommended:** Avoid this field and use Bytes layer2 long.
Packets long	Yes	Yes	Yes	Yes	Yes	Yes	
Timestamp absolute first	Yes	Yes	Yes	Yes	Yes	Yes	
Timestamp absolute last	Yes	Yes	Yes	Yes	Yes	Yes	
TCP flags	Yes	Yes	Yes	Yes	Yes	Yes	Collects all flags.
Bytes layer2 long	Yes	Yes	Yes	Yes	Yes	Yes	

Default Settings

The following table lists the Flexible NetFlow default settings for the controller.

Table 117: Default Flexible NetFlow Settings

Setting	Default
Flow active timeout	1800 seconds
Flow timeout inactive	Enabled, 15 seconds

How to Configure Flexible NetFlow

To configure Flexible NetFlow, follow these general steps:

1. Create a flow record by specifying keys and non-key fields to the flow.
2. Create an optional flow exporter by specifying the protocol and transport destination port, destination, and other parameters.
3. Create a flow monitor based on the flow record and flow exporter.
4. Create a sampler.
5. Apply the flow monitor to a Layer 2 port, Layer 3 port, or VLAN.
6. If applicable to your configuration, configure a WLAN to apply a flow monitor to.

Creating a Flow Record

You can create a flow record and add keys to match on and fields to collect in the flow.

SUMMARY STEPS

1. **configure terminal**
2. **flow record** *name*
3. **description** *string*
4. **match** *type*
5. **collect** *type*
6. **end**
7. **show flow record** [**name** *record-name*]
8. **copy running-config startup-config**

DETAILED STEPS

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

Step 2	**flow record** *name*
	
	Example:
	
	`Controller(config)# flow record test`
	`Controller(config-flow-record)#`
		Creates a flow record and enters flow record configuration mode.

	Command or Action	Purpose
Step 3	**description** *string* **Example:** `Controller(config-flow-record)# description Ipv4Flow`	(Optional) Describes this flow record as a maximum 63-character string.

Step 4	**match** *type*
	
	Example:
	
	`Controller(config-flow-record)# match ipv4 source address`
	`Controller(config-flow-record)# match ipv4 destination address`
	`Controller(config-flow-record)# match flow direction`
		Specifies a match key. For information about possible match key values, see Flexible NetFlow Match Parameters, on page 1277.

Step 5	**collect** *type*
	
	Example:
	
	`Controller(config-flow-record)# collect counter bytes layer2 long`
	`Controller(config-flow-record)# collect counter bytes long`
	`Controller(config-flow-record)# collect timestamp absolute first`
	`Controller(config-flow-record)# collect transport tcp flags`
		Specifies the collection field. For information about possible collection field values, see Flexible NetFlow Collect Parameters, on page 1279.

Step 6	**end** **Example:** `Controller(config-flow-record)# end`	Returns to privileged EXEC mode.
Step 7	**show flow record** [**name** *record-name*] **Example:** `Controller show flow record test`	(Optional) Displays information about NetFlow flow records.
Step 8	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

What to Do Next

Define an optional flow exporter by specifying the export format, protocol, destination, and other parameters.

Related Topics

Flow Records, on page 1277

Creating a Flow Exporter

You can create a flow export to define the export parameters for a flow.

SUMMARY STEPS

1. **configure terminal**
2. **flow exporter** *name*
3. **description** *string*
4. **dscp** *value*
5. **destination** { *ipv4-address* }
6. **source** { *source type* }
7. **transport udp** *number*
8. **end**
9. **show flow exporter** [**name** *record-name*]
10. **copy running-config startup-config**

DETAILED STEPS

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

Step 2	**flow exporter** *name*
	
	Example:
	
	`Controller(config)# flow exporter ExportTest`
	`Controller (config-flow-exporter)#`
		Creates a flow exporter and enters flow exporter configuration mode.

Step 3	**description** *string*
	
	Example:
	
	`Controller(config-flow-exporter)#`
	`description ExportV9`
		(Optional) Describes this flow record as a maximum 63-character string.

	Command or Action	Purpose
Step 4	**dscp** *value* **Example:** `Controller(config-flow-exporter)# dscp 0`	(Optional) Specifies the differentiated services codepoint value. The range is from 0 to 63.
Step 5	**destination** { *ipv4-address* } **Example:** `Controller(config-flow-exporter)# destination 192.0.2.1`	Sets the destination IPv4 address or hostname for this exporter.
Step 6	**source** { *source type* } **Example:** `Controller(config-flow-exporter)# source gigabitEthernet1/0/1`	Specifies the interface to use to reach the NetFlow collector at the configured destination. The following interfaces can be configured as source: • **Auto Template—**Auto-Template interface • **Capwap—**CAPWAP tunnel interface • **GigabitEthernet—**Gigabit Ethernet IEEE 802 • **GroupVI—**Group virtual interface • **Internal Interface—**Internal interface • **Loopback—**Loopback interface • **Null—**Null interface • **Port-channel—**Ethernet Channel of interface • **TenGigabitEthernet—**10-Gigabit Ethernet • **Tunnel—**Tunnel interface • **Vlan—**Catalyst VLANs
Step 7	**transport udp** *number* **Example:** `Controller(config-flow-exporter)# transport udp 200`	(Optional) Specifies the UDP port to use to reach the NetFlow collector. The range is from 0 to 65535.
Step 8	**end** **Example:** `Controller(config-flow-record)# end`	Returns to privileged EXEC mode.
Step 9	**show flow exporter** [**name** *record-name*]	(Optional) Displays information about NetFlow flow exporters.

	Command or Action	Purpose
	Example: `Controller show flow exporter ExportTest`	
Step 10	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

What to Do Next

Define a flow monitor based on the flow record and flow exporter.

Related Topics

Exporters, on page 1280

Creating a Flow Monitor

You can create a flow monitor and associate it with a flow record and a flow exporter.

SUMMARY STEPS

1. **configure terminal**
2. **flow monitor** *name*
3. **description** *string*
4. **exporter** *name*
5. **record** *name*
6. **cache** { **timeout** {**active** | **inactive**} *seconds* | **type normal** }
7. **end**
8. **show flow monitor** [**name** *record-name*]
9. **copy running-config startup-config**

DETAILED STEPS

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

Step 2	**flow monitor** *name*
	
	Example:
	
	`Controller(config)# flow monitor MonitorTest`
	`Controller (config-flow-monitor)#`
		Creates a flow monitor and enters flow monitor configuration mode.

Step 3	**description** *string* **Example:** `Controller(config-flow-monitor)# description Ipv4Monitor`	(Optional) Describes this flow record as a maximum 63-character string.
Step 4	**exporter** *name* **Example:** `Controller(config-flow-monitor)# exporter ExportTest`	Associates a flow exporter with this flow monitor.
Step 5	**record** *name* **Example:** `Controller(config-flow-monitor)# record test`	Associates a flow record with the specified flow monitor.
Step 6	**cache** { **timeout** {**active** | **inactive**} *seconds* | **type normal** } **Example:** `Controller(config-flow-monitor)# cache timeout active 15000`	Associates a flow cache with the specified flow monitor.
Step 7	**end** **Example:** `Controller(config-flow-monitor)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 8	**show flow monitor** [**name** *record-name*] **Example:** `Controller show flow monitor name MonitorTest`	(Optional) Displays information about NetFlow flow monitors.
Step 9	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

What to Do Next

Apply the flow monitor to a Layer 2 interface, Layer 3 interface, or VLAN.

Related Topics

Monitors, on page 1281

Creating a Sampler

You can create a sampler to define the NetFlow sampling rate for a flow.

SUMMARY STEPS

1. **configure terminal**
2. **sampler** *name*
3. **description** *string*
4. **mode** {**random**}
5. **end**
6. **show sampler** [*name*]
7. **copy running-config startup-config**

DETAILED STEPS

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

	Command or Action	Purpose

Step 2	**sampler** *name*
	
	Example:
	`Controller(config)# sampler SampleTest`
	`Controller(config-flow-sampler)#`
		Creates a sampler and enters flow sampler configuration mode.

Step 3	**description** *string* **Example:** `Controller(config-flow-sampler)# description samples`	(Optional) Describes this flow record as a maximum 63-character string.
Step 4	**mode** {**random**} **Example:** `Controller(config-flow-sampler)# mode random 1 out-of 1024`	Defines the random sample mode.
Step 5	**end** **Example:** `Controller(config-flow-sampler)# end`	Returns to privileged EXEC mode.
Step 6	**show sampler** [*name*] **Example:** `Controller show sample SampleTest`	(Optional) Displays information about NetFlow samplers.
Step 7	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

What to Do Next

Apply the flow monitor to a source interface, subinterface, VLAN interface, or a VLAN.

Related Topics

Samplers, on page 1282

Applying a Flow to an Interface

You can apply a flow monitor and an optional sampler to an interface.

Note the following when applying a flow monitor to an interface:

- If you apply a flow monitor in the input direction:
  - Use the **match** keyword and use the input interface as a key field.
  - Use the **collect** keyword and use the output interface as a collect field. This field will be present in the exported records but with a value of 0.
- If you apply a flow monitor in the output direction:
  - Use the **match** keyword and use the output interface as a key field.
  - Use the **collect** keyword and use the input interface as a collect field. This field will be present in the exported records but with a value of 0.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *type*
3. **ip flow monitor** *name* [**sampler** *name*] { **input** |**output** }
4. **end**
5. **show flow interface** [*interface-type number*]
6. **copy running-config startup-config**

DETAILED STEPS

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

Step 2	**interface** *type*
	
	Example:
	
	`Controller(config)# interface GigabitEthernet1/0/1`
	`Controller(config-if)#`
		Enters interface configuration mode and configures an interface.
		Command parameters for the interface configuration include:
		• **Auto—** Auto-Template interface
		• **Capwap—**CAPWAP tunnel interface
		• **GigabitEthernet—**GigabitEthernet IEEE 802
		• **GroupVI—**Group Virtual interface
		• **Internal Interface—**Internal Interface
		• **Loopback—**Loopback interface
		• **Null—**Null interface
		• **Port-channel—**Ethernet channel of interface
		• **TenGigabitEthernet—**10- Gigabit Ethernet

	Command or Action	Purpose
		• **Tunnel—**Tunnel interface • **Vlan—**Catalyst VLANs • **Range—**Interface range
Step 3	**ip flow monitor** *name* [**sampler** *name*] { **input** |**output** } **Example:** `Controller(config-if)# ip flow monitor MonitorTest input`	Associates an IPv4 flow monitor and an optional sampler to the interface for input or output packets.
Step 4	**end** **Example:** `Controller(config-flow-monitor)# end`	Returns to privileged EXEC mode.
Step 5	**show flow interface** [*interface-type number*] **Example:** `Controller# show flow interface`	(Optional) Displays information about NetFlow on an interface.
Step 6	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring a Bridged NetFlow on a VLAN

You can apply a flow monitor and an optional sampler to a VLAN.

SUMMARY STEPS

1. **configure terminal**
2. **vlan** [**configuration**] *vlan-id*
3. **ip flow monitor** *name* [**sampler** *name*] {**input** |**output**}
4. **copy running-config startup-config**

DETAILED STEPS

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

Step 2	**vlan** [**configuration**] *vlan-id*
	
	Example:
	
	`Controller(config)# vlan configuration 30`
	`Controller(config-vlan-config)#`
		Enters VLAN or VLAN configuration mode.

Step 3	**ip flow monitor** *name* [**sampler** *name*] {**input** |**output**} **Example:** `Controller(config-vlan-config)# ip flow monitor MonitorTest input`	Associates a flow monitor and an optional sampler to the VLAN for input or output packets.
Step 4	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring Layer 2 NetFlow

You can define Layer 2 keys in Flexible NetFlow records that you can use to capture flows in Layer 2 interfaces.

SUMMARY STEPS

1. **configure terminal**
2. **flow record** *name*
3. **match datalink** {**dot1q** |**ethertype** | **mac** | **vlan**}
4. **end**
5. **show flow record** [*name*]
6. **copy running-config startup-config**

DETAILED STEPS

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

Step 2	**flow record** *name*
	
	Example:
	`Controller(config)# flow record L2_record`
	`Controller(config-flow-record)#`
		Enters flow record configuration mode.

Step 3	**match datalink** {**dot1q** |**ethertype** | **mac** | **vlan**} **Example:** `Controller(config-flow-record)# match datalink ethertype`	Specifies the Layer 2 attribute as a key.
Step 4	**end** **Example:** `Controller(config-flow-record)# end`	Returns to privileged EXEC mode.
Step 5	**show flow record** [*name*] **Example:** `Controller# show flow record`	(Optional) Displays information about NetFlow on an interface.
Step 6	**copy running-config startup-config** **Example:** `Controller# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Configuring WLAN to Apply Flow Monitor in Data Link Input/Output Direction

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan-name*
3. **datalink flow monitor monitor-name {input | output}**
4. **end**
5. **show wlan** *wlan-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *wlan-name* **Example:** `Controller (config) # wlan mywlan`	Enters WLAN configuration submode. For *wlan-name*, enter the profile name. The range is 1 to 32 characters.
Step 3	**datalink flow monitor monitor-name {input | output}** **Example:** `Controller (config-wlan) # datalink flow monitor flow-monitor-1 {input | output}`	Applies flow monitor to Layer 2 traffic in the direction of interest.
Step 4	**end** **Example:** `Controller (config) # end`	Returns to privileged EXEC mode.
Step 5	**show wlan** *wlan-name* **Example:** `Controller # show wlan mywlan`	(Optional) Verifies your configuration.

Configuring WLAN to Apply Flow Monitor in IPV4 and IPv6 Input/Output Direction

SUMMARY STEPS

1. **configure terminal**
2. **wlan** *wlan-id*
3. **{ipv4 |ipv6} flow monitor monitor-name {input | output}**
4. **end**
5. **show wlan** *wlan-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *wlan-id* **Example:** `Controller (config) # wlan 1`	Enters WLAN configuration submode. For *wlan-id*, enter the WLAN ID. The range is 1 to 64.
Step 3	**{ipv4 |ipv6} flow monitor monitor-name {input | output}** **Example:** `Controller (config-wlan) # ip flow monitor flow-monitor-1 input`	Associates a flow monitor to the WLAN for input or output packets.
Step 4	**end** **Example:** `Controller (config) # end`	Returns to privileged EXEC mode.
Step 5	**show wlan** *wlan-name* **Example:** `Controller # show wlan mywlan`	(Optional) Verifies your configuration.

Related Topics

Wireless Flexible NetFlow Overview, on page 1276

Example: Configuring IPv4 Flexible NetFlow in WLAN (Ingress Direction), on page 1302

Example: Configuring IPv6 and Transport Flag Flexible NetFlow in WLAN (Egress Direction), on page 1302
Example: Configuring IPv6 Flexible NetFlow in WLAN (Both Ingress and Egress Directions), on page 1303

Monitoring Flexible NetFlow

The commands in the following table can be used to monitor Flexible NetFlow.

Table 118: Flexible NetFlow Monitoring Commands

Command	Purpose
show flow exporter [**broker** | **export-ids** | **name** | *name* | **statistics** | **templates**]	Displays information about NetFlow flow exporters and statistics.
show flow exporter [**name** *exporter-name*]	Displays information about NetFlow flow exporters and statistics.
show flow interface	Displays information about NetFlow interfaces.
show flow monitor [**name** *exporter-name*]	Displays information about NetFlow flow monitors and statistics.
show flow record [**name** *record-name*]	Displays information about NetFlow flow records.
show flow ssid	Displays NetFlow monitor installation status for a WLAN.
show sampler [**broker** | **name** | *name*]	Displays information about NetFlow samplers.
show wlan *wlan-name*	Displays the WLAN configured on the device.

Configuration Examples for Flexible NetFlow

Example: Configuring a Flow

This example shows how to create a flow and apply it to an interface:

```
Controller# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.

Controller(config)# flow export export1
Controller(config-flow-exporter)# destination 10.0.101.254
Controller(config-flow-exporter)# transport udp 2055
Controller(config-flow-exporter)# exit
Controller(config)# flow record record1
Controller(config-flow-record)# match ipv4 source address
```

```
Controller(config-flow-record)# match ipv4 destination address
Controller(config-flow-record)# match ipv4 protocol
Controller(config-flow-record)# match transport source-port
Controller(config-flow-record)# match transport destination-port
Controller(config-flow-record)# collect counter byte long
Controller(config-flow-record)# collect counter packet long
Controller(config-flow-record)# collect timestamp absolute first
Controller(config-flow-record)# collect timestamp absolute last
Controller(config-flow-record)# exit
Controller(config)# flow monitor monitor1
Controller(config-flow-monitor)# record record1
Controller(config-flow-monitor)# exporter export1
Controller(config-flow-monitor)# exit
Controller(config)# interface tenGigabitEthernet 1/0/1
Controller(config-if)# ip flow monitor monitor1 input
Controller(config-if)# end
```

Example: Configuring IPv4 Flexible NetFlow in WLAN (Ingress Direction)

The following example shows how to configure IPv4 Flexible NetFlow on WLAN ingress direction:

```
Controller# configure terminal
Controller(config)# flow record fr_v4
Controller(config-flow-record)# match ipv4 destination
Controller(config-flow-record)# match ipv4 source
Controller(config-flow-record)# match ipv4 protocol
Controller(config-flow-record)# match ipv4 tos
Controller(config-flow-record)# match ipv4 ttl
Controller(config-flow-record)# match ipv4 version
Controller(config-flow-record)# collect counter packets long
Controller(config-flow-record)# collect counter bytes long
Controller(config-flow-record)# collect timestamp sys-uptime first
Controller(config-flow-record)# collect timestamp sys-uptime last
Controller(config-flow-record)# exit

Controller(config)# flow monitor fm_v4
Controller(config-flow-monitor)# record fr_v4
Controller(config-flow-record)# exit

Controller(config)# wlan 1
Controller(config-wlan)# ip flow monitor fm_v4 in
Controller(config-wlan)# end

Controller# show flow monitor fm_v4 cache
```

Related Topics

Configuring WLAN to Apply Flow Monitor in IPV4 and IPv6 Input/Output Direction, on page 1300
Wireless Flexible NetFlow Overview, on page 1276

Example: Configuring IPv6 and Transport Flag Flexible NetFlow in WLAN (Egress Direction)

The following example shows how to configure IPv6 and transport flag Flexible NetFlow on WLAN egress direction:

```
Controller# configure terminal
Controller(config)# flow record fr_v6
Controller(config-flow-record)# match ipv6 destination
Controller(config-flow-record)# match ipv6 source
```

```
Controller(config-flow-record)# match ipv6 hop-limit
Controller(config-flow-record)# match ipv6 protocol
Controller(config-flow-record)# match ipv6 traffic class
Controller(config-flow-record)# match ipv6 version
Controller(config-flow-record)# collect counter bytes long
Controller(config-flow-record)# collect transport tcp flags
Controller(config-flow-record)# exit

Controller(config)# flow monitor fm_v6
Controller(config-flow-monitor)# record fr_v6
Controller(config-flow-monitor)# exit

Controller(config)# wlan 1
Controller(config-wlan)# ipv6 flow monitor fm_v6 out
Controller(config-wlan)# end

Controller# show flow monitor fm_v6 cache
```

Note On the controller, you cannot specify which TCP flag to collect. You can only specify to collect transport TCP flags.

Related Topics

Configuring WLAN to Apply Flow Monitor in IPV4 and IPv6 Input/Output Direction, on page 1300
Wireless Flexible NetFlow Overview, on page 1276

Example: Configuring IPv6 Flexible NetFlow in WLAN (Both Ingress and Egress Directions)

The following example shows how to configure IPv6 Flexible NetFlow on WLAN in both directions:

```
Controller# configure terminal
Controller (config)# flow record fr_v6
Controller (config-flow-record)# match ipv6 destination
Controller (config-flow-record)# match ipv6 source
Controller (config-flow-record)# match ipv6 hop-limit
Controller (config-flow-record)# match ipv6 protocol
Controller (config-flow-record)# match ipv6 traffic class
Controller (config-flow-record)# match ipv6 version
Controller (config-flow-record)# collect counter packets long
Controller (config-flow-record)# exit

Controller (config)# flow monitor fm_v6
Controller (config-flow-monitor)# record fr_v6
Controller (config-flow-monitor)# exit

Controller (config)# wlan 1
Controller (config-wlan)# ipv6 flow monitor fm_v6 in
Controller (config-wlan)# ipv6 flow monitor fm_v6 out
Controller (config-wlan)# end

Controller# show flow monitor fm_v6 cache
```

Related Topics

Configuring WLAN to Apply Flow Monitor in IPV4 and IPv6 Input/Output Direction, on page 1300
Wireless Flexible NetFlow Overview, on page 1276

Additional References

Related Documents

Related Topic	Document Title
Flexible NetFlow CLI Commands	*Flexible NetFlow Command Reference, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)*

Standards and RFCs

Standard/RFC	Title
RFC 3954	Cisco Systems NetFlow Services Export Version 9

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

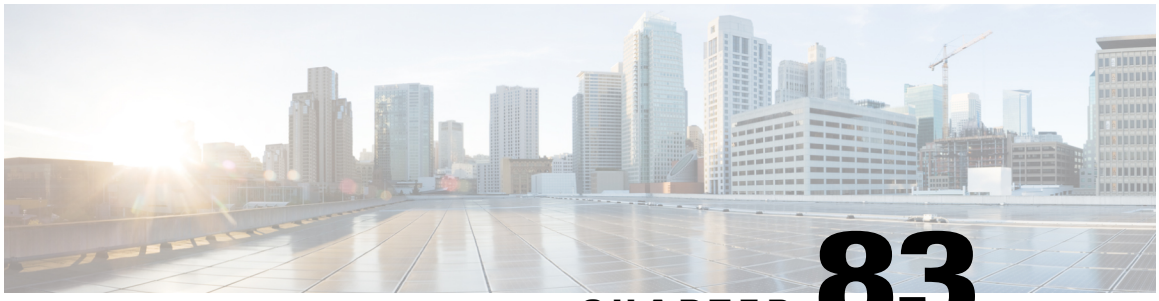
Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, 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 Information for Flexible NetFlow

Feature Name	Releases	Feature Information
Flexible NetFlow feature support		Flexible NetFlow uses flows to provide statistics for accounting, network monitoring, and network planning.

PART XVI

High Availability



CHAPTER 83

High Availability

This chapter contains the following sections:

High Availability

Cisco high availability technology helps to facilitate network-wide resilience to increase IP network availability. It provides continuous access to applications, data, and content anywhere, anytime by addressing potential causes of downtime with functionality, design, and best practices.

This module provides information on the theory of operation for Cisco 5700 Series Wireless Controller as it supports stateful switchover of access points (AP SSO). In addition to AP SSO, the chapter also provides information on the ISSU high availability feature. These features reduce major downtime in wireless networks due to failure conditions that may occur due to box failover or network failover.

In Cisco 5700 Series Wireless Controller a high availability SKU AIR-CT5760-HA-K9 is available to deploy as a redundant N+1 controller in case the primary controller goes down. You do not need to purchase duplicate AP scale licensing on the redundant controller.

In Cisco 5700 Series Wireless Controller the high availability feature is enabled by default and you will not be able to disable it. You can only initiate a manual graceful-switchover using the command line interface.

During an AP SSO, all the AP sessions are switched over statefully and all the clients are deauthenticated and reassociated with the new active controller except for the locally switched clients in FlexConnect mode.

Before you enable HA (AP SSO), ensure that both controllers are physically connected through the redundant port using an Ethernet cable. Also, ensure that the uplink is connected to an infrastructure switch and that the gateway is reachable from both the controllers.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 Switchover

This section lists the restrictions that you should keep in mind while performing a switchover from the active unit to the standby unit:

- Verify that the unit has been reloaded using the stack manager.
- Verify if the logging console is enabled.
- Verify the progression failures. The progression is bound by a time unit of 30 seconds.
- Verify if no process failures occur.
- Verify if any congestion occurs in the up coming unit.

Post Switchover Tasks

- This section defines the steps that you must perform to ensure that successful switchover from the active to standby switch is performed. On successful switchover of the standby switch as active, all access points connected to the active need to re-join the standby (then active) switch.

SUMMARY STEPS

1. **show ap uptime**
2. **show wireless summary**
3. **show wcdb database all**
4. **show power inline**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show ap uptime**	Verify the uptime of the access point uptime after the switchover is large enough
Step 2	**show wireless summary**	Display the clients connected in the active switch.
Step 3	**show wcdb database all**	Display if the client has reached the uptime.
Step 4	**show power inline**	Display the power over Ethernet power state.

Information on Mobility

On switchover, the mobility states will be removed from the mobility domain. Once the switchover is complete, the MC and MA messages are sent to the peer nodes. These messages aid in identifying the status of the peers in the network. Each switch is part of the stack, and the MA and MC software is available on both the active and standby switches. The mobility control session up and down is identified using the keep alive messages. The details of the mobility client database data synchronization between the active and standby is performed. For more details on mobility, see the Cisco

You must remove the following when client SSO is supported as the switch-over event in this release is AP SSO:

- The switch-over notification from mobility controller to mobility oracle and to all the mobility anchors in the same group.
- Mobility anchors in the sub-domain that deletes all the client.
- The mobility oracle deletes clients with switchover mobility controller as foreign or anchor.
- The other mobility agents of the group cleans up the clients that are associated to the switchover mobility agent.
- Delete client messages to the mobility oracle.

Debugging Mobility before the Switchover

This section provides steps on how you can debug mobility before the switch-over.

SUMMARY STEPS

1. **show etherchannel summary**
2. **show platform etherchannel**
3. **show wireless mobility summary**
4. **show wireless mobility dtls connections**
5. **show mgmt-infra trace messages devshell**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show etherchannel summary**	Display port channel summary.
Step 2	**show platform etherchannel**	Display the states of the platform port channel.
Step 3	**show wireless mobility summary**	Display the mobility link state.
Step 4	**show wireless mobility dtls connections**	Display the mobility DTLS connection table.
Step 5	**show mgmt-infra trace messages devshell**	Display the mobility DTLS lookup table.

Debugging Mobility after the Switchover

This section provides steps on how you can debug mobility after the switchover. You must perform the same steps displayed in the debug mobility before the switchover to debug the mobility after switchover.

Information About Radio Resource Management

This section provides the theory of operations for the Radio Resource Management (RRM) feature in this release. RRM allows Cisco's Unified WLAN Architecture to continuously analyze the existing RF environment, automatically adjusting APs' power levels and channel configurations to help mitigate such things as co-channel interference and signal coverage problems. RRM reduces the need to perform exhaustive site surveys, increases system capacity, and provides automated self-healing functionality to compensate for RF dead zones and AP failures.

During a switchover, the AP's Radio Slot Data, RF measurement, CleanAir, and RF grouping state is synchronized to the standby switch during AP SSO. The standby switch then does not participate in any RF grouping unless it is in active state. The standby unit participates in the RF group as a new member. During a switchover, incremental data synchronization of data from active switch to standby switch is performed.

Information on Security

Wireless security is the prevention of unauthorized access or damage to computers using wireless networks. In Cisco Wireless Controller 5700 series, you can:

- Configure IEEE 802.1x global configurations.
- Configure Locally Significant Certificates.
- Configure strong password enforcement options.
- Enable over-the-air frame padding.
- Configure access point neighbor authentication.
- Enable protection from Denial of Service (DoS) attacks.

- Configure Intrusion Detection System (IDS) sensors for the Wireless Protection System (WPS).
- Configure client exclusion policies.
- Configure Management Frame Protection (MFP).
- Enforce the controller to synchronize with other controllers in the mobility group for the shun list.

On switchover, the following security features continue to be up and running:

- Datagram Transport Layer Security (DTLS) protocol for CAPWAP and mobility.
- Infrastructure Management Frame.
- Rogue access points and client detection.
- The access point list in the controller continues to work by re-learning.
- Synchronization of the Intrusion Detection System (IDS)/ Shun List in the mobility anchor and mobility client.
- The Wireless Intrusion Prevention System (wIPS) continues to work by re-learning and configuration methods.
- Synchronization of pairwise master key to enable faster re-association of the wireless clients.

Information on Location and Certificate Management

The states of the TCP port (16113) that the controller and mobility services engine communicate are not synchronized. So, once a switchover occurs, the client must re-associate to the location server. You must also manually download the LSC certificate and install in both the units.

-

Information on CAPWAP, Multicast, and CDP

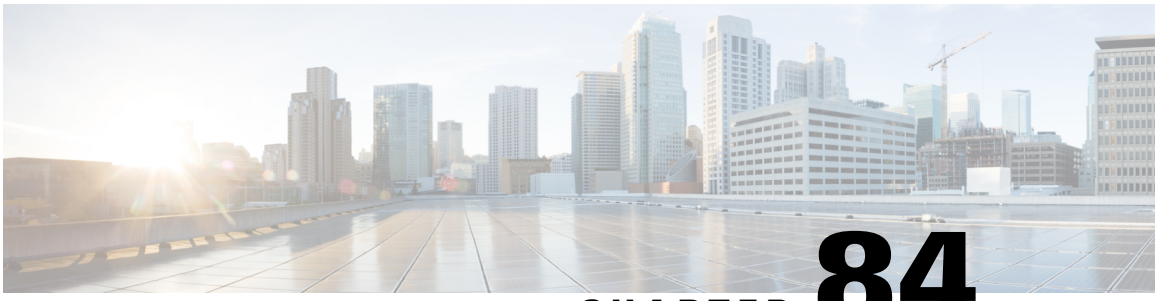
In a CAPWAP environment, a lightweight access point discovers a controller by using CAPWAP discovery mechanisms and then sends the controller a CAPWAP join request. The controller sends the access point a CAPWAP join response allowing the access point to join the controller. When the access point joins the controller, the controller manages its configuration, firmware, control transactions, and data transactions. On switch-over, all the clients associated need to de-associate. The following occurs, after a switchover:

- The CAPWAP control message sequence number continues to be functional.
- The multicast protocol functions by re-learning the routes.
- The CDP protocol discovers all network elements in the network, and the access point sends the CDP update.
- Event log of the access point will not be synchronized during the switchover.

Information on Voice and QoS

QoS enables you to provide preferential treatment to specific types of traffic at the expense of other traffic types. Without QoS, the controller offers best-effort service to each packet, regardless of the packet contents r size. On switchover, the voice and QoS statistics values displayed in the controller will be reset. This implies that the Voice and QoS statistics are not synchronized while switchover happens from active unit to standby unit.

-
-



CHAPTER 84

Configuring Cisco NSF with SSO

How to configure Cisco NSF with SSO

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 NSF with SSO

You must ensure the following before configuring NSF with SSO.

- Use of the routing protocols requires the IP Services license level. EIGRP-stub and OSPF for routed access are supported on IP Base license level.
- BGP support in NSF requires that neighbor networking devices be NSF-aware; that is, the devices must have the graceful restart capability and advertise that capability in their OPEN message during session establishment. If an NSF-capable router discovers that a particular BGP neighbor does not have graceful restart capability, it does not establish an NSF-capable session with that neighbor. All other neighbors that have graceful restart capability continue to have NSF-capable sessions with this NSF-capable networking device.
- OSPF support in NSF requires that all neighbor networking devices be NSF-aware. If an NSF-capable router discovers that it has non-NSF -aware neighbors on a particular network segment, it disables NSF capabilities for that segment. Other network segments composed entirely of NSF-capable or NSF-aware routers continue to provide NSF capabilities.
- OSPF support in NSF requires that all neighbor networking devices be NSF-aware. If an NSF-capable router discovers that it has non-NSF -aware neighbors on a particular network segment, it disables NSF

capabilities for that segment. Other network segments composed entirely of NSF-capable or NSF-aware routers continue to provide NSF capabilities.

Restrictions for NSF with SSO

- NSF does not support IPv6 and is IPv4 Unicast only.
- SSO is not supported if the IOS-XE software is running in the LAN Base mode.
- For NSF operation, you must have SSO configured on the device.
- NSF with SSO supports IP Version 4 traffic and protocols only; NSF with SSO does not support IPv6 traffic.
- The Virtual Redundancy Routing Protocols (VRRP) is not SSO-aware, meaning state information is not maintained between the active and standby switches during normal operation. VRRP and SSO can coexist but both features work independently. Traffic that relies on VRRP may switch to the VRRP standby in the event of an active switch switchover.
- All neighboring devices participating in BGP NSF must be NSF-capable and configured for BGP graceful restart.
- All OSPF networking devices on the same network segment must be NSF-aware (running an NSF software image).
- For IETF, all neighboring devices must be running an NSF-aware software image.

Information About NSF with SSO Supervisor Engine Active Switch Redundancy

Overview of NSF with SSO

The switch supports fault resistance by allowing a standby switch to take over if the active switch becomes unavailable. Cisco nonstop forwarding (NSF) works with stateful switchover (SSO) to minimize the amount of time a network is unavailable

NSF provides these benefits:

- Improved network availability

  NSF continues forwarding network traffic and application state information so that user session information is maintained after a switchover.

- Overall network stability

  Network stability may be improved with the reduction in the number of route flaps, which were created when routers in the network failed and lost their routing tables.

- Neighboring routers do not detect a link flap

  Because the interfaces remain up during a switchover, neighboring routers do not detect a link flap (the link does not go down and come back up).

- Prevents routing flaps

  Because SSO continues forwarding network traffic during a switchover, routing flaps are avoided.

- Maintains user sessions established prior to the switchover

SSO Operation

When a standby switch runs in SSO mode, the standby switch starts up in a fully-initialized state and synchronizes with the persistent configuration and the running configuration of the active switch. It subsequently maintains the state on the protocols listed below, and all changes in hardware and software states for features that support stateful switchover are kept in synchronization. Consequently, it offers minimum interruption to Layer 2 sessions in a redundant active switch configuration.

If the active switch fails, the standby switch becomes the active switch. This new active switch uses existing Layer 2 switching information to continue forwarding traffic. Layer 3 forwarding will be delayed until the routing tables have been repopulated in the newly active switch.

Note SSO is not supported if the IOS-XE software is running the LAN Base license level.

The state of these features is preserved between both the active and standby switches:

- 802.3
- 802.3u
- 802.3x (Flow Control)
- 802.3ab (GE)
- 802.3z (Gigabit Ethernet including CWDM)
- 802.3ad (LACP)
- 802.1p (Layer 2 QoS)
- 802.1q
- 802.1X (Authentication)
- 802.1D (Spanning Tree Protocol)
- 802.3af (Inline power)
- PAgP
- VTP
- Dynamic ARP Inspection
- DHCP snooping
- IP source guard
- IGMP snooping (versions 1 and 2)
- DTP (802.1q and ISL)
- MST
- PVST+
- Rapid-PVST
- PortFast/UplinkFast/BackboneFast
- BPDU guard and filtering

- Voice VLAN
- Port security
- Unicast MAC filtering
- ACL (VACLS, PACLS, RACLS)
- QOS (DBL)
- Multicast storm control/broadcast storm control

SSO is compatible with the following list of features. However, the protocol database for these features is not synchronized between the standby and active switches:

- 802.1Q tunneling with Layer 2 Protocol Tunneling (L2PT)
- Baby giants
- Jumbo frame support
- CDP
- Flood blocking
- UDLD
- SPAN/RSPAN
- NetFlow

The following features are learned on the standbyswitch if the SSO feature is enabled:

- All Layer 3 protocols on switch.

NSF Operation

Cisco IOS Nonstop Forwarding (NSF) always runs with stateful switchover (SSO) and provides redundancy for Layer 3 traffic. NSF is supported by the BGP, OSPF, and EIGRP routing protocols and is supported by Cisco Express Forwarding (CEF) for forwarding. The routing protocols have been enhanced with NSF-capability and awareness, which means that routers running these protocols can detect a switchover and take the necessary actions to continue forwarding network traffic and to recover route information from the peer devices.

Each protocol depends on CEF to continue forwarding packets during switchover while the routing protocols rebuild the Routing Information Base (RIB) tables. After the routing protocols have converged, CEF updates the FIB table and removes stale route entries. CEF then updates the hardware with the new FIB information.

If the active switch is configured for BGP (with the graceful-restart command), OSPF, or EIGRP routing protocols, routing updates are automatically sent during the active switch election

The switch supports NSF-awareness and NSF-capability for the BGP, OSPF, and EIGRP protocols in IP Services license level and NSF-awareness for the EIGRP-stub in IP Base license level.

NSF has two primary components:

- NSF-awareness

  A networking device is NSF-aware if it is running NSF-compatible software. If neighboring router devices detect that an NSF router can still forward packets when an active switch election happens, this capability is referred to as NSF-awareness. Cisco IOS enhancements to the Layer 3 routing protocols

(BGP, OSPF, and EIGRP) are designed to prevent route-flapping so that the CEF routing table does not time out or the NSF router does not drop routes. An NSF-aware router helps to send routing protocol information to the neighboring NSF router. NSF-awareness is enabled by default for EIGRP-stub, EIGRP, and OSPF protocols. NSF-awareness is disabled by default for BGP.

- NSF-capability

  A device is NSF-capable if it has been configured to support NSF; it rebuilds routing information from NSF-aware or NSF-capable neighbors. NSF works with SSO to minimize the amount of time that a Layer 3 network is unavailable following an active switch election by continuing to forward IP packets. Reconvergence of Layer 3 routing protocols (BGP, OSPFv2, and EIGRP) is transparent to the user and happens automatically in the background. The routing protocols recover routing information from neighbor devices and rebuild the Cisco Express Forwarding (CEF) table.

Note NSF does not support IPv6 and is IPv4 Unicast only.

Cisco Express Forwarding

A key element of Cisco IOS Nonstop Forwarding (NSF) is packet forwarding. In a Cisco networking device, packet forwarding is provided by Cisco Express Forwarding (CEF). CEF maintains the FIB and uses the FIB information that was current at the time of the switchover to continue forwarding packets during a switchover. This feature reduces traffic interruption during the switchover.

During normal NSF operation, CEF on the active supervisor switch synchronizes its current FIB and adjacency databases with the FIB and adjacency databases on the standby switch. Upon switchover, the standby switch initially has FIB and adjacency databases that are mirror images of those that were current on the active switch. CEF keeps the forwarding engine on the standby switch current with changes that are sent to it by CEF on the active switch. The forwarding engine can continue forwarding after a switchover as soon as the interfaces and a data path are available.

As the routing protocols start to repopulate the RIB on a prefix-by-prefix basis, the updates cause prefix-by-prefix updates to CEF, which it uses to update the FIB and adjacency databases. Existing and new entries receive the new version (“epoch”) number, indicating that they have been refreshed. The forwarding information is updated on the forwarding engine during convergence. The switch signals when the RIB has converged. The software removes all FIB and adjacency entries that have an epoch older than the current switchover epoch. The FIB now represents the newest routing protocol forwarding information.

BGP Operation

When an NSF-capable router begins a BGP session with a BGP peer, it sends an OPEN message to the peer. Included in the message is a statement that the NSF-capable device has “graceful” restart capability. Graceful restart is the mechanism by which BGP routing peers avoid a routing flap following a switchover. If the BGP peer has received this capability, it is aware that the device sending the message is NSF-capable. Both the NSF-capable router and its BGP peers need to exchange the graceful restart capability in their OPEN messages at the time of session establishment. If both the peers do not exchange the graceful restart capability, the session will not be capable of a graceful restart.

If the BGP session is lost during the active switch switchover, the NSF-aware BGP peer marks all the routes associated with the NSF-capable router as stale; however, it continues to use these routes to make forwarding

decisions for a set period of time. This functionality prevents packets from being lost while the newly active switch is waiting for convergence of the routing information with the BGP peers.

After an active switch switchover occurs, the NSF-capable router reestablishes the session with the BGP peer. In establishing the new session, it sends a new graceful restart message that identifies the NSF-capable router as having restarted.

At this point, the routing information is exchanged between the two BGP peers. After this exchange is complete, the NSF-capable device uses the routing information to update the RIB and the FIB with the new forwarding information. The NSF-aware device uses the network information to remove stale routes from its BGP table; the BGP protocol then is fully converged.

If a BGP peer does not support the graceful restart capability, it ignores the graceful restart capability in an OPEN message but establishes a BGP session with the NSF-capable device. This function allows interoperability with non-NSF-aware BGP peers (and without NSF functionality), but the BGP session with non-NSF-aware BGP peers is not capable of a graceful restart.

Note BGP support in NSF requires that neighbor networking devices be NSF-aware; that is, the devices must have the graceful restart capability and advertise that capability in their OPEN message during session establishment. If an NSF-capable router discovers that a particular BGP neighbor does not have graceful restart capability, it does not establish an NSF-capable session with that neighbor. All other neighbors that have graceful restart capability continue to have NSF-capable sessions with this NSF-capable networking device.

OSPF Operation

When an OSPF NSF-capable router performs an active switch switchover, it must perform the following tasks in order to resynchronize its link state database with its OSPF neighbors:

- Relearn the available OSPF neighbors on the network without causing a reset of the neighbor relationship
- Reacquire the contents of the link state database for the network

As quickly as possible after an active switch switchover, the NSF-capable router sends an OSPF NSF signal to neighboring NSF-aware devices. Neighbor networking devices recognize this signal as an indicator that the neighbor relationship with this router should not be reset. As the NSF-capable router receives signals from other routers on the network, it can begin to rebuild its neighbor list.

After neighbor relationships are reestablished, the NSF-capable router begins to resynchronize its database with all of its NSF-aware neighbors. At this point, the routing information is exchanged between the OSPF neighbors. Once this exchange is complete, the NSF-capable device uses the routing information to remove stale routes, update the RIB, and update the FIB with the new forwarding information. The OSPF protocols are then fully converged.

Note OSPF support in NSF requires that all neighbor networking devices be NSF-aware. If an NSF-capable router discovers that it has non-NSF -aware neighbors on a particular network segment, it disables NSF capabilities for that segment. Other network segments composed entirely of NSF-capable or NSF-aware routers continue to provide NSF capabilities.

EIGRP Operation

When an EIGRP NSF-capable router initially re-boots after an NSF restart, it has no neighbor and its topology table is empty. The router is notified by the standby (now active) switch when it needs to bring up the interfaces, reacquire neighbors, and rebuild the topology and routing tables. The restarting router and its peers must accomplish these tasks without interrupting the data traffic directed toward the restarting router. EIGRP peer routers maintain the routes learned from the restarting router and continue forwarding traffic through the NSF restart process.

To prevent an adjacency reset by the neighbors, the restarting router uses a new Restart (RS) bit in the EIGRP packet header to indicate a restart. The RS bit is set in the hello packets and in the initial INIT update packets during the NSF restart period. The RS bit in the hello packets allows the neighbors to be quickly notified of the NSF restart. Without seeing the RS bit, the neighbor can only detect an adjacency reset by receiving an INIT update or by the expiration of the hello hold timer. Without the RS bit, a neighbor does not know if the adjacency reset should be handled using NSF or the normal startup method.

When the neighbor receives the restart indication, either by receiving the hello packet or the INIT packet, it recognizes the restarting peer in its peer list and maintains the adjacency with the restarting router. The neighbor then sends it topology table to the restarting router with the RS bit set in the first update packet indicating that it is NSF-aware and is helping out the restarting router. The neighbor does not set the RS bit in their hello packets, unless it is also a NSF restarting neighbor.

Note A router may be NSF-aware but may not be helping the NSF restarting neighbor because booting from a cold start.

If at least one of the peer routers is NSF-aware, the restarting router would then receive updates and rebuild its database. The restarting router must then find out if it had converged so that it can notify the routing information base (RIB). Each NSF-aware router is required to send an end of table (EOT) marker in the last update packet to indicate the end of the table content. The restarting router knows it has converged when it receives the EOT marker. The restarting router can then begin sending updates.

An NSF-aware peer would know when the restarting router had converged when it receives an EOT indication from the restarting router. The peer then scans its topology table to search for the routes with the restarted neighbor as the source. The peer compares the route timestamp with the restart event timestamp to determine if the route is still available. The peer then goes active to find alternate paths for the routes that are no longer available through the restarted router.

When the restarting router has received all EOT indications from its neighbors or when the NSF converge timer expires, EIGRP notifies the RIB of convergence. EIGRP waits for the RIB convergence signal and then floods its topology table to all awaiting NSF-aware peers.

How to configure Cisco NSF with SSO

Configuring SSO

- You must configure SSO in order to use NSF with any supported protocol.
- The SSO keyword is not supported if the Cisco IOS-XE software is running the LAN Base level license.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**redundancy** **Example:** `Switch(config)# redundancy`	Enters redundancy configuration mode.
Step 2	**mode sso** **Example:** `Switch(config-red)# mode sso`	Configures SSO. When this command is entered, the standby switch is reloaded and begins to work in SSO mode. **Note** The SSO keyword is not supported if the Cisco IOS-XE software is running the LAN Base level license.
Step 3	**end** **Example:** `Switch(config-red)# end`	Returns to EXEC mode.
Step 4	**show running-config** **Example:** `Switch# show running-config`	Verifies that SSO is enabled.
Step 5	**show redundancy states** **Example:** `Switch# show redundancy states`	Displays the operating redundancy mode.

Configuring SSO Example

This example shows how to configure the system for SSO and display the redundancy state.

```
Controller(config)# redundancy
Controller(config)# mode sso
Controller(config)# end
Controller# show redundancy states
my state = 13 -ACTIVE
peer state = 8 -STANDBY HOT
Mode = Duplex
Unit = Primary
Unit ID = 5
Redundancy Mode (Operational) = sso
Redundancy Mode (Configured) = sso
Split Mode = Disabled
Manual Swact = Enabled
Communications = Up
client count = 29
client_notification_TMR = 30000 milliseconds
keep_alive TMR = 9000 milliseconds
keep_alive count = 1
keep_alive threshold = 18
RF debug mask = 0x0
```

Configuring BGP for NSF

Before You Begin

Note You must configure BGP graceful restart on all peer devices participating in BGP NSF.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller(config)# configure terminal`	Enters global configuration mode.
Step 2	**router bgp as-number** **Example:** `Controller(config)# router bgp as-number`	Enables a BGP routing process, which places the switch in switch configuration mode.
Step 3	**bgp graceful-restart** **Example:** `Controller(config)# bgp graceful-restart`	Enables the BGP graceful restart capability, starting BGP NSF. If you enter this command after the BGP session has been established, you must restart the session for the capability to be exchanged with the BGP neighbor. Use this command on the restarting switch and all of its peers.

Configuring CEF NSF

The CEF NSF feature operates by default while the networking device is running in SSO mode. No configuration is necessary.

Verifying CEF NSF

```
Controllershow cef state
CEF Status:
RP instance
common CEF enabled
IPv4 CEF Status:
CEF enabled/running
dCEF enabled/running
CEF switching enabled/running
universal per-destination load sharing algorithm, id DEA83012
IPv6 CEF Status:
CEF disabled/not running
dCEF disabled/not running
universal per-destination load sharing algorithm, id DEA83012
RRP state:
I am standby RRP: no
RF Peer Presence: yes
RF PeerComm reached: yes
RF Progression blocked: never
Redundancy mode: rpr(1)
```

```
CEF NSF sync: disabled/not running
CEF ISSU Status:
FIBHWIDB broker
No slots are ISSU capable.
FIBIDB broker
No slots are ISSU capable.
FIBHWIDB Subblock broker
No slots are ISSU capable.
FIBIDB Subblock broker
No slots are ISSU capable.
Adjacency update
No slots are ISSU capable.
IPv4 table broker
No slots are ISSU capable.
CEF push
No slots are ISSU capable.
```

Configuring BGP for NSF

Before You Begin

Note You must configure BGP graceful restart on all peer devices participating in BGP NSF.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller(config)# configure terminal`	Enters global configuration mode.
Step 2	**router bgp as-number** **Example:** `Controller(config)# router bgp as-number`	Enables a BGP routing process, which places the switch in switch configuration mode.
Step 3	**bgp graceful-restart** **Example:** `Controller(config)# bgp graceful-restart`	Enables the BGP graceful restart capability, starting BGP NSF. If you enter this command after the BGP session has been established, you must restart the session for the capability to be exchanged with the BGP neighbor. Use this command on the restarting switch and all of its peers.

Verifying BGP NSF

To verify BGP NSF, you must check that BGP graceful restart is configured on the SSO-enabled networking device and on the neighbor devices. To verify, follow these steps:

Step 1 Verify that "bgp graceful-restart" appears in the BGP configuration of the SSO-enabled switch by entering the show running-config command:

Example:

```
Switch# show running-config
.
.
.
router bgp 120
.
.
.
bgp graceful-restart
neighbor 10.2.2.2 remote-as 300
.
.
.
```

Step 2 Repeat Step 1 on each of the BGP neighbors.

Step 3 On the SSO device and the neighbor device, verify that the graceful restart function is shown as both advertised and received, and confirm the address families that have the graceful restart capability. If no address families are listed, BGP NSF does not occur either:

Example:

```
Switch# show ip bgp neighbors
BGP neighbor is 31.31.31.7, remote AS 1, internal link
BGP version 4, remote router ID 7.7.7.7
BGP state = Established, up for 00:02:38
Last read 00:00:38, last write 00:00:35, hold time is 180, keepalive interval is 60
seconds
Neighbor capabilities:
Route refresh: advertised and received(new)
Address family IPv4 Unicast: advertised and received
Message statistics:
InQ depth is 0
OutQ depth is 0
Sent Rcvd
Opens: 1 1
Notifications: 0 0
Updates: 0 0
Keepalives: 4 4
Route Refresh: 0 0
Total: 5 5
Default minimum time between advertisement runs is 0 seconds
.....................................................
(Remaining output deleted)
```

Configuring OSPF NSF

Note All peer devices participating in OSPF NSF must be made OSPF NSF-aware, which happens automatically when you install an NSF software image on the device.

SUMMARY STEPS

1. **configure terminal**
2. **router ospf** *processID*
3. **nsf**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller(config)# configure terminal`	Enters global configuration mode.
Step 2	**router ospf** *processID* **Example:** `Controller(config)# router ospf processID`	Enables an OSPF routing process, which places the switch in router configuration mode.
Step 3	**nsf** **Example:** `Controller(config)# nsf`	Enables NSF operations for OSPF.

Verifying OSPF NSF

Step 1 Verify that 'nsf' appears in the OSPF configuration of the SSO-enabled device by entering the show running-config command:

Example:

```
Controller(config)#show running-config
route ospf 120
log-adjacency-changes
nsf
network 192.168.20.0 0.0.0.255 area 0
network 192.168.30.0 0.0.0.255 area 1
network 192.168.40.0 0.0.0.255 area 2
.
```

.
.

Step 2 Enter the **show ip ospf** command to verify that NSF is enabled on the device:

Example:

```
Controller show ip ospf
Routing Process "ospf 1" with ID 187.1.1.1
Start time: 00:02:07.532, Time elapsed: 00:39:05.052
Supports only single TOS(TOS0) routes
Supports opaque LSA
Supports Link-local Signaling (LLS)
transit capable is 0
External flood list length 0
IETF Non-Stop Forwarding enabled
restart-interval limit: 120 sec
IETF NSF helper support enabled
Cisco NSF helper support enabled
Reference bandwidth unit is 100 mbps
Area BACKBONE(0)
Number of interfaces in this area is 3 (1 loopback)
Area has no authentication
SPF algorithm last executed 00:08:53.760 ago
SPF algorithm executed 2 times
Area ranges are
Number of LSA 3. Checksum Sum 0x025BE0
Number of opaque link LSA 0. Checksum Sum 0x000000
Number of DCbitless LSA 0
Number of indication LSA 0
Number of DoNotAge LSA 0
Flood list length 0
```

Configuring EIGRP NSF

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller configure terminal`	Enters global configuration mode.
Step 2	**router eigrp** *as-number* **Example:** `Controller(config)# router eigrp as-number`	Enables an EIGRP routing process, which places the switch in router configuration mode.
Step 3	**nsf** **Example:** `Controller(config-router)# nsf`	Enables EIGRP NSF. Use this command on the “restarting” switch and all of its peers.

Verifying EIGRP NSF

Step 1 Verify that “nsf” appears in the EIGRP configuration of the SSO-enabled device by entering the show running-config command:

Example:

```
Controller show running-config
..
.
router eigrp 100
auto-summary
nsf
..
.
```

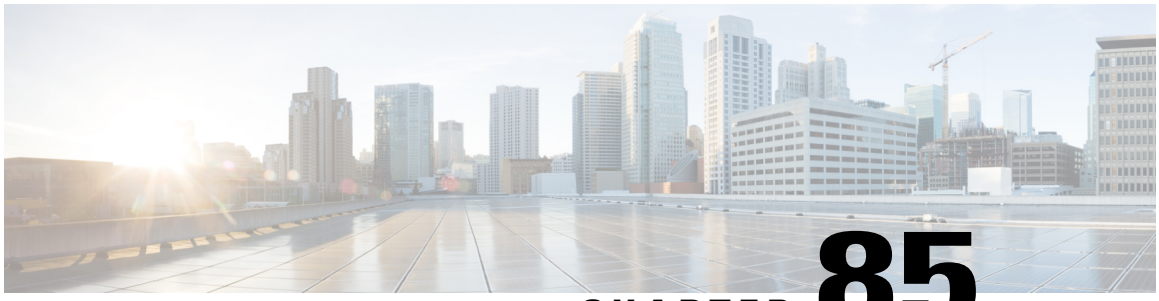
Step 2 Enter the show ip protocols command to verify that NSF is enabled on the device:

Example:

```
Controller show ip protocols
*** IP Routing is NSF aware ***
Routing Protocol is "ospf 1"
Outgoing update filter list for all interfaces is not set
Incoming update filter list for all interfaces is not set
Router ID 187.1.1.1
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
Maximum path: 1
Routing for Networks:
Routing on Interfaces Configured Explicitly (Area 0):
Loopback0
GigabitEthernet5/3
TenGigabitEthernet3/1
Routing Information Sources:
Gateway Distance Last Update
121.1.1.1 110 00:01:02
Distance: (default is 110)
Routing Protocol is "bgp 601"
Outgoing update filter list for all interfaces is not set
Incoming update filter list for all interfaces is not set
IGP synchronization is disabled
Automatic route summarization is disabled
Neighbor(s):
Address FiltIn FiltOut DistIn DistOut Weight RouteMap
150.1.1.1
Maximum path: 1
Routing Information Sources:
Gateway Distance Last Update
150.1.1.1 20 00:01:03
Distance: external 20 internal 200 local 200
```

PART XVII

Network Management



CHAPTER 85

Configuring Cisco IOS Configuration Engine

This chapter describes how to configure the Cisco IOS Configuration Engine.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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

The following are 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 controller.
- All controllers configured with the **cns config partial** global configuration command must access the event bus. Therefore, the DeviceID, as originated on the controller, must match the DeviceID of the corresponding controller definition in the Cisco Configuration Engine. You must know the hostname of the event bus to which you are connecting.

Related Topics

Cisco Networking Services (CNS) IDs and Device Hostnames, on page 1334
DeviceID, on page 1335

Restrictions for Configuring the Configuration Engine

The following are the restrictions for configuring the Cisco IOS Configuration Server:

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

Related Topics

Cisco Networking Services (CNS) IDs and Device Hostnames, on page 1334

Information About Configuring the Configuration Engine

Cisco Configuration Engine Software

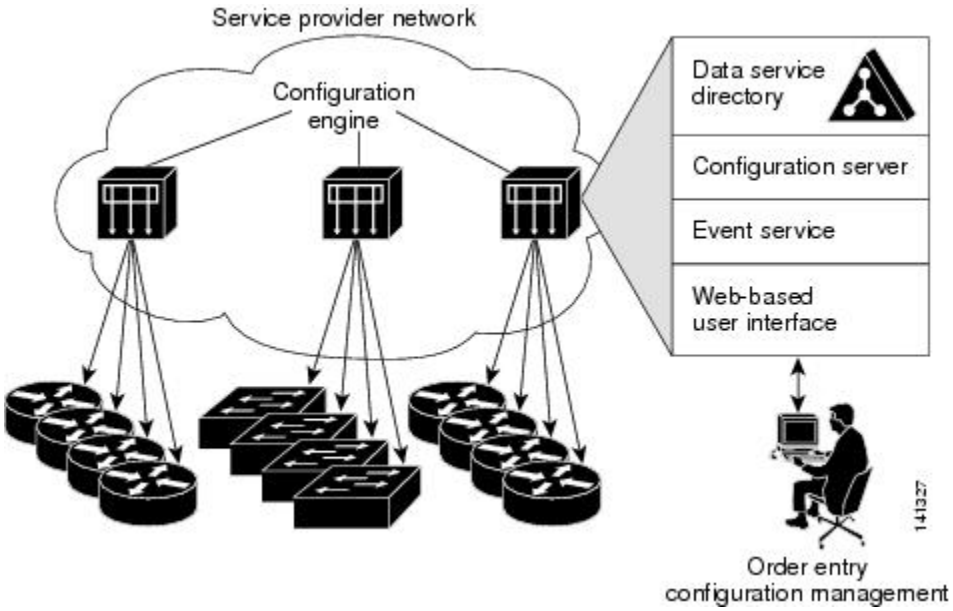
The Cisco Configuration Engine is network management utility software that acts as a configuration service for automating the deployment and management of network devices and services. Each Cisco Configuration Engine manages a group of Cisco devices (controllers and routers) and the services that they 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 59: 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 controller. The Configuration Service delivers device and service configurations to the controller for initial configuration and mass reconfiguration by logical groups. Controllers 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 controller and facilitates the communication between the controller 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

Enabling the CNS Event Agent, on page 1338

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 (CNS) IDs and Device Hostnames

The Cisco Configuration Engine assumes that a unique identifier is associated with each configured controller. 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

Prerequisites for Configuring the Configuration Engine, on page 1331

Restrictions for Configuring the Configuration Engine, on page 1332

ConfigID

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

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

DeviceID

Each configured controller participating on the event bus has a unique DeviceID, which is analogous to the controller source address so that the controller can be targeted as a specific destination on the bus.

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

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 controller. The event gateway represents the controller and its corresponding DeviceID to the event bus.

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

Related Topics

Prerequisites for Configuring the Configuration Engine, on page 1331

Hostname and DeviceID

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

When changing the controller hostname on the controller, the only way to refresh the DeviceID is to break the connection between the controller and the event gateway. For instructions on refreshing DeviceIDs, see Related Topics.

When the connection is reestablished, the controller 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 controller 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 1346

Hostname, DeviceID, and ConfigID

In standalone mode, when a hostname value is set for a controller, 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 controller.

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 controller to publish and subscribe to events on the event bus and works with the Cisco IOS CNS agent. These agents, embedded in the controller Cisco IOS software, allow the controller to be connected and automatically configured.

Initial Configuration

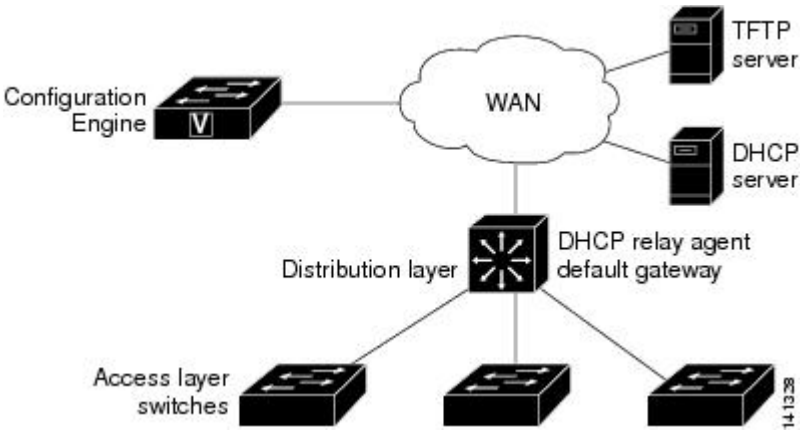
When the controller first comes up, it attempts to get an IP address by broadcasting a Dynamic Host Configuration Protocol (DHCP) request on the network. Assuming there is no DHCP server on the subnet, the distribution controller acts as a DHCP relay agent and forwards the request to the DHCP server. Upon receiving the request, the DHCP server assigns an IP address to the new controller 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 controller.

The controller 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 controller 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 controller.

The following figure shows a sample network configuration for retrieving the initial bootstrap configuration file by using DHCP-based autoconfiguration.

Figure 60: Initial Configuration



Related Topics

Automated CNS Configuration, on page 1337

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 controller. 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 controller to initiate a pull operation.

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

Synchronized Configuration

When the controller receives a configuration, it can defer application of the configuration upon receipt of a write-signal event. The write-signal event tells the controller not to save the updated configuration into its NVRAM. The controller uses the updated configuration as its running configuration. This ensures that the controller 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 controller, you must first complete the prerequisites listed in this topic. When you complete them, power on the controller. At the **setup** prompt, do nothing; the controller begins the initial configuration. When the full configuration file is loaded on your controller, you do not need to do anything else.

For more information on what happens during initial configuration, see Related Topics below.

Table 119: Prerequisites for Enabling Automatic Configuration

Device	Required Configuration
Access controller	Factory default (no configuration file)
Distribution controller	• IP helper address • Enable DHCP relay agent[22] • 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

Device	Required Configuration
TFTP server	• A bootstrap configuration file that includes the CNS configuration commands that enable the controller to communicate with the Configuration Engine • The controller configured to use either the controller 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 controller
CNS Configuration Engine	One or more templates for each type of device, with the ConfigID of the device mapped to the template.

[22] A DHCP Relay is needed only when the DHCP Server is on a different subnet from the client.

Related Topics

Initial Configuration, on page 1336

How to Configure the Configuration Engine

Enabling the CNS Event Agent

Note You must enable the CNS event agent on the controller before you enable the CNS configuration agent.

Beginning in privileged EXEC mode, follow these steps to enable the CNS event agent on the controller.

SUMMARY STEPS

1. **configure terminal**
2. **cns event** {*hostname* | *ip-address*} [*port-number*] [**backup**] [**failover-time** *seconds*] [**keepalive** *seconds retry-count*] [**reconnect** *time*] [**source** *ip-address*]
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**cns event** {*hostname* | *ip-address*} [*port-number*] [**backup**] [**failover-time** *seconds*] [**keepalive** *seconds retry-count*] [**reconnect** *time*] [**source** *ip-address*] **Example:** `Controller(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) Enter **backup** to show that this is the backup gateway. (If omitted, this is the primary gateway.) • (Optional) For **failover-time** *seconds*, enter how long the controller waits for the primary gateway route after the route to the backup gateway is established. • (Optional) For **keepalive** *seconds*, enter how often the controller sends keepalive messages. For *retry-count*, enter the number of unanswered keepalive messages that the controller sends before the connection is terminated. The default for each is 0. • (Optional) For **reconnect** *time*, enter the maximum time interval that the controller waits before trying to reconnect to the event gateway. • (Optional) For **source** *ip-address*, enter the source IP address of this device. **Note** Though visible in the command-line help string, the **encrypt** and the **clock-timeout** *time* keywords are not supported.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

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

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

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

Event Service, on page 1334

Enabling the Cisco IOS CNS Agent

Beginning in privileged EXEC mode, follow these steps to enable the Cisco IOS CNS agent on the controller.

Before You Begin

You must enable the CNS event agent on the controller before you enable this agent.

SUMMARY STEPS

1. **configure terminal**
2. **cns config initial** {*hostname* | *ip-address*} [*port-number*] [**source** *ip-address*]
3. **cns config partial** {*hostname* | *ip-address*} [*port-number*]
4. **end**
5. Start the Cisco IOS CNS agent on the controller.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**cns config initial** {*hostname* | *ip-address*} [*port-number*] [**source** *ip-address*] **Example:** `Controller(config)# cns config initial 10.180.1.27 10 cnshost`	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. • (Optional) Enter **source** *ip-address* to use for the source IP address. This command enables the Cisco IOS CNS agent and initiates an initial configuration on the controller.
Step 3	**cns config partial** {*hostname* | *ip-address*} [*port-number*]	Enables the Cisco IOS CNS agent, and enters the configuration server parameters.

	Command or Action	Purpose
	Example: `Controller(config)# cns config partial 10.180.1.27 10`	• 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. Enables the Cisco IOS CNS agent and initiates a partial configuration on the controller.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.
Step 5	Start the Cisco IOS CNS agent on the controller.	

What to Do Next

You can now use the Cisco Configuration Engine to remotely send incremental configurations to the controller.

Related Topics

Refreshing DeviceIDs, on page 1346

Enabling an Initial Configuration for Cisco IOS CNS Agent

Beginning in privileged EXEC mode, follow these steps to enable the CNS configuration agent and initiate an initial configuration on the controller.

SUMMARY STEPS

1. **configure terminal**
2. **cns template connect** *name*
3. **cli** *config-text*
4. Repeat Steps 2 to 3 to configure another CNS connect template.
5. **exit**
6. **cns connect** *name* [**retries** *number*] [**retry-interval** *seconds*] [**sleep** *seconds*] [**timeout** *seconds*]
7. **discover** {**controller** *controller-type* | **dlci** [**subinterface** *subinterface-number*] | **interface** [*interface-type*] | **line** *line-type*}
8. **template** *name* [*... name*]
9. Repeat Steps 7 to 8 to specify more interface parameters and CNS connect templates in the CNS connect profile.
10. **exit**
11. **hostname** *name*
12. **ip route** *network-number*
13. **cns id** *interface num* {**dns-reverse** | **ipaddress** | **mac-address**} [**event**] [**image**]
14. **cns id** {**hardware-serial** | **hostname** | **string** *string* | **udi**} [**event**] [**image**]
15. **cns config initial** {*hostname* | *ip-address*} [*port-number*] [**event**] [**no-persist**] [**page** *page*] [**source** *ip-address*] [**syntax-check**]
16. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**cns template connect** *name* **Example:** `Controller(config)# cns template connect template-dhcp`	Enters CNS template connect configuration mode, and specifies the name of the CNS connect template.
Step 3	**cli** *config-text* **Example:** `Controller(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 4	Repeat Steps 2 to 3 to configure another CNS connect template.	

	Command or Action	Purpose
Step 5	**exit** **Example:** `Controller(config)# exit`	Returns to global configuration mode.
Step 6	**cns connect** *name* [**retries** *number*] [**retry-interval** *seconds*] [**sleep** *seconds*] [**timeout** *seconds*] **Example:** `Controller(config)# cns connect dhcp`	Enters CNS connect configuration mode, specifies the name of the CNS connect profile, and defines the profile parameters. The controller uses the CNS connect profile to connect to the Configuration Engine. • 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 7	**discover** {**controller** *controller-type* | **dlci** [**subinterface** *subinterface-number*] | **interface** [*interface-type*] | **line** *line-type*} **Example:** `Controller(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 8	**template** *name* [*... name*] **Example:** `Controller(config-cns-conn)# template template-dhcp`	Specifies the list of CNS connect templates in the CNS connect profile to be applied to the controller configuration. You can specify more than one template.
Step 9	Repeat Steps 7 to 8 to specify more interface parameters and CNS connect templates in the CNS connect profile.	

	Command or Action	Purpose
Step 10	**exit** **Example:** `Controller(config-cns-conn)# exit`	Returns to global configuration mode.
Step 11	**hostname** *name* **Example:** `Controller(config)# hostname device1`	Enters the hostname for the controller.
Step 12	**ip route** *network-number* **Example:** `RemoteController(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 13	**cns id** *interface num* {**dns-reverse** | **ipaddress** | **mac-address**} [**event**] [**image**] **Example:** `RemoteController(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 controller. • (Optional) Enter **image** to set the ID to be the image-id value used to identify the controller. **Note** If both the **event** and **image** keywords are omitted, the image-id value is used to identify the controller.
Step 14	**cns id** {**hardware-serial** | **hostname** | **string** *string* | **udi**} [**event**] [**image**] **Example:** `RemoteController(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 controller serial number as the unique ID, enter **hostname** (the default) to select the controller 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.

	Command or Action	Purpose
Step 15	**cns config initial** {*hostname* | *ip-address*} [*port-number*] [**event**] [**no-persist**] [**page** *page*] [**source** *ip-address*] [**syntax-check**] **Example:** `RemoteController(config)# cns id ethernet 0 ipaddress`	Enables the Cisco IOS agent, and initiates an initial configuration. • 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 16	**end** **Example:** `RemoteController(config)# end`	Returns to privileged EXEC mode.

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

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

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

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

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.

Refreshing DeviceIDs

Beginning in privileged EXEC mode, follow these steps to refresh a DeviceID when changing the hostname on the controller.

SUMMARY STEPS

1. **show cns config connections**
2. Make sure that the CNS event agent is properly connected to the event gateway.
3. **show cns event gateway**
4. Record from the output of Step 3 the information for the currently connected gateway listed below. You will be using the IP address and port number in subsequent steps of these instructions.
5. **configure terminal**
6. **no cns event** *ip-address port-number*
7. **cns event** *ip-address port-number*
8. **end**
9. Make sure that you have reestablished the connection between the controller and the event gateway by examining the output from **show cns event gateway**.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show cns config connections** **Example:** `Controller# 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 2	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: • Connection is active. • Connection is using the currently configured controller hostname. The DeviceID will be refreshed to correspond to the new hostname configuration using these instructions.
Step 3	**show cns event gateway** **Example:** `Controller# show cns event gateway`	Displays the event gateway information for your controller.
Step 4	Record from the output of Step 3 the information for the currently connected gateway listed below. You will be using the IP address and port number in subsequent steps of these instructions.	
Step 5	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 6	**no cns event** *ip-address port-number* **Example:** `Controller(config)# no cns event 172.28.129.22 2012`	Specifies the IP address and port number that you recorded in Step 4 in this command. This command breaks the connection between the controller and the event gateway. It is necessary to first break, then reestablish, this connection to refresh the DeviceID.
Step 7	**cns event** *ip-address port-number* **Example:** `Controller(config)# cns event 172.28.129.22 2012`	Specifies the IP address and port number that you recorded in Step 4 in this command. This command reestablishes the connection between the controller and the event gateway.
Step 8	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 9	Make sure that you have reestablished the connection between the controller and the event gateway by examining the output from **show cns event gateway**.	

Related Topics

Enabling the Cisco IOS CNS Agent, on page 1340

Hostname and DeviceID, on page 1335

Enabling a Partial Configuration for Cisco IOS CNS Agent

Beginning in privileged EXEC mode, follow these steps to enable the Cisco IOS CNS agent and to initiate a partial configuration on the controller.

SUMMARY STEPS

1. **configure terminal**
2. **cns config partial** {*ip-address* | *hostname*} [*port-number*] [**source** *ip-address*]
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**cns config partial** {*ip-address* | *hostname*} [*port-number*] [**source** *ip-address*] **Example:** `Controller(config)# cns config partial 172.28.129.22 cnshost`	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. • (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.

	Command or Action	Purpose
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

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.

Monitoring CNS Configurations

Table 120: CNS show Commands

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

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

Standards and RFCs

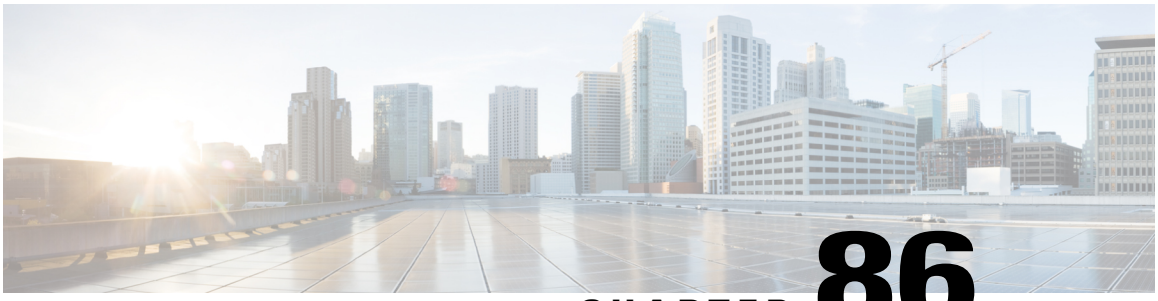
Standard/RFC	Title

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



CHAPTER 86

Configuring the Cisco Discovery Protocol

This chapter describes the configuration of the Cisco Discovery Protocol (CDP).

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 controller, CDP enables Network Assistant to display a graphical view of the network. The controller 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 controller by default.

CDP and Device Stacks

A controller stack appears as a single controller in the network. Therefore, CDP discovers the controller stack, not the individual stack members. The controller stack sends CDP messages to neighboring network devices when there are changes to the controller 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

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 2 through 4 are all optional and can be performed in any order.

Beginning in privileged EXEC mode, follow these steps to configure these characteristics.

SUMMARY STEPS

1. **configure terminal**
2. **cdp timer** *seconds*
3. **cdp holdtime** *seconds*
4. **cdp advertise-v2**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**cdp timer** *seconds* **Example:** `Controller(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 3	**cdp holdtime** *seconds* **Example:** `Controller(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 4	**cdp advertise-v2** **Example:** `Controller(config)# cdp advertise-v2`	(Optional) Configures CDP to send Version-2 advertisements. This is the default state.
Step 5	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Example

The following example shows how to configure CDP characteristics:

```
Controller# configure terminal
Controller(config)# cdp timer 50
Controller(config)# cdp holdtime 120
Controller(config)# cdp advertise-v2
Controller(config)# end
```

What to Do Next

Use the **no** form of the CDP commands to return to the default settings.

Related Topics

Monitoring and Maintaining CDP, on page 1358

Disabling CDP

CDP is enabled by default.

Note Controller clusters and other Cisco devices (such as Cisco IP Phones) regularly exchange CDP messages. Disabling CDP can interrupt cluster discovery and device connectivity.

Beginning in privileged EXEC mode, follow these steps to disable the CDP device discovery capability.

SUMMARY STEPS

1. **configure terminal**
2. **no cdp run**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**no cdp run** **Example:** `Controller(config)# no cdp run`	Disables CDP.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

What to Do Next

You must reenable CDP to use it.

Related Topics

Enabling CDP, on page 1355

Enabling CDP

CDP is enabled by default.

Note Controller clusters and other Cisco devices (such as Cisco IP Phones) regularly exchange CDP messages. Disabling CDP can interrupt cluster discovery and device connectivity.

Beginning in privileged EXEC mode, 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. **configure terminal**
2. **cdp run**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**cdp run** **Example:** `Controller(config)# cdp run`	Enables CDP if it has been disabled.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Enabling CDP

The following example shows how to enable CDP if it has been disabled:

```
Controller# configure terminal
Controller(config)# cdp run
Controller(config)# end
```

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

Disabling CDP, on page 1354

Disabling CDP on an Interface

CDP is enabled by default on all supported interfaces to send and to receive CDP information.

Note Controller clusters and other Cisco devices (such as Cisco IP Phones) regularly exchange CDP messages. Disabling CDP can interrupt cluster discovery and device connectivity.

Beginning in privileged EXEC mode, follow these steps to disable CDP on a port.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **no cdp enable**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet1/0/1`	Specifies the interface on which you are disabling CDP, and enters interface configuration mode.

	Command or Action	Purpose
Step 3	**no cdp enable** **Example:** `Controller(config-if)# no cdp enable`	Disables CDP on the interface specified in Step 2.
Step 4	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.

Related Topics

Enabling CDP on an Interface, on page 1357

Enabling CDP on an Interface

CDP is enabled by default on all supported interfaces to send and to receive CDP information.

Note Controller clusters and other Cisco devices (such as Cisco IP Phones) regularly exchange CDP messages. Disabling CDP can interrupt cluster discovery and device connectivity.

Beginning in privileged EXEC mode, 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. **configure terminal**
2. **interface** *interface-id*
3. **cdp enable**
4. **end**

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	**interface** *interface-id* **Example:** `Controller(config)# interface gigabitethernet1/0/1`	Specifies the interface on which you are enabling CDP, and enters interface configuration mode.
Step 3	**cdp enable** **Example:** `Controller(config-if)# cdp enable`	Enables CDP on a disabled interface.
Step 4	**end** **Example:** `Controller(config-if)# end`	Returns to privileged EXEC mode.

Example

The following example shows how to enable CDP on a disabled port:

```
Controller# configure terminal
Controller(config)# interface gigabitethernet1/0/1
Controller(config-if)# cdp enable
Controller(config-if)# end
```

Related Topics

Disabling CDP on an Interface, on page 1356

Monitoring and Maintaining CDP

Table 121: 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.

Command	Description
show cdp entry *entry-name* [**protocol** | **version**]	Displays information about a specific neighbor. You can enter an asterisk (*) to display all CDP neighbors, or you can enter the name of the neighbor about which you want information. You can also limit the display to information about the protocols enabled on the specified neighbor or information about the version of software running on the device.
show cdp interface [*interface-id*]	Displays information about interfaces where CDP is enabled. You can limit the display to the interface about which you want information.
show cdp neighbors [*interface-id*] [*detail*]	Displays information about neighbors, including device type, interface type and number, holdtime settings, capabilities, platform, and port ID. You can limit the display to neighbors of a specific interface or expand the display to provide more detailed information.
show cdp traffic	Displays CDP counters, including the number of packets sent and received and checksum errors.

Related Topics

Configuring CDP Characteristics, on page 1352

Additional References

Related Documents

Related Topic	Document Title
System Management Commands	*Cisco IOS Configuration Fundamentals Command Reference, Cisco IOS XE Release 3S (Catalyst 3850 Switches)*

Standards and RFCs

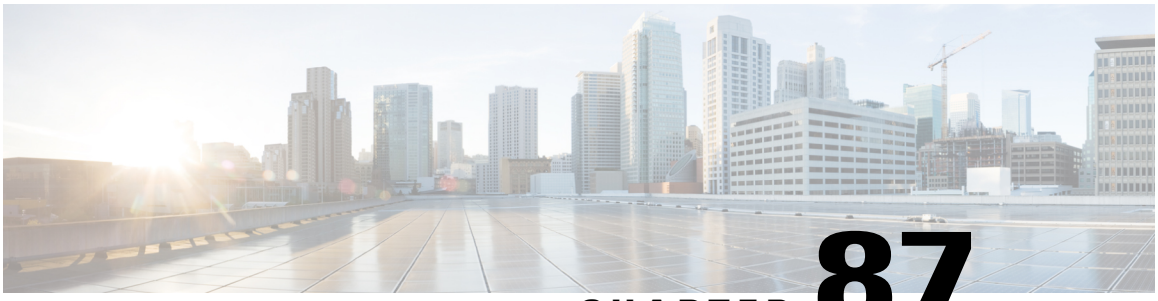
Standard/RFC	Title

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



CHAPTER 87

Configuring Simple Network Management Protocol

This chapter describes the Simple Network Management Protocol (SNMP) configuration.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 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 122: 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.

Model	Level	Authentication	Encryption	Result
SNMPv3	noAuthNoPriv	Username	No	Uses a username match for authentication.
SNMPv3	authNoPriv	Message Digest 5 (MD5) or Secure Hash Algorithm (SHA)	No	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms.
SNMPv3	authPriv	MD5 or SHA	Data Encryption Standard (DES) or Advanced Encryption Standard (AES)	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. Allows specifying the User-based Security Model (USM) with these encryption algorithms: • DES 56-bit encryption in addition to authentication based on the CBC-DES (DES-56) standard. • 3DES 168-bit encryption • AES 128-bit, 192-bit, or 256-bit encryption

You must configure the SNMP agent to use the SNMP version supported by the management station. Because an agent can communicate with multiple managers, you can configure the software to support communications using SNMPv1, SNMPv2C, or SNMPv3.

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 controller. To configure SNMP on the controller, 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.

The active controller handles the SNMP requests and traps for the whole controller stack. The active controller transparently manages any requests or traps that are related to all stack members. When a new active controller is elected, the new active controller continues to handle SNMP requests and traps as configured on the previous active controller, assuming that IP connectivity to the SNMP management stations is still in place after the new active controller has taken control.

SNMP Manager Functions

The SNMP manager uses information in the MIB to perform the operations described in the following table:

Table 123: 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.[23]
get-bulk-request[24]	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.

Operation	Description
trap	An unsolicited message sent by an SNMP agent to an SNMP manager when some event has occurred.

23 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.

24 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 controller, the community string definitions on the NMS must match at least one of the three community string definitions on the controller.

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 controller manages the exchange of messages among member controllers and the SNMP application. The Network Assistant software appends the member controller number (@esN, where N is the controller number) to the first configured RW and RO community strings on the command controller and propagates them to the member controllers.

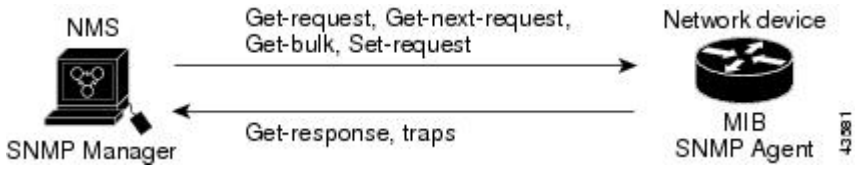
SNMP MIB Variables Access

An example of an NMS is the Cisco Prime Infrastructure network management software. Cisco Prime Infrastructure 2.0 software uses the controller 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 61: SNMP Network



SNMP Notifications

SNMP allows the controller 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 controller 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 controller 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 controller reboots or the controller software is upgraded, the controller uses this same value for the interface. For example, if the controller assigns a port 2 an ifIndex value of 10003, this value is the same after the controller reboots.

The controller uses one of the values in the following table to assign an ifIndex value to an interface:

Table 124: ifIndex Values

Interface Type	ifIndex Range
SVI[25]	1–4999

Interface Type	ifIndex Range
EtherChannel	5000–5012
Loopback	5013–5077
Tunnel	5078–5142
Physical (such as Gigabit Ethernet or SFP[26]-module interfaces)	10000–14500
Null	14501

25 SVI = switch virtual interface

26 SFP = small form-factor pluggable

Default SNMP Configuration

Feature	Default Setting
SNMP agent	Disabled[27].
SNMP trap receiver	None configured.
SNMP traps	None enabled except the trap for TCP connections (tty).
SNMP version	If no version keyword is present, the default is Version 1.
SNMPv3 authentication	If no keyword is entered, the default is the **noauth** (noAuthNoPriv) security level.
SNMP notification type	If no type is specified, all notifications are sent.

27 This is the default when the controller starts and the startup configuration does not have any **snmp-server** global configuration commands.

SNMP Configuration Guidelines

If the controller starts and the controller 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 controller does not send informs for the **auth** (authNoPriv) and the **priv** (authPriv) authentication levels.
- Changing the value of the SNMP engine ID has important side effects. 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.

Beginning in privileged EXEC mode, 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. **configure terminal**
2. **no snmp-server**
3. **end**

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	**no snmp-server** **Example:** `Controller(config)# no snmp-server`	Disables the SNMP agent operation.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

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 controller. 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

Beginning in privileged EXEC mode, follow these steps to configure a community string on the controller.

SUMMARY STEPS

1. **configure terminal**
2. **snmp-server community** *string* [**view** *view-name*] [**ro** | **rw**] [*access-list-number*]
3. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
4. **end**

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	**snmp-server community** *string* [**view** *view-name*] [**ro** | **rw**] [*access-list-number*] **Example:** `Controller(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-name*, 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 3	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] **Example:** `Controller(config)# access-list 4 deny any`	(Optional) If you specified an IP standard access list number in Step 2, then create the list, repeating the command as many times as necessary. • For *access-list-number*, enter the access list number specified in Step 2. • The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • For *source*, enter the 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.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

This example shows how to assign the string comaccess to SNMP, to allow read-only access, and to specify that IP access list 4 can use the community string to gain access to the controller SNMP agent:

```
Controller(config)# snmp-server community comaccess ro 4
```

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 controller. 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 controller. You can configure an SNMP server group that maps SNMP users to SNMP views, and you can add new users to the SNMP group.

Beginning in privileged EXEC mode, follow these steps to configure SNMP groups and users on the controller.

SUMMARY STEPS

1. **configure terminal**
2. **snmp-server engineID** {**local** *engineid-string* | **remote** *ip-address* [**udp-port** *port-number*] *engineid-string*}
3. **snmp-server group** *group-name* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*]
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*] } [*priv* {**des** | **3des** | **aes** {**128** | **192** | **256**}} *priv-password*]
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**snmp-server engineID** {**local** *engineid-string* | **remote** *ip-address* [**udp-port** *port-number*] *engineid-string*} **Example:** `Controller(config)# snmp-server engineID local 1234`	Configures a name for either the local or remote copy of SNMP. • The *engineid-string* is a 24-character ID string with the name of the copy of SNMP. You need not specify the entire 24-character engine ID if it has trailing zeros. Specify only the portion of the engine ID up to the point where only zeros remain in the value. The Step Example configures an engine ID of 123400000000000000000000. • If you select **remote**, specify the *ip-address* of the device that contains the remote copy of SNMP and the optional User Datagram Protocol (UDP) port on the remote device. The default is 162.

	Command or Action	Purpose
Step 3	**snmp-server group** *group-name* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*] **Example:** `Controller(config)# snmp-server group public v2c access lmnop`	Configures a new SNMP group on the remote device. For *group-name*, specify the name of the group. Specify one of the following security models: • **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. **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 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*] } [*priv* {**des** | **3des** | **aes** {**128** | **192** | **256**}} *priv-password*] **Example:** `Controller(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).

	Command or Action	Purpose
		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. • **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 5	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Configuring SNMP Notifications

A trap manager is a management station that receives and processes traps. Traps are system alerts that the controller generates when certain events occur. By default, no trap manager is defined, and no traps are sent. Controllers 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.

The table below describes the supported controller traps (notification types). You can enable any or all of these traps and configure a trap manager to receive them.

You can use the **snmp-server host** global configuration command to a specific host to receive the notification types listed in the following table:

Table 125: Device Notification Types

Notification Type Keyword	Description
bgp	Generates Border Gateway Protocol (BGP) state change traps. This option is only available when the IP services feature set is enabled.
bridge	Generates STP bridge MIB traps.

Notification Type Keyword	Description
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.
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.
flash	Generates SNMP FLASH notifications. In a controller stack, you can optionally enable notification for flash insertion or removal, which would cause a trap to be issued whenever a controller 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 controller stack, this trap refers to the insertion or removal of a controller 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).

Notification Type Keyword	Description
snmp	Generates a trap for SNMP-type notifications for authentication, cold start, warm start, link up or link down.
storm-control	Generates a trap for SNMP storm-control. You can also set a maximum trap rate per minute. The range is from 0 to 1000; the default is 0 (no limit is imposed; a trap is sent at every occurrence).
stpx	Generates SNMP STP Extended MIB traps.
syslog	Generates SNMP syslog traps.
tty	Generates a trap for TCP connections. This trap is enabled by default.
vlan-membership	Generates a trap for SNMP VLAN membership changes.
vlancreate	Generates SNMP VLAN created traps.
vlandelete	Generates SNMP VLAN deleted traps.
vtp	Generates a trap for VLAN Trunking Protocol (VTP) changes.

Beginning in privileged EXEC mode, follow these steps to configure the controller to send traps or informs to a host.

SUMMARY STEPS

1. **configure terminal**
2. **snmp-server engineID remote** *ip-address engineid-string*
3. **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*] }
4. **snmp-server group** *group-name* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*]
5. **snmp-server host** *host-addr* [**informs** | **traps**] [**version** {**1** | **2c** | **3** {**auth** | **noauth** | **priv**}}] *community-string* [*notification-type*]
6. **snmp-server enable traps** *notification-types*
7. **snmp-server trap-source** *interface-id*
8. **snmp-server queue-length** *length*
9. **snmp-server trap-timeout** *seconds*
10. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**snmp-server engineID remote** *ip-address engineid-string* **Example:** `Controller(config)# snmp-server engineID remote 192.180.1.27 00000063000100a1c0b4011b`	Specify the engine ID for the remote host.
Step 3	**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:** `Controller(config)# snmp-server user Pat public v2c`	Configures an SNMP user to be associated with the remote host created in Step 2. **Note** You cannot configure a remote user for an address without first configuring the engine ID for the remote host. Otherwise, you receive an error message, and the command is not executed.
Step 4	**snmp-server group** *group-name* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*] **Example:** `Controller(config)# snmp-server group public v2c access lmnop`	Configures an SNMP group.
Step 5	**snmp-server host** *host-addr* [**informs** | **traps**] [**version** {**1** | **2c** | **3** {**auth** | **noauth** | **priv**}}] *community-string* [*notification-type*] **Example:** `Controller(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. 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**. 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. The @ symbol is used for delimiting the context information. Avoid using the @ symbol as part of the SNMP community string when configuring this command.

	Command or Action	Purpose
		(Optional) For *notification-type*, use the keywords listed in the table above. If no type is specified, all notifications are sent.
Step 6	**snmp-server enable traps** *notification-types* **Example:** `Controller(config)# snmp-server enable traps snmp`	Enable the controller to send traps or informs and specify 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: **1 snmp-server enable traps port-security** **2 snmp-server enable traps port-security trap-rate** *rate*
Step 7	**snmp-server trap-source** *interface-id* **Example:** `Controller(config)# snmp-server trap-source GigabitEthernet1/0/1`	(Optional) Specify the source interface, which provides the IP address for the trap message. This command also sets the source IP address for informs.
Step 8	**snmp-server queue-length** *length* **Example:** `Controller(config)# snmp-server queue-length 20`	(Optional) Establish the message queue length for each trap host. The range is 1 to 1000; the default is 10.
Step 9	**snmp-server trap-timeout** *seconds* **Example:** `Controller(config)# snmp-server trap-timeout 60`	(Optional) Define how often to resend trap messages. The range is 1 to 1000; the default is 30 seconds.
Step 10	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

What to Do Next

The **snmp-server host** command specifies which hosts receive the notifications. The **snmp-server enable trap** command globally enables the mechanism 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

Beginning in privileged EXEC mode, 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. **configure terminal**
2. **snmp-server contact** *text*
3. **snmp-server location** *text*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**snmp-server contact** *text* **Example:** `Controller(config)# snmp-server contact Dial System Operator at beeper 21555`	Sets the system contact string.
Step 3	**snmp-server location** *text* **Example:** `Controller(config)# snmp-server location Building 3/Room 222`	Sets the system location string.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Limiting TFTP Servers Used Through SNMP

Beginning in privileged EXEC mode, 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. **configure terminal**
2. **snmp-server tftp-server-list** *access-list-number*
3. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**snmp-server tftp-server-list** *access-list-number* **Example:** `Controller(config)# snmp-server tftp-server-list 44`	Limit 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 3	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] **Example:** `Controller(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 2. The **deny** keyword denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. For *source*, enter the IP address of the TFTP servers that can access the controller. (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.
Step 4	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Configuring Trap Flags for SNMP

SUMMARY STEPS

1. **configure terminal**
2. **trapflags ap** { **interfaceup | register**}
3. **trapflags client** {**dot11 | excluded**}
4. **trapflags dot11-security** {**ids-sig-attack | wep-decrypt-error**}
5. **trapflags mesh**
6. **trapflags rogueap**
7. **trapflags rrm-params** {**channels | tx-power**}
8. **trapflags rrm-profile** {**coverage | interference | load | noise**}
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**trapflags ap** { **interfaceup	register**} **Example:** `Controller(config)# trapflags ap interfaceup`	Enables sending AP related traps. Use the no form of the command to disable the trap flags. • **interfaceup–** Enables trap when a Cisco AP interface(A or B) comes up. • **register–** Enables trap when a Cisco AP registers with a Cisco controller.
Step 3	**trapflags client** {**dot11	excluded**} **Example:** `Controller(config)# trapflags client excluded`	Enables sending client-related dot11 traps. Add "no" in the command to disable the trapflags. • **dot11–** Enables dot11 traps for clients. • **excluded–** Enables excluded traps for clients.
Step 4	**trapflags dot11-security** {**ids-sig-attack	wep-decrypt-error**} **Example:** `Controller(config)# trapflags dot11-security wep-decrypt-error`	Enables sending 802.11 security-related traps. Add "no" in the command to disable the trapflags. • **ids-sig-attack–** Enables IDS signature attack traps. • **wep-decrypt-error–** Enables traps for WEP decrypt error for clients.

	Command or Action	Purpose
Step 5	**trapflags mesh** **Example:** `Controller(config)# trapflags mesh`	Enables trap for the mesh. Use the no form of the command to disable the trapflags.
Step 6	**trapflags rogueap** **Example:** `Controller(config)# trapflags rogueap`	Enables trap for rogue ap detection. Use the no form of the command to disable the trapflags.
Step 7	**trapflags rrm-params** {**channels** | **tx-power**} **Example:** `Controller(config)# trapflags rrm-params tx-power`	Enables sending RRM-parameter update related traps. Add "no" in the command to disable the trapflags. • **channels–** Enables trap when RF Manager automatically changes a channel number for the Cisco AP interface. • **tx-power–** Enables the trap when RF Manager automatically changes Tx-Power level for the Cisco AP interface.
Step 8	**trapflags rrm-profile** {**coverage** | **interference** | **load** | **noise**} **Example:** `Controller(config)# trapflags rrm-profile interference`	Enables sending RRM-profile-related traps. Add "no" in the command to disable the trapflags. • **coverage–** Enables the trap when the coverage profile maintained by RF Manager fails. • **interference–** Enables the trap when the interference profile maintained by RF Manager fails. • **load–** Enables trap when the load profile maintained by RF Manager fails. • **noise–** Enables trap when the noise profile maintained by RF Manager fails.
Step 9	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Enabling SNMP Wireless Trap Notification

SUMMARY STEPS

1. **configure terminal**
2. **snmp-server enable traps wireless** [**AP** | **RRM** | **bsn80211SecurityTrap** | **bsnAPParamUpdate** | **bsnAPProfile** | **bsnAccessPoint** | **bsnMobileStation** | **bsnRogue** | **client** | **mfp** | **rogue**]
3. **end**

DETAILED STEPS

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

Step 2	**snmp-server enable traps wireless** [**AP** | **RRM** | **bsn80211SecurityTrap** | **bsnAPParamUpdate** | **bsnAPProfile** | **bsnAccessPoint** | **bsnMobileStation** | **bsnRogue** | **client** | **mfp** | **rogue**]
	
	Example:
	`Controller(config)# snmp-server enable traps wireless AP`
	`Controller(config)# snmp-server enable traps wireless RRM`
	`Controller(config)# snmp-server enable traps wireless bsn80211SecurityTrap`
	`Controller(config)# snmp-server enable traps wireless bsnAPParamUpdate`
	`Controller(config)# snmp-server enable traps wireless bsnAPProfile`
	`Controller(config)# snmp-server enable traps wireless bsnAccessPoint`
	`Controller(config)# snmp-server enable traps wireless bsnMobileStation`
	`Controller(config)# snmp-server enable traps wireless bsnRogue`
	`Controller(config)# snmp-server enable traps wireless client`
	`Controller(config)# snmp-server enable traps wireless mfp`
	`Controller(config)# snmp-server enable traps wireless rogue`
		Enable SNMP wireless trap notification.
		• **AP–** Enables access point traps.
		• **RRM–** Enables RRM traps.
		• **bsn80211SecurityTrap–** Enables the security-related trap.
		• **bsnAPParamUpdate–** Enables the trap for AP parameters that get updated.
		• **bsnAPProfile–** Enables BSN AP profile traps.
		• **bsnAccessPoint–** Enables BSN access point traps.
		• **bsnMobileStation–** Controls wireless client traps.
		• **bsnRogue–** Enables BSN rogue-related traps.
		• **client–** Enables client traps.
		• **mfp–** Enables MFP traps.
		• **rogue–** Enables rogue-related traps.

	Command or Action	Purpose
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

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 126: Commands for Displaying SNMP Information

Feature	Default Setting
show snmp	Displays SNMP statistics.
show snmp engineID	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 controller to send any traps.

```
Controller(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 controller 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.

```
Controller(config)# snmp-server community public
Controller(config)# snmp-server enable traps vtp
Controller(config)# snmp-server host 192.180.1.27 version 2c public
Controller(config)# snmp-server host 192.180.1.111 version 1 public
Controller(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*.

```
Controller(config)# snmp-server community comaccess ro 4
Controller(config)# snmp-server enable traps snmp authentication
Controller(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 controller 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*.

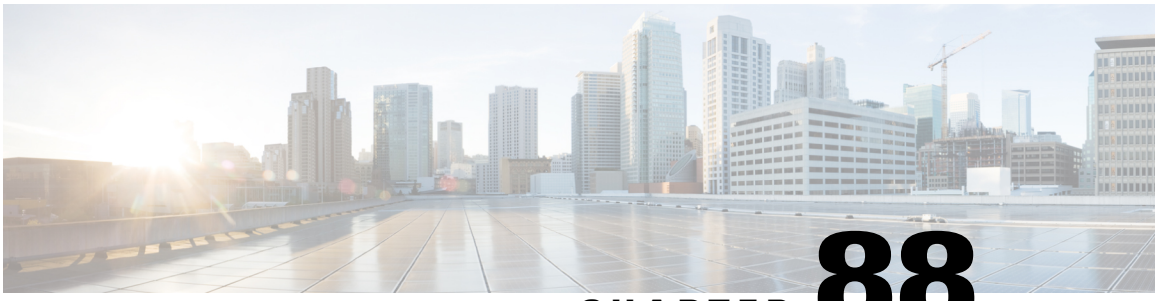
```
Controller(config)# snmp-server enable traps entity
Controller(config)# snmp-server host cisco.com restricted entity
```

This example shows how to enable the controller to send all traps to the host *myhost.cisco.com* using the community string *public*:

```
Controller(config)# snmp-server enable traps
Controller(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:

```
Controller(config)# snmp-server engineID remote 192.180.1.27 00000063000100a1c0b4011b
Controller(config)# snmp-server group authgroup v3 auth
Controller(config)# snmp-server user authuser authgroup remote 192.180.1.27 v3 auth md5
mypassword
Controller(config)# snmp-server user authuser authgroup v3 auth md5 mypassword
Controller(config)# snmp-server host 192.180.1.27 informs version 3 auth authuser config
Controller(config)# snmp-server enable traps
Controller(config)# snmp-server inform retries 0
```



CHAPTER 88

Configuring Service Level Agreements

This chapter describes how to configure Service Level Agreements (SLAs).

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 on SLAs

This section lists the restrictions on SLAs.

The following are restrictions on IP SLAs network performance measurement:

- The controller does not support Voice over IP (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.

Related Topics

Implementing IP SLAs Network Performance Measurement, on page 1393
Network Performance Measurement with Cisco IOS IP SLAs, on page 1387
IP SLAs Responder and IP SLAs Control Protocol, on page 1388

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 a unique subset of 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.

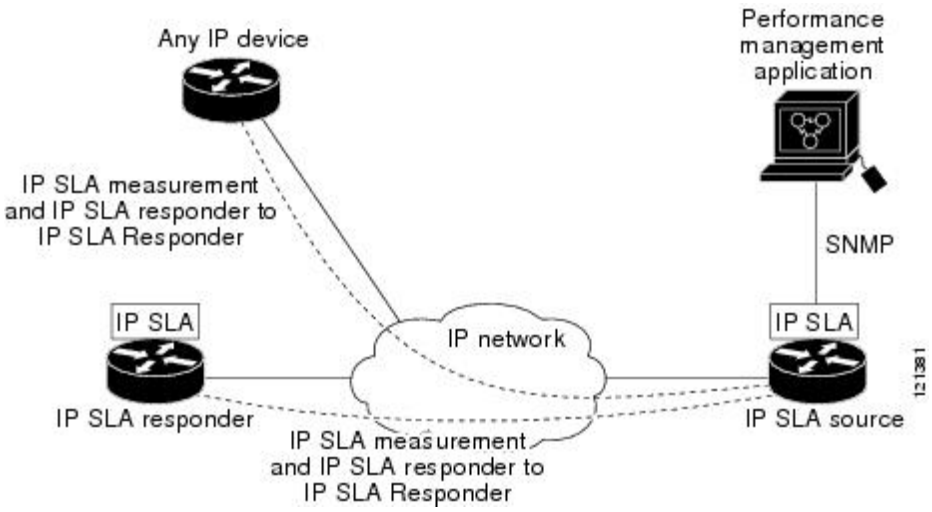
- IP service network health assessment to verify that the existing QoS is sufficient for new IP services.
- Edge-to-edge network availability monitoring for proactive verification and connectivity testing of network resources (for example, shows the network availability of an NFS server used to store 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 controller 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.

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.

Figure 62: Cisco IOS IP SLAs Operation



Related Topics

Implementing IP SLAs Network Performance Measurement, on page 1393

Restrictions on SLAs, on page 1385

IP SLAs Responder and IP SLAs Control Protocol

The IP SLAs responder is a component embedded in the destination Cisco device that allows the system to anticipate and respond to IP SLAs request packets. The responder provides accurate measurements without the need for dedicated probes. The responder uses the Cisco IOS IP SLAs Control Protocol to provide a mechanism through which it can be notified on which port it should listen and respond.

Note The IP SLAs responder can be a Cisco IOS Layer 2, responder-configurable controller. The responder does not need to support full IP SLAs functionality.

The following figure shows where the Cisco IOS IP SLAs responder fits in the IP network. The responder listens on a specific port for control protocol messages sent by an IP SLAs operation. Upon receipt of the control message, it enables the specified UDP or TCP port for the specified duration. During this time, the responder accepts the requests and responds to them. It disables the port after it responds to the IP SLAs packet, or when the specified time expires. MD5 authentication for control messages is available for added security.

You do not need to enable the responder on the destination device for all IP SLAs operations. For example, a responder is not required for services that are already provided by the destination router (such as Telnet or HTTP).

Related Topics

Restrictions on SLAs, on page 1385

Response Time Computation for IP SLAs

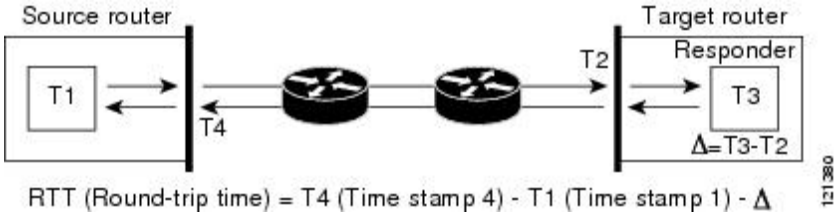
Switches, controllers, and routers can take tens of milliseconds to process incoming packets due to other high priority processes. This delay affects the response times because the test-packet reply might be in a queue while waiting to be processed. In this situation, the response times would not accurately represent true network delays. IP SLAs minimizes these processing delays on the source device as well as on the target device (if the responder is being used) to determine true round-trip times. IP SLAs test packets use time stamping to minimize the processing delays.

When the IP SLAs responder is enabled, it allows the target device to take time stamps when the packet arrives on the interface at interrupt level and again just as it is leaving, eliminating the processing time. This time stamping is made with a granularity of sub-milliseconds (ms).

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.

Figure 63: Cisco IOS IP SLAs Responder Time Stamping



An additional benefit of the two time stamps at the target device is the ability to track one-way delay, jitter, and directional packet loss. Because much network behavior is asynchronous, it is critical to have these statistics. However, to capture one-way delay measurements, you must configure both the source router and target router with Network Time Protocol (NTP) so that the source and target are synchronized to the same clock source. One-way jitter measurements do not require clock synchronization.

IP SLAs Operation Scheduling

When you configure an IP SLAs operation, you must schedule the operation to begin capturing statistics and collecting error information. You can schedule an operation to start immediately or to start at a certain month, day, and hour. You can use the *pending* option to set the operation to start at a later time. The pending option is an internal state of the operation that is visible through SNMP. The pending state is also used when an operation is a reaction (threshold) operation waiting to be triggered. You can schedule a single IP SLAs operation or a group of operations at one time.

You can schedule several IP SLAs operations by using a single command through the Cisco IOS CLI or the CISCO RTTMON-MIB. Scheduling the operations to run at evenly distributed times allows you to control the amount of IP SLAs monitoring traffic. This distribution of IP SLA operations helps minimize the CPU utilization and thus improves network scalability.

For more details about the IP SLA multi-operations scheduling functionality, see the "IP SLAs—Multiple Operation Scheduling" chapter of the *Cisco IOS IP SLAs Configuration Guide*.

IP SLAs Operation Threshold Monitoring

To support successful service level agreement monitoring, you must have mechanisms that notify you immediately of any possible violation. IP SLAs can send SNMP traps that are triggered by events such as the following:

- Connection loss
- Timeout
- Round-trip time threshold
- Average jitter threshold
- One-way packet loss
- One-way jitter
- One-way mean opinion score (MOS)

- One-way latency

An IP SLA threshold violation can also trigger another IP SLA operation for further analysis. For example, the frequency could be increased or an Internet Control Message Protocol (ICMP) path echo or ICMP path jitter operation could be initiated for troubleshooting.

ICMP Echo

The ICMP echo operation measures the end-to-end response time between a Cisco device and any other device that uses IP. The response time is computed by measuring the time it takes to send an ICMP echo request message to a destination and receive an ICMP echo reply. Many customers use IP SLAs ICMP-based operations, in-house ping testing, or ping-based dedicated probes to measure this response time. The IP SLAs ICMP echo operation conforms to the same specifications as ICMP ping testing, and both methods result in the same response times.

Related Topics

Analyzing IP Service Levels by Using the ICMP Echo Operation, on page 1399

UDP Jitter

Jitter is a simple term that describes interpacket delay variance. When multiple packets are sent consecutively at an interval of 10 ms from source to destination, the destination should receive them 10 ms apart (if the network is behaving correctly). However if there are delays in the network (like queuing, arriving through alternate routes, and so on), the time interval between packet arrivals might be more or less than 10 ms. A positive jitter value indicates that the packets arrived more than 10 ms apart. A negative jitter value indicates that the packets arrived less than 10 ms apart. If the packets arrive 12 ms apart, the positive jitter is 2 ms; if the packets arrive 8 ms apart, the negative jitter is 2 ms. For delay-sensitive networks, positive jitter values are undesirable, and a jitter value of 0 is ideal.

In addition to monitoring jitter, the IP SLAs UDP jitter operation can be used as a multipurpose data gathering operation. The packets generated by IP SLAs carry sequence info and time stamps from the source and operational target that include packet sending and receiving data. Based on this data, UDP jitter operations measure the following:

- Per-direction jitter (source to destination and destination to source)
- Per-direction packet-loss
- Per-direction delay (one-way delay)
- Round-trip delay (average round-trip time)

Because the paths for the sending and receiving of data can be different (asymmetric), you can use the per-direction data to more readily identify where congestion or other problems are occurring in the network.

The UDP jitter operation generates synthetic (simulated) UDP traffic and sends a number of UDP packets, each of a specified size, sent a specified number of milliseconds apart, from a source router to a target router, at a given frequency. By default, ten packet-frames, each with a payload size of 10 bytes are generated every 10 ms, and the operation is repeated every 60 seconds. You can configure each of these parameters to best simulate the IP service you want to provide.

To provide accurate one-way delay (latency) measurements, time synchronization, such as that provided by NTP, is required between the source and the target device. Time synchronization is not required for the one-way jitter and packet loss measurements. If the time is not synchronized between the source and target devices,

one-way jitter and packet loss data is returned, but values of *0* are returned for the one-way delay measurements provided by the UDP jitter operation.

Related Topics

Analyzing IP Service Levels by Using the UDP Jitter Operation, on page 1396

Configuration Guidelines

For information on the IP SLAs 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*.

Note that not all of the IP SLAs commands or operations described in the above referenced guide are supported on the controller. The controller supports IP service level analysis by using UDP jitter, UDP echo, HTTP, TCP connect, ICMP echo, ICMP path echo, ICMP path jitter, FTP, DNS, and DHCP, as well as multiple operation scheduling and proactive threshold monitoring. It does not support VoIP service levels using the gatekeeper registration delay operations measurements.

Before configuring any IP SLAs application, you can use the **show ip sla application** privileged EXEC command to verify that the operation type is supported on your software image. This is an example of the output from the command:

```
Controller# show ip sla application

        IP Service Level Agreements
Version: Round Trip Time MIB 2.2.0, Infrastructure Engine-III

Supported Operation Types:
        icmpEcho, path-echo, path-jitter, udpEcho, tcpConnect, http
        dns, udpJitter, dhcp, ftp, udpApp, wspApp

Supported Features:
        IPSLAs Event Publisher

IP SLAs low memory water mark: 33299323
Estimated system max number of entries: 24389

Estimated number of configurable operations: 24389
Number of Entries configured  : 0
Number of active Entries      : 0
Number of pending Entries     : 0
Number of inactive Entries    : 0
Time of last change in whole IP SLAs: *13:04:37.668 UTC Wed Dec 19 2012
```

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*.

Configuring the IP SLAs Responder

The IP SLAs responder is available only on Cisco IOS software-based devices, including some Layer 2 controllers that do not support full IP SLAs functionality.

Beginning in privileged EXEC mode, follow these steps to configure the IP SLAs responder on the target device (the operational target):

SUMMARY STEPS

1. **configure terminal**
2. **ip sla responder** {**tcp-connect** | **udp-echo**} **ipaddress** *ip-address* **port** *port-number*
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ip sla responder** {**tcp-connect** | **udp-echo**} **ipaddress** *ip-address* **port** *port-number* **Example:** `Controller(config)# ip sla responder udp-echo 172.29.139.134 5000`	Configures the controller as an IP SLAs 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 SLAs operation.
Step 3	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

UDP Jitter Example

This example shows how to configure the device as a responder for the UDP jitter IP SLAs operation in the next procedure:

```
Controller(config)# ip sla responder udp-echo 172.29.139.134 5000
```

Implementing IP SLAs Network Performance Measurement

Beginning in privileged EXEC mode, follow these steps to implement IP SLAs network performance measurement on your controller:

Before You Begin

Use the **show ip sla application** privileged EXEC command to verify that the desired operation type is supported on your software image.

SUMMARY STEPS

1. **configure terminal**
2. **ip sla** *operation-number*
3. **udp-jitter** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*}] [**source-port** *port-number*] [**control** {**enable** | **disable**}] [**num-packets** *number-of-packets*] [**interval** *interpacket-interval*]
4. **frequency** *seconds*
5. **threshold** *milliseconds*
6. **exit**
7. **ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm* [*:ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**]
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ip sla** *operation-number* **Example:** `Controller(config)# ip sla 10`	Creates an IP SLAs operation, and enters IP SLAs configuration mode.

	Command or Action	Purpose
Step 3	**udp-jitter** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*}] [**source-port** *port-number*] [**control** {**enable** | **disable**}] [**num-packets** *number-of-packets*] [**interval** *interpacket-interval*] **Example:** `Controller(config-ip-sla)# udp-jitter 172.29.139.134 5000`	Configures the IP SLAs operation as the operation type of your choice (a UDP jitter operation is used in the example), and enters its configuration mode (UDP jitter configuration mode is used in the example). • *destination-ip-address* | *destination-hostname*—Specifies the destination IP address or hostname. • *destination-port*—Specifies the destination port number in the range from 1 to 65535. • (Optional) **source-ip** {*ip-address* | *hostname*}—Specifies the source IP address or hostname. When a source IP address or hostname is not specified, IP SLAs chooses the IP address nearest to the destination • (Optional) **source-port** *port-number*—Specifies the source port number in the range from 1 to 65535. When a port number is not specified, IP SLAs chooses an available port. • (Optional) **control**—Enables or disables sending of IP SLAs control messages to the IP SLAs responder. By default, IP SLAs control messages are sent to the destination device to establish a connection with the IP SLAs responder • (Optional) **num-packets** *number-of-packets*—Enters the number of packets to be generated. The range is 1 to 6000; the default is 10. • (Optional) **interval** *inter-packet-interval*—Enters the interval between sending packets in milliseconds. The range is 1 to 6000; the default value is 20 ms.
Step 4	**frequency** *seconds* **Example:** `Controller(config-ip-sla-jitter)# frequency 45`	(Optional) Configures options for the SLA operation. This example sets the rate at which a specified IP SLAs operation repeats. The range is from 1 to 604800 seconds; the default is 60 seconds.
Step 5	**threshold** *milliseconds* **Example:** `Controller(config-ip-sla-jitter)# threshold 200`	(Optional) Configures threshold conditions. This example sets the threshold of the specified IP SLAs operation to 200. The range is from 0 to 60000 milliseconds.
Step 6	**exit** **Example:** `Controller(config-ip-sla-jitter)# exit`	Exits the SLA operation configuration mode (UDP jitter configuration mode in this example), and returns to global configuration mode.

	Command or Action	Purpose
Step 7	**ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm* [*:ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**] **Example:** `Controller(config)# ip sla schedule 10 start-time now life forever`	Configures the scheduling parameters for an individual IP SLAs operation. • *operation-number*—Enter the RTR entry number. • (Optional) **life**—Sets the operation to run indefinitely (**forever**) or for a specific number of *seconds*. The range is from 0 to 2147483647. The default is 3600 seconds (1 hour). • (Optional) **start-time**—Enters the time for the operation to begin collecting information: To start at a specific time, enter the hour, minute, second (in 24-hour notation), and day of the month. If no month is entered, the default is the current month. Enter **pending** to select no information collection until a start time is selected. Enter **now** to start the operation immediately. Enter **after** *hh:mm:ss* to show that the operation should start after the entered time has elapsed. • (Optional) **ageout** *seconds*—Enter the number of seconds to keep the operation in memory when it is not actively collecting information. The range is 0 to 2073600 seconds, the default is 0 seconds (never ages out). • (Optional) **recurring**—Set the operation to automatically run every day.
Step 8	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

UDP Jitter Configuration

This example shows how to configure a UDP jitter IP SLAs operation:

```
Controller(config)# ip sla 10
Controller(config-ip-sla)# udp-jitter 172.29.139.134 5000
Controller(config-ip-sla-jitter)# frequency 30
Controller(config-ip-sla-jitter)# exit
Controller(config)# ip sla schedule 5 start-time now life forever
Controller(config)# end
Controller# show ip sla configuration 10
IP SLAs, Infrastructure Engine-II.

Entry number: 10
Owner:
Tag:
Type of operation to perform: udp-jitter
Target address/Source address: 1.1.1.1/0.0.0.0
Target port/Source port: 2/0
Request size (ARR data portion): 32
```

```
Operation timeout (milliseconds): 5000
Packet Interval (milliseconds)/Number of packets: 20/10
Type Of Service parameters: 0x0
Verify data: No
Vrf Name:
Control Packets: enabled
Schedule:
    Operation frequency (seconds): 30
    Next Scheduled Start Time: Pending trigger
    Group Scheduled : FALSE
    Randomly Scheduled : FALSE
    Life (seconds): 3600
    Entry Ageout (seconds): never
    Recurring (Starting Everyday): FALSE
    Status of entry (SNMP RowStatus): notInService
Threshold (milliseconds): 5000
Distribution Statistics:
    Number of statistic hours kept: 2
    Number of statistic distribution buckets kept: 1
    Statistic distribution interval (milliseconds): 20
Enhanced History:
```

Related Topics

Network Performance Measurement with Cisco IOS IP SLAs, on page 1387

Restrictions on SLAs, on page 1385

Analyzing IP Service Levels by Using the UDP Jitter Operation

Beginning in privileged EXEC mode, follow these steps to configure a UDP jitter operation on the source device:

Before You Begin

You must enable the IP SLAs responder on the target device (the operational target) to configure a UDP jitter operation on the source device.

SUMMARY STEPS

1. **configure terminal**
2. **ip sla** *operation-number*
3. **udp-jitter** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*}] [**source-port** *port-number*] [**control** {**enable** | **disable**}] [**num-packets** *number-of-packets*] [**interval** *interpacket-interval*]
4. **frequency** *seconds*
5. **exit**
6. **ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm* [*:ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**]
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ip sla** *operation-number* **Example:** `Controller(config)# ip sla 10`	Creates an IP SLAs operation, and enters IP SLAs configuration mode.
Step 3	**udp-jitter** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*}] [**source-port** *port-number*] [**control** {**enable** | **disable**}] [**num-packets** *number-of-packets*] [**interval** *interpacket-interval*] **Example:** `Controller(config-ip-sla)# udp-jitter 172.29.139.134 5000`	Configures the IP SLAs operation as a UDP jitter operation, and enters UDP jitter configuration mode. • *destination-ip-address* | *destination-hostname*—Specifies the destination IP address or hostname. • *destination-port*—Specifies the destination port number in the range from 1 to 65535. • (Optional) **source-ip** {*ip-address* | *hostname*}—Specifies the source IP address or hostname. When a source IP address or hostname is not specified, IP SLAs chooses the IP address nearest to the destination. • (Optional) **source-port** *port-number*—Specifies the source port number in the range from 1 to 65535. When a port number is not specified, IP SLAs chooses an available port. • (Optional) **control**—Enables or disables sending of IP SLAs control messages to the IP SLAs responder. By default, IP SLAs control messages are sent to the destination device to establish a connection with the IP SLAs responder. • (Optional) **num-packets** *number-of-packets*—Enters the number of packets to be generated. The range is 1 to 6000; the default is 10. • (Optional) **interval** *inter-packet-interval*—Enters the interval between sending packets in milliseconds. The range is 1 to 6000; the default value is 20 ms.
Step 4	**frequency** *seconds* **Example:** `Controller(config-ip-sla-jitter)# frequency 45`	(Optional) Sets the rate at which a specified IP SLAs operation repeats. The range is from 1 to 604800 seconds; the default is 60 seconds.

	Command or Action	Purpose
Step 5	**exit** **Example:** `Controller(config-ip-sla-jitter)# exit`	Exits UDP jitter configuration mode, and returns to global configuration mode.
Step 6	**ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm* [*:ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**] **Example:** `Controller(config)# ip sla schedule 10 start-time now life forever`	Configures the scheduling parameters for an individual IP SLAs operation. • *operation-number*—Enter the RTR entry number. • (Optional) **life**—Sets the operation to run indefinitely (**forever**) or for a specific number of *seconds*. The range is from 0 to 2147483647. The default is 3600 seconds (1 hour). • (Optional) **start-time**—Enters the time for the operation to begin collecting information: To start at a specific time, enter the hour, minute, second (in 24-hour notation), and day of the month. If no month is entered, the default is the current month. Enter **pending** to select no information collection until a start time is selected. Enter **now** to start the operation immediately. Enter **after** *hh:mm:ss* to show that the operation should start after the entered time has elapsed. • (Optional) **ageout** *seconds*—Enter the number of seconds to keep the operation in memory when it is not actively collecting information. The range is 0 to 2073600 seconds, the default is 0 seconds (never ages out). • (Optional) **recurring**—Set the operation to automatically run every day.
Step 7	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Configuring a UDP Jitter IP SLAs Operation

This example shows how to configure a UDP jitter IP SLAs operation:

```
Controller(config)# ip sla 10
Controller(config-ip-sla)# udp-jitter 172.29.139.134 5000
Controller(config-ip-sla-jitter)# frequency 30
Controller(config-ip-sla-jitter)# exit
Controller(config)# ip sla schedule 5 start-time now life forever
Controller(config)# end
```

```
Controller# show ip sla configuration 10
IP SLAs, Infrastructure Engine-II.

Entry number: 10
Owner:
Tag:
Type of operation to perform: udp-jitter
Target address/Source address: 1.1.1.1/0.0.0.0
Target port/Source port: 2/0
Request size (ARR data portion): 32
Operation timeout (milliseconds): 5000
Packet Interval (milliseconds)/Number of packets: 20/10
Type Of Service parameters: 0x0
Verify data: No
Vrf Name:
Control Packets: enabled
Schedule:
    Operation frequency (seconds): 30
    Next Scheduled Start Time: Pending trigger
    Group Scheduled : FALSE
    Randomly Scheduled : FALSE
    Life (seconds): 3600
    Entry Ageout (seconds): never
    Recurring (Starting Everyday): FALSE
    Status of entry (SNMP RowStatus): notInService
Threshold (milliseconds): 5000
Distribution Statistics:
    Number of statistic hours kept: 2
    Number of statistic distribution buckets kept: 1
    Statistic distribution interval (milliseconds): 20
Enhanced History:
```

Related Topics

UDP Jitter, on page 1390

Analyzing IP Service Levels by Using the ICMP Echo Operation

Beginning in privileged EXEC mode, follow these steps to configure an ICMP echo operation on the source device:

Before You Begin

This operation does not require the IP SLAs responder to be enabled.

SUMMARY STEPS

1. **configure terminal**
2. **ip sla** *operation-number*
3. **icmp-echo** {*destination-ip-address* | *destination-hostname*} [**source-ip** {*ip-address* | *hostname*} | **source-interface** *interface-id*]
4. **frequency** *seconds*
5. **exit**
6. **ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm* [*:ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**]
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**ip sla** *operation-number* **Example:** `Controller(config)# ip sla 10`	Creates an IP SLAs operation and enters IP SLAs configuration mode.
Step 3	**icmp-echo** {*destination-ip-address* | *destination-hostname*} [**source-ip** {*ip-address* | *hostname*} | **source-interface** *interface-id*] **Example:** `Controller(config-ip-sla)# icmp-echo 172.29.139.134`	Configures the IP SLAs operation as an ICMP Echo operation and enters ICMP echo configuration mode. • *destination-ip-address* | *destination-hostname*—Specifies the destination IP address or hostname. • (Optional) **source-ip** {*ip-address* | *hostname*}—Specifies the source IP address or hostname. When a source IP address or hostname is not specified, IP SLAs chooses the IP address nearest to the destination. • (Optional) **source-interface** *interface-id*—Specifies the source interface for the operation.
Step 4	**frequency** *seconds* **Example:** `Controller(config-ip-sla-echo)# frequency 30`	(Optional) Sets the rate at which a specified IP SLAs operation repeats. The range is from 1 to 604800 seconds; the default is 60 seconds.
Step 5	**exit** **Example:** `Controller(config-ip-sla-echo)# exit`	Exits UDP echo configuration mode, and returns to global configuration mode.
Step 6	**ip sla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm* [*:ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**] **Example:** `Controller(config)# ip sla schedule 5 start-time now life forever`	Configures the scheduling parameters for an individual IP SLAs operation. • *operation-number*—Enter the RTR entry number. • (Optional) **life**—Sets the operation to run indefinitely (**forever**) or for a specific number of *seconds*. The range is from 0 to 2147483647. The default is 3600 seconds (1 hour) • (Optional) **start-time**—Enter the time for the operation to begin collecting information:

	Command or Action	Purpose
		To start at a specific time, enter the hour, minute, second (in 24-hour notation), and day of the month. If no month is entered, the default is the current month. Enter **pending** to select no information collection until a start time is selected. Enter **now** to start the operation immediately. Enter **after** *hh:mm:ss* to indicate that the operation should start after the entered time has elapsed. • (Optional) **ageout** *seconds*—Enter the number of seconds to keep the operation in memory when it is not actively collecting information. The range is 0 to 2073600 seconds; the default is 0 seconds (never ages out). • (Optional) **recurring**—Sets the operation to automatically run every day.
Step 7	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Configuring an ICMP Echo IP SLAs Operation

This example shows how to configure an ICMP echo IP SLAs operation:

```
Controller(config)# ip sla 12
Controller(config-ip-sla)# icmp-echo 172.29.139.134
Controller(config-ip-sla-echo)# frequency 30
Controller(config-ip-sla-echo)# exit
Controller(config)# ip sla schedule 5 start-time now life forever
Controller(config)# end
Controller# show ip sla configuration 22
IP SLAs, Infrastructure Engine-II.

Entry number: 12
Owner:
Tag:
Type of operation to perform: echo
Target address: 2.2.2.2
Source address: 0.0.0.0
Request size (ARR data portion): 28
Operation timeout (milliseconds): 5000
Type Of Service parameters: 0x0
Verify data: No
Vrf Name:
Schedule:
    Operation frequency (seconds): 60
    Next Scheduled Start Time: Pending trigger
    Group Scheduled : FALSE
    Randomly Scheduled : FALSE
    Life (seconds): 3600
```

```
    Entry Ageout (seconds): never
    Recurring (Starting Everyday): FALSE
    Status of entry (SNMP RowStatus): notInService
Threshold (milliseconds): 5000
Distribution Statistics:
    Number of statistic hours kept: 2
    Number of statistic distribution buckets kept: 1
    Statistic distribution interval (milliseconds): 20
History Statistics:
    Number of history Lives kept: 0
    Number of history Buckets kept: 15
    History Filter Type: None
Enhanced History:
```

Related Topics

IP SLAs Operation Threshold Monitoring, on page 1389

Monitoring IP SLAs Operations

The following table describes the commands used to display IP SLAs operation configurations and results:

Table 127: Monitoring IP SLAs Operations

show ip sla application	Displays global information about Cisco IOS IP SLAs.
show ip sla authentication	Displays IP SLAs authentication information.
show ip sla configuration [*entry-number*]	Displays configuration values including all defaults for all IP SLAs operations or a specific operation.
show ip sla enhanced-history {**collection-statistics** | **distribution statistics**} [*entry-number*]	Displays enhanced history statistics for collected history buckets or distribution statistics for all IP SLAs operations or a specific operation.
show ip sla ethernet-monitor configuration [*entry-number*]	Displays IP SLAs automatic Ethernet configuration.
show ip sla group schedule [*schedule-entry-number*]	Displays IP SLAs group scheduling configuration and details.
show ip sla history [*entry-number* | **full** | **tabular**]	Displays history collected for all IP SLAs operations
show ip sla mpls-lsp-monitor {**collection-statistics** | **configuration** | **ldp operational-state** | **scan-queue** | **summary** [*entry-number*] | **neighbors**}	Displays MPLS label switched path (LSP) Health Monitor operations.

show ip sla reaction-configuration [*entry-number*]	Displays the configured proactive threshold monitoring settings for all IP SLAs operations or a specific operation.
show ip sla reaction-trigger [*entry-number*]	Displays the reaction trigger information for all IP SLAs operations or a specific operation.
show ip sla responder	Displays information about the IP SLAs responder.
show ip sla statistics [*entry-number* | **aggregated** | **details**]	Displays current or aggregated operational status and statistics.

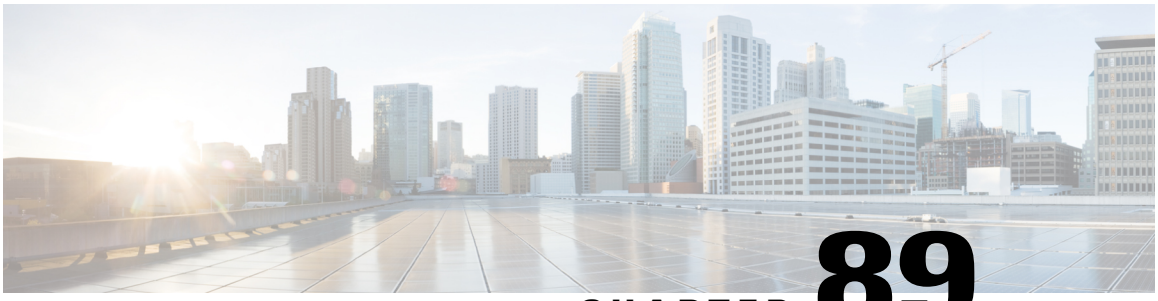
Monitoring IP SLAs Operation Examples

The following example shows all IP SLAs by application:

```
Controller# show ip sla application

        IP Service Level Agreements
Version: Round Trip Time MIB 2.2.0, Infrastructure Engine-III

Supported Operation Types:
        icmpEcho, path-echo, path-jitter, udpEcho, tcpConnect, http
        dns, udpJitter, dhcp, ftp, udpApp, wspApp

Supported Features:
        IPSLAs Event Publisher

IP SLAs low memory water mark: 33299323
Estimated system max number of entries: 24389

Estimated number of configurable operations: 24389
Number of Entries configured  : 0
Number of active Entries      : 0
Number of pending Entries     : 0
Number of inactive Entries    : 0
Time of last change in whole IP SLAs: *13:04:37.668 UTC Wed Dec 19 2012
```

The following example shows all IP SLAs distribution statistics:

```
Controller# show ip sla enhanced-history distribution-statistics

Point by point Enhanced History
Entry    = Entry Number
Int      = Aggregation Interval
BucI     = Bucket Index
StartT   = Aggregation Start Time
Pth      = Path index
Hop      = Hop in path index
Comps    = Operations completed
OvrTh    = Operations completed over thresholds
SumCmp   = Sum of RTT (milliseconds)
SumCmp2L = Sum of RTT squared low 32 bits (milliseconds)
SumCmp2H = Sum of RTT squared high 32 bits (milliseconds)
TMax     = RTT maximum (milliseconds)
TMin     = RTT minimum (milliseconds)

Entry Int BucI StartT     Pth Hop Comps OvrTh SumCmp     SumCmp2L   SumCmp2H   T
Max    TMin
```



CHAPTER 89

Configuring SPAN and RSPAN

This chapter describes how to configure Switched Port Analyzer (SPAN) and Remote SPAN (RSPAN).

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release.

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 controller, you can configure a maximum of 4 source sessions and 64 RSPAN destination 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 controller port as a SPAN destination port, it is no longer a normal controller 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.
- Stacking is supported only on switches running the LAN base image.

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.
- The controller supports up to two local SPAN or RSPAN source sessions.
    - You can run both a local SPAN and an RSPAN source session in the same controller or controller stack. The controller or controller stack supports a total of 64 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 controller stack.

- SPAN sessions do not interfere with the normal operation of the controller. 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 controller 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 controller or controller stack.

RSPAN

The restrictions for RSPAN are as follows:

- RSPAN does not support BPDU packet monitoring or other Layer 2 controller 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 controllers.
- 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 controller 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 controller.
- 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.

Flow-Based SPAN (FSPAN) and Flow-Based RSPAN (FRSPAN)

The restrictions for flow-based SPAN (FSPAN) and flow-based RSPAN (FRSPAN) are as follows:

- You can attach ACLs to only one SPAN or RSPAN session at a time.
- When no FSPAN ACLs are attached, FSPAN is disabled, and all traffic is copied to the SPAN destination ports.
- When you attach an empty FSPAN ACL to a SPAN session, it does not filter packets, and all traffic is monitored.
- Port-based FSPAN sessions can be configured on a stack that includes Catalyst 3750 or Catalyst 3750-E controllers as long as the session only includes Catalyst 3750-X ports as source ports. If the session has any Catalyst 3750 or Catalyst 3750-E ports as source ports, the FSPAN ACL command is rejected. If the session has FSPAN ACL configured, any commands including Catalyst 3750 or Catalyst 3750-E

ports as source ports are rejected. The Catalyst 3750 or Catalyst 3750-E ports can be added as destination ports in an FSPAN session.

- VLAN-based FSPAN sessions cannot be configured on a stack that includes Catalyst 3750 controllers.
- FSPAN ACLs cannot be applied to per-port-per-VLAN sessions. You can configure per-port-per-VLAN sessions by first configuring a port-based session and then configuring specific VLANs to the session. For example:

```
Controller(config)# monitor session session_number source interface interface-id
Controller(config)# monitor session session_number filter vlan vlan-id
Controller(config)# monitor session session_number filter ip access-group {access-list-number | name}
```

Note Both the **filter vlan** and **filter ip access-group** commands cannot be configured at the same time. Configuring one results in rejection of the other.

- EtherChannels are not supported in an FSPAN session.
- FSPAN ACLs with TCP flags or the **log** keyword are not supported.
- If you configure an IPv6 FSPAN ACL when the controller is running the advanced IP Services feature set but later run a different feature set, after rebooting the controller, the controller might lose the IPv6 FSPAN ACL configuration.
- IPv6 FSPAN ACLs are supported only on IPv6-enabled SDM templates. If you configure an IPv6 FSPAN ACL when running an IPv6 enabled SDM template, but later configure a non-IPv6 SDM template and reboot the controller, you lose the IPv6 FSPAN ACL configuration.

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 controller or on another controller 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.

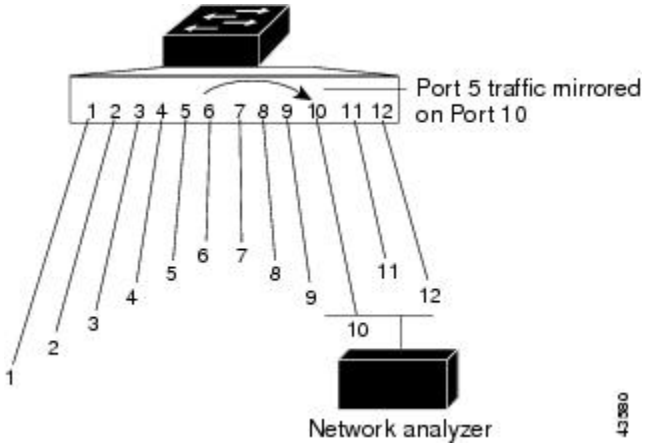
You can use the SPAN or RSPAN destination port to inject traffic from a network security device. For example, if you connect a Cisco Intrusion Detection System (IDS) sensor appliance to a destination port, the IDS device can send TCP reset packets to close down the TCP session of a suspected attacker.

Local SPAN

Local SPAN supports a SPAN session entirely within one controller; all source ports or source VLANs and destination ports are in the same controller or controller 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.

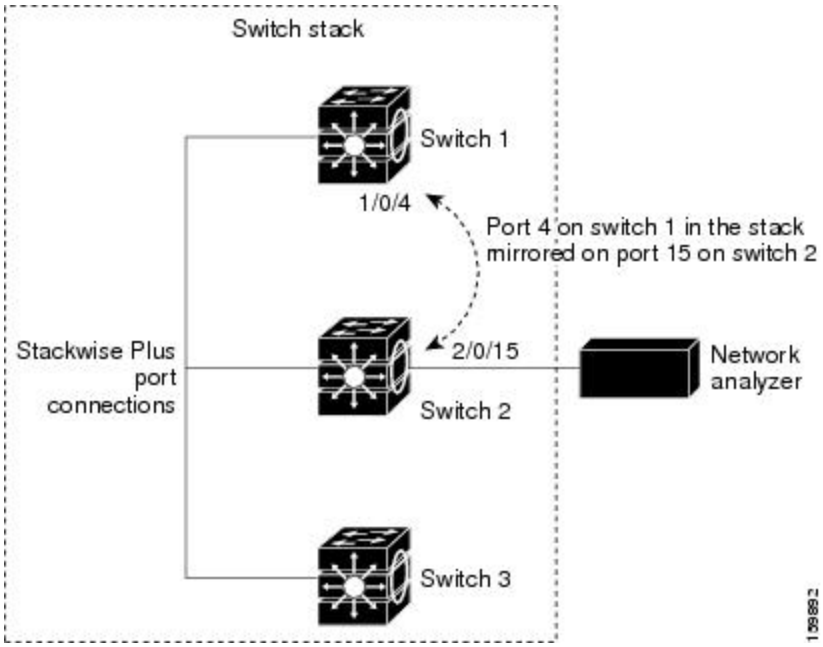
All traffic on port 5 (the source port) is mirrored to port 10 (the destination port). A network analyzer on port 10 receives all network traffic from port 5 without being physically attached to port 5.

Figure 64: Example of Local SPAN Configuration on a Single Device



This is an example of a local SPAN in a controller stack, where the source and destination ports reside on different stack members.

Figure 65: Example of Local SPAN Configuration on a Device Stack

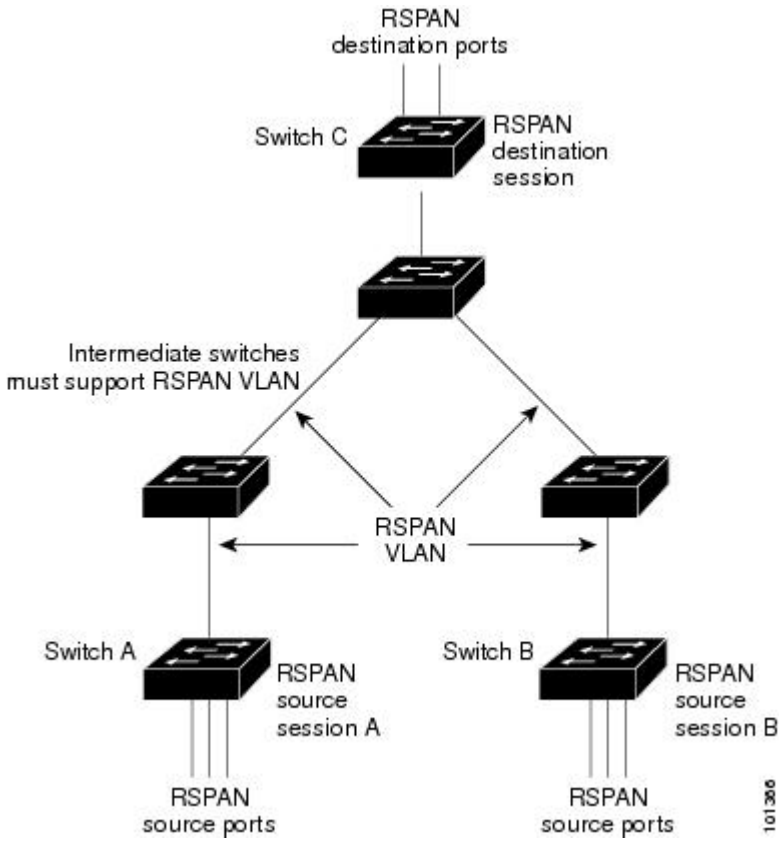


Remote SPAN

RSPAN supports source ports, source VLANs, and destination ports on different Controller (or different controller stacks), enabling remote monitoring of multiple controllers across your network.

This shows source ports on Controller A and Controller 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 controllers. 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 controller must have either ports or VLANs as RSPAN sources. The destination is always a physical port, as shown on Controller C in the figure.

Figure 66: Example of RSPAN Configuration



SPAN and RSPAN Concepts and Terminology

- SPAN Sessions, page 31-4
- 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 re-labeled with the RSPAN VLAN ID and directed over normal trunk ports to the destination controller.

An RSPAN destination session takes all packets received on the RSPAN VLAN, strips off the VLAN tagging, and presents them on the destination port. Its purpose is to present a copy of all RSPAN VLAN packets (except Layer 2 control packets) to the user for analysis.

There can be more than one source session and more than one destination session active in the same RSPAN VLAN. There can also be intermediate controllers separating the RSPAN source and destination sessions. These controllers need not be capable of running RSPAN, but they must respond to the requirements of the RSPAN VLAN.

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 controller. 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 controller. 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, Inter-Switch Link (ISL), 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, ISL, and IEEE 802.1Q tagged packets appear on the destination port.

Controller 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 controller 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 controller 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 controller supports any number of source ports (up to the maximum number of available ports on the controller) and any number of source VLANs (up to the maximum number of VLANs supported). However, the controller supports a maximum of two sessions (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 controller or controller stack as the source port. For an RSPAN session, it is located on the controller containing the RSPAN destination session. There is no destination port on a controller or controller 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 cannot be an EtherChannel group or a VLAN.
- It can participate in only one SPAN session at a time (a destination port in one SPAN session cannot be a destination port for a second SPAN session).
- When it is active, incoming traffic is disabled. The port does not transmit any traffic except that required for the SPAN session. Incoming traffic is never learned or forwarded on a destination port.
- If ingress traffic forwarding is enabled for a network security device, the destination port forwards traffic at Layer 2.
- It does not participate in any of the Layer 2 protocols (STP, VTP, CDP, DTP, PagP).
- A destination port that belongs to a source VLAN of any SPAN session is excluded from the source list and is not monitored.
- The maximum number of destination ports in a controller or controller 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 controllers.

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.

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 controller, not traffic that is routed between VLANs. For example, if a VLAN is being Rx-monitored and the controller routes traffic from another VLAN to the monitored VLAN, that traffic is not monitored and not received on the SPAN destination port.
- STP—A destination port does not participate in STP while its SPAN or RSPAN session is active. The destination port can participate in STP after the SPAN or RSPAN session is disabled. On a source port, SPAN does not affect the STP status. STP can be active on trunk ports carrying an RSPAN VLAN.
- CDP—A SPAN destination port does not participate in CDP while the SPAN session is active. After the SPAN session is disabled, the port again participates in CDP.
- VTP—You can use VTP to prune an RSPAN VLAN between controllers.
- VLAN and trunking—You can modify VLAN membership or trunk settings for source or destination ports at any time. However, changes in VLAN membership or trunk settings for a destination port do not take effect until you remove the SPAN destination configuration. Changes in VLAN membership or trunk settings for a source port immediately take effect, and the respective SPAN sessions automatically adjust accordingly.
- EtherChannel—You can configure an EtherChannel group as a source port but not as a SPAN destination port. When a group is configured as a SPAN source, the entire group is monitored.

  If a physical port is added to a monitored EtherChannel group, the new port is added to the SPAN source port list. If a port is removed from a monitored EtherChannel group, it is automatically removed from the source port list.

  A physical port that belongs to an EtherChannel group can be configured as a SPAN source port and still be a part of the EtherChannel. In this case, data from the physical port is monitored as it participates in the EtherChannel. However, if a physical port that belongs to an EtherChannel group is configured as a SPAN destination, it is removed from the group. After the port is removed from the SPAN session, it rejoins the EtherChannel group. Ports removed from an EtherChannel group remain members of the group, but they are in the inactive or suspended state.

  If a physical port that belongs to an EtherChannel group is a destination port and the EtherChannel group is a source, the port is removed from the EtherChannel group and from the list of monitored ports.
- Multicast traffic can be monitored. For egress and ingress port monitoring, only a single unedited packet is sent to the SPAN destination port. It does not reflect the number of times the multicast packet is sent.
- A private-VLAN port cannot be a SPAN destination port.

- A secure port cannot be a SPAN destination port.

  For SPAN sessions, do not enable port security on ports with monitored egress when ingress forwarding is enabled on the destination port. For RSPAN source sessions, do not enable port security on any ports with monitored egress.

- An IEEE 802.1x port can be a SPAN source port. You can enable IEEE 802.1x on a port that is a SPAN destination port; however, IEEE 802.1x is disabled until the port is removed as a SPAN destination.

  For SPAN sessions, do not enable IEEE 802.1x on ports with monitored egress when ingress forwarding is enabled on the destination port. For RSPAN source sessions, do not enable IEEE 802.1x on any ports that are egress monitored.

SPAN and RSPAN and Device Stacks

Because the stack of controllers represents one logical controller, local SPAN source ports and destination ports can be in different controllers in the stack. Therefore, the addition or deletion of controllers 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 controller is removed from the stack or an inactive session can become active when a controller 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 controller. 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 controller. 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 controllers, it is treated as unloaded on those controllers, and traffic meant for the FSPAN ACL and sourcing on that controller 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 controllers 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.

Default SPAN and RSPAN Configuration

Table 128: 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.

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 controllers.
- For RSPAN configuration, you can distribute the source ports and the destination ports across multiple controllers 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 controllers.
  - All participating controllers support RSPAN.

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.

How to Configure SPAN and RSPAN

Creating a Local SPAN Session

Beginning in privileged EXEC mode, follow these steps to create a SPAN session and specify the source (monitored) ports or VLANs and the destination (monitoring) ports.

SUMMARY STEPS

1. **configure terminal**
2. **no monitor session** {*session_number* | **all** | **local** | **remote**}
3. **monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**]
4. **monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**]}
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**no monitor session** {*session_number* | **all** | **local** | **remote**} **Example:** `Controller(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 3	**monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**] **Example:** `Controller(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 or the source VLAN to monitor. • For source *interface-id*, specify the source port to monitor. Only physical interfaces are valid. • 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) 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. • **tx—**Monitors sent traffic. **Note** You can use the **monitor session** *session_number* **source** command multiple times to configure multiple source ports.
Step 4	**monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**]}	Specifies the SPAN session and the destination port (monitoring port). **Note** For local SPAN, you must use the same session number for the source and destination interfaces.

	Command or Action	Purpose
	Example: `Controller(config)# monitor session 1 destination interface gigabitethernet1/0/2 encapsulation replicate`	• For *session_number*, specify the session number entered in step 3. • 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 5	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Creating a Local SPAN Session and Configuring Incoming Traffic

Beginning in privileged EXEC mode, 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. **configure terminal**
2. **no monitor session** {*session_number* | **all** | **local** | **remote**}
3. **monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**]
4. **monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**] [**ingress** {**dot1q vlan** *vlan-id* | **isl** | **untagged vlan** *vlan-id* | **vlan** *vlan-id*}]}
5. **end**

DETAILED STEPS

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

	Command or Action	Purpose
Step 2	**no monitor session** {*session_number* | **all** | **local** | **remote**} **Example:** `Controller(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 3	**monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**] **Example:** `Controller(config)# monitor session 2 source gigabitethernet1/0/1 rx`	Specifies the SPAN session and the source port (monitored port).
Step 4	**monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**] [**ingress** {**dot1q vlan** *vlan-id* **| isl | untagged vlan** *vlan-id* **| vlan** *vlan-id*}]} **Example:** `Controller(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 3. • 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) [**,** | **-**] Specify 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. ◦ **isl—**Forwards ingress packets with ISL encapsulation. ◦ **untagged vlan** *vlan-id* or **vlan** *vlan-id***—**Accepts incoming packets with untagged encapsulation type with the specified VLAN as the default VLAN. • **dot1q vlan** *vlan-id***—**Accept incoming packets with IEEE 802.1Q encapsulation with the specified VLAN as the default VLAN. • **isl—**Forward ingress packets with ISL encapsulation.

	Command or Action	Purpose
		• **untagged vlan** *vlan-id* or **vlan** *vlan-id*—Accept incoming packets with untagged encapsulation type with the specified VLAN as the default VLAN.
Step 5	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Specifying VLANs to Filter

Beginning in privileged EXEC mode, follow these steps to limit SPAN source traffic to specific VLANs.

SUMMARY STEPS

1. **configure terminal**
2. **no monitor session** {*session_number* | **all** | **local** | **remote**}
3. **monitor session** *session_number* **source interface** *interface-id*
4. **monitor session** *session_number* **filter vlan** *vlan-id* [**,** | **-**]
5. **monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**]}
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**no monitor session** {*session_number* | **all** | **local** | **remote**} **Example:** `Controller(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.

	Command or Action	Purpose
Step 3	**monitor session** *session_number* **source interface** *interface-id* **Example:** `Controller(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 4	**monitor session** *session_number* **filter vlan** *vlan-id* [**,** | **-**] **Example:** `Controller(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 3. • 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 5	**monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**]} **Example:** `Controller(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 3. • 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).
Step 6	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Configuring a VLAN as an RSPAN VLAN

Beginning in privileged EXEC mode, follow these steps to create a new VLAN, then configure it to be the RSPAN VLAN for the RSPAN session.

SUMMARY STEPS

1. **configure terminal**
2. **vlan** *vlan-id*
3. **remote-span**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**vlan** *vlan-id* **Example:** `Controller(config)# vlan 100`	Enter a VLAN ID to create a VLAN, or enter the VLAN ID of an existing VLAN, and enter 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 3	**remote-span** **Example:** `Controller(config-vlan)# remote-span`	Configures the VLAN as an RSPAN VLAN.
Step 4	**end** **Example:** `Controller(config-vlan)# end`	Returns to privileged EXEC mode.

What to Do Next

You must create the RSPAN VLAN in all controllers 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 controller, and VTP propagates it to the other controllers in the VTP domain. For extended-range VLANs (greater than 1005), you must configure RSPAN VLAN on both source and destination controllers and any intermediate controllers.

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

Beginning in privileged EXEC mode, 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. **configure terminal**
2. **no monitor session** {*session_number* | **all** | **local** | **remote**}
3. **monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**]
4. **monitor session** *session_number* **destination remote vlan** *vlan-id*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**no monitor session** {*session_number* | **all** | **local** | **remote**} **Example:** `Controller(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 3	**monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**] **Example:** `Controller(config)# monitor session 1 source interface gigabitethernet1/0/1 tx`	Specifies the RSPAN session and the source port (monitored port). • For *session_number*, the range is 1 to 66. • 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).

	Command or Action	Purpose
		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 4	**monitor session** *session_number* **destination remote vlan** *vlan-id* **Example:** `Controller(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 3. • For *vlan-id*, specify the source RSPAN VLAN to monitor.
Step 5	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Specifying VLANs to Filter

Beginning in privileged EXEC mode, follow these steps to configure the RSPAN source session to limit RSPAN source traffic to specific VLANs.

SUMMARY STEPS

1. **configure terminal**
2. **no monitor session** {*session_number* | **all** | **local** | **remote**}
3. **monitor session** *session_number* **source interface** *interface-id*
4. **monitor session** *session_number* **filter vlan** *vlan-id* [**,** | **-**]
5. **monitor session** *session_number* **destination remote vlan** *vlan-id*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**no monitor session** {*session_number* | **all** | **local** | **remote**} **Example:** `Controller(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 3	**monitor session** *session_number* **source interface** *interface-id* **Example:** `Controller(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 4	**monitor session** *session_number* **filter vlan** *vlan-id* [**,** | **-**] **Example:** `Controller(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 3. • 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 5	**monitor session** *session_number* **destination remote vlan** *vlan-id* **Example:** `Controller(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 3. • For *vlan-id*, specify the RSPAN VLAN to carry the monitored traffic to the destination port.
Step 6	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Creating an RSPAN Destination Session

You configure an RSPAN destination session on a different controller or controller stack; that is, not the controller or controller stack on which the source session was configured.

Beginning in privileged EXEC mode, follow these steps to define the RSPAN VLAN on that controller, to create an RSPAN destination session, and to specify the source RSPAN VLAN and the destination port.

SUMMARY STEPS

1. **configure terminal**
2. **vlan** *vlan-id*
3. **remote-span**
4. **exit**
5. **no monitor session** {*session_number* | **all** | **local** | **remote**}
6. **monitor session** *session_number* **source remote vlan** *vlan-id*
7. **monitor session** *session_number* **destination interface** *interface-id*
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**vlan** *vlan-id* **Example:** `Controller(config)# vlan 901`	Specifies the VLAN ID of the RSPAN VLAN created from the source controller, and enters VLAN configuration mode. If both controllers are participating in VTP and the RSPAN VLAN ID is from 2 to 1005, Steps 2 through 4 are not required because the RSPAN VLAN ID is propagated through the VTP network.
Step 3	**remote-span** **Example:** `Controller(config-vlan)# remote-span`	Identifies the VLAN as the RSPAN VLAN.
Step 4	**exit** **Example:** `Controller(config-vlan)# exit`	Returns to global configuration mode.

	Command or Action	Purpose
Step 5	**no monitor session** {*session_number* | **all** | **local** | **remote**} **Example:** `Controller(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 6	**monitor session** *session_number* **source remote vlan** *vlan-id* **Example:** `Controller(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 7	**monitor session** *session_number* **destination interface** *interface-id* **Example:** `Controller(config)# monitor session 1 destination interface gigabitethernet2/0/1`	Specify the RSPAN session and the destination interface. • For *session_number*, enter the number defined in Step 6. • In an RSPAN destination session, you must use the same session number for the source RSPAN VLAN and the destination port. • 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 8	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Creating an RSPAN Destination Session and Configuring Incoming Traffic

Beginning in privileged EXEC mode, 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. **configure terminal**
2. **no monitor session** {*session_number* | **all** | **local** | **remote**}
3. **monitor session** *session_number* **source remote vlan** *vlan-id*
4. **monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**ingress** {**dot1q vlan** *vlan-id* | **isl** | **untagged vlan** *vlan-id* | **vlan** *vlan-id*}]}
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters the global configuration mode.
Step 2	**no monitor session** {*session_number* | **all** | **local** | **remote**} **Example:** `Controller(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 3	**monitor session** *session_number* **source remote vlan** *vlan-id* **Example:** `Controller(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 4	**monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**ingress** {**dot1q vlan** *vlan-id* | **isl** | **untagged vlan** *vlan-id* | **vlan** *vlan-id*}]} **Example:** `Controller(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 4. 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.

	Command or Action	Purpose
		• (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. ◦ **isl—**Forwards ingress packets with ISL encapsulation. ◦ **untagged vlan** *vlan-id* or **vlan** *vlan-id***—**Forwards incoming packets with untagged encapsulation type with the specified VLAN as the default VLAN.
Step 5	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Configuring an FSPAN Session

Beginning in privileged EXEC mode, 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. **configure terminal**
2. **no monitor session** {*session_number* | **all** | **local** | **remote**}
3. **monitor session** session_number **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**]
4. **monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**]}
5. **monitor session** *session_number* **filter** {**ip** | **ipv6** | **mac**} **access-group** {*access-list-number* | *name*}
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**no monitor session** {*session_number* | **all** | **local** | **remote**} **Example:** `Controller(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 3	**monitor session** session_number **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**] **Example:** `Controller(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*, specify the source port or the source VLAN to monitor. • For source *interface-id*, specify the source port to monitor. Only physical interfaces are valid. • 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. ◦ **tx—**Monitors sent traffic. **Note** You can use the **monitor session** *session_number* **source** command multiple times to configure multiple source ports.
Step 4	**monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**]}	Specifies the SPAN session and the destination port (monitoring port). • For *session_number*, specify the session number entered in Step 3.

	Command or Action	Purpose
	Example: `Controller(config)# monitor session 2 destination interface gigabitethernet1/0/2 encapsulation replicate`	• For **destination**: The monitoring destination specified by 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 5	**monitor session** *session_number* **filter** {**ip** | **ipv6** | **mac**} **access-group** {*access-list-number* | *name*} **Example:** `Controller(config)# monitor session 2 filter ipv6 access-group 4`	Specifies the SPAN session, the types of packets to filter, and the ACLs to use in an FSPAN session. • For *session_number*, specify the session number entered in Step 3. • 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 6	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

Configuring an FRSPAN Session

Beginning in privileged EXEC mode, 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. **configure terminal**
2. **no monitor session** {*session_number* | **all** | **local** | **remote**}
3. **monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**]
4. **monitor session** *session_number* **destination remote vlan** *vlan-id*
5. **vlan** *vlan-id*
6. **remote-span**
7. **exit**
8. **monitor session** *session_number* **filter** {**ip** | **ipv6** | **mac**} **access-group** {*access-list-number* | *name*}
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Controller# configure terminal`	Enters global configuration mode.
Step 2	**no monitor session** {*session_number* | **all** | **local** | **remote**} **Example:** `Controller(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 3	**monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**] **Example:** `Controller(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*, specify the source port or the source VLAN to monitor. • For source *interface-id*, specify the source port to monitor. Only physical interfaces are valid. • 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 SPAN monitors both sent and received traffic. • **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 4	**monitor session** *session_number* **destination remote vlan** *vlan-id* **Example:** `Controller(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 3. • For *vlan-id*, specify the destination RSPAN VLAN to monitor.
Step 5	**vlan** *vlan-id* **Example:** `Controller(config)# vlan 10`	Enters the VLAN configuration mode. For *vlan-id*, specify the source RSPAN VLAN to monitor.
Step 6	**remote-span** **Example:** `Controller(config-vlan)# remote-span`	Specifies that the VLAN you specified in Step 5 is part of the RSPAN VLAN.
Step 7	**exit** **Example:** `Controller(config-vlan)# exit`	Returns to global configuration mode.
Step 8	**monitor session** *session_number* **filter** {**ip** | **ipv6** | **mac**} **access-group** {*access-list-number* | *name*} **Example:** `Controller(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 3. • 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 9	**end** **Example:** `Controller(config)# end`	Returns to privileged EXEC mode.

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 129: Monitoring SPAN and RSPAN Operations

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.

```
Controller(config)# no monitor session 1
Controller(config)# monitor session 1 source interface gigabitethernet1/0/1
Controller(config)# monitor session 1 destination interface gigabitethernet1/0/2
encapsulation replicate
Controller(config)# end
```

This example shows how to remove port 1 as a SPAN source for SPAN session 1:

```
Controller(config)# no monitor session 1 source interface gigabitethernet1/0/1
Controller(config)# end
```

This example shows how to disable received traffic monitoring on port 1, which was configured for bidirectional monitoring:

```
Controller(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.

```
Controller(config)# no monitor session 2
Controller(config)# monitor session 2 source vlan 1 - 3 rx
Controller(config)# monitor session 2 destination interface gigabitethernet1/0/2
Controller(config)# monitor session 2 source vlan 10
Controller(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:

```
Controller(config)# no monitor session 2
Controller(config)# monitor session 2 source gigabitethernet1/0/1 rx
Controller(config)# monitor session 2 destination interface gigabitethernet1/0/2 encapsulation
 replicate ingress dot1q vlan 6
Controller(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:

```
Controller(config)# no monitor session 2
Controller(config)# monitor session 2 source interface gigabitethernet1/0/2 rx
Controller(config)# monitor session 2 filter vlan 1 - 5 , 9
Controller(config)# monitor session 2 destination interface gigabitethernet1/0/1
Controller(config)# end
```

Examples: Creating an RSPAN VLAN

This example shows how to create the RSPAN VLAN 901:

```
Controller(config)# vlan 901
Controller(config-vlan)# remote span
Controller(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:

```
Controller(config)# no monitor session 1
Controller(config)# monitor session 1 source interface gigabitethernet1/0/1 tx
Controller(config)# monitor session 1 source interface gigabitethernet1/0/2 rx
Controller(config)# monitor session 1 source interface port-channel 2
Controller(config)# monitor session 1 destination remote vlan 901
Controller(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:

```
Controller(config)# no monitor session 2
Controller(config)# monitor session 2 source interface gigabitethernet1/0/2 rx
Controller(config)# monitor session 2 filter vlan 1 - 5 , 9
Controller(config)# monitor session 2 destination remote vlan 902
Controller(config)# end
```

This example shows how to configure VLAN 901 as the source remote VLAN and port 1 as the destination interface:

```
Controller(config)# monitor session 1 source remote vlan 901
Controller(config)# monitor session 1 destination interface gigabitethernet2/0/1
Controller(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:

```
Controller(config)# monitor session 2 source remote vlan 901
Controller(config)# monitor session 2 destination interface gigabitethernet1/0/2 ingress
```

```
vlan 6
Controller(config)# end
```

Feature Information for SPAN and RSPAN

This table lists the features in this module and provides links to specific configuration information.

Table 130: Feature Information for SPAN and RSPAN

Feature Name	Releases	Feature Information
Switch Port Analyzer (SPAN)		Allows monitoring of controller traffic on a port or VLAN using a sniffer/analyzer or RMON probe.
Flow-Based Switch Port Analyzer (SPAN)		Provides a method to capture only required (interesting) 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.
SPAN Destination Port Support On Etherchannels		Provides the ability to configure a SPAN destination port on an EtherChannel.

Switch Port Analyzer (SPAN) - Distributed Egress SPAN		Provides distribute 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.

INDEX

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

Z